

Optimizing the Vertical Alignment under Earthwork Block Removal Constraints in Road Construction

by

Md. Faisal Rahman

B.Sc., Bangladesh University of Engineering and Technology, 2009

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

in

The College of Graduate Studies

(Interdisciplinary Studies)

THE UNIVERSITY OF BRITISH COLUMBIA

(Okanagan)

July 2012

© Md. Faisal Rahman 2012

Abstract

The road design problem is usually split into the horizontal alignment, the vertical alignment, and the earthwork scheduling problems. Optimizing the vertical alignment assumes a predetermined horizontal alignment and changes the height of the road at different points to minimize overall construction costs, while maintaining the design requirements. The problem gets complicated because of the natural blocks like rivers, mountains, etc., in the road construction area. Existing vertical alignment models deal with the blocks before the beginning of road construction, which sometimes results in a non-optimal solution. A substantial portion of the total construction cost comes from the earthwork operations. As a result, some existing models only find the optimal earthwork schedule for a predetermined vertical alignment. This research extends a recent mixed integer linear programming (MILP) model that considers blocks when optimizing earthwork operations.

We propose a novel model of vertical alignment that considers blocks. In the model, we propose a novel way of incorporating side-slopes of the road to improve the accuracy of the model. Our numerical results show a considerable improvement in the accuracy of the model without introducing significant computational burden. Finally, we propose another novel model using concepts from the network flow algorithms that gives more than 75% speedup for 87% of the problems for our test set of 280 problems.

Table of Contents

Abstract	ii
Table of Contents	iii
List of Tables	v
List of Figures	viii
List of Algorithms	ix
Glossary of Notation	x
Acknowledgements	xi
Dedication	xii
1 Introduction	1
1.1 Road design	1
1.2 Intelligent road design	5
2 Models for Vertical Alignment and Earthwork Optimization	13
2.1 Spline LP model for vertical alignment optimization	13
2.2 MILP model for earthwork optimization with blocks	24
3 A Basic MILP Model for Vertical Alignment Considering Blocks and Side-slopes	32
3.1 Definitions and model parameters	33
3.2 Model description	36
3.3 Model summary	51
3.4 Model evaluation	54
3.5 Algorithms	55

Table of Contents

4	A Quasi-network-flow Model for Vertical Alignment Considering Blocks and Side-slopes	60
4.1	Definitions and model parameters	61
4.2	Model description	64
4.3	Model summary	72
4.4	Model evaluation	78
5	Numerical Results	80
5.1	Experimental equipments	81
5.2	Basic problem set and parameters	81
5.3	Test configuration 1	82
5.4	Test configuration 2	86
5.5	Test configuration 3	89
5.6	Summary of the results	92
6	Conclusion and Future Work	93
6.1	Conclusion	93
6.2	Future work	93
	Bibliography	96
	Appendix A Asymptotic Bounds	103
	Appendix B Tables	105

List of Tables

1.1	Total transportation expenses (in millions of dollars) by all government levels of Canada, 2005/06-2009/10	2
1.2	Casualty collisions, fatalities, and injuries in roads of Canada, 2005-2009	2
3.1	Number of constraints created by different equations of the basic MILP model	56
4.1	Number of constraints created by different equations of the quasi-network-flow model	79
5.1	The basic problem set showing the roads, their fixed length for all test problems, and their fixed section length for most of the test problems	81
5.2	Problem (R - k , where k is the number of cut and fill entries in the lookup table, AM or NGM for road R) set summary for Test configuration 1	83
5.3	Problem (R - n_g , where n_g is the number of sections per segment for road R) set summary for Test configuration 2	87
5.4	Problem (R - k, n_z, n_r , where $k = k_i^+ = k_i^-$, n_z is the number of blocks, and n_r is the number of access roads for road R) set summary for Test configuration 3	90
B.1	Required time in seconds ($t_{R,k}$ s), normalized time ($\bar{t}_{R,k}$), optimal cost ($C_{R,k}$), optimal cost with corrected volume ($C_{R,k}^{400}$), and percent absolute relative error ($ \epsilon_{R,k} $) for the problems (R - k) of Test configuration 1 (Roads-A,B)	105
B.2	Required time in seconds ($t_{R,k}$ s), normalized time ($\bar{t}_{R,k}$), optimal cost ($C_{R,k}$), optimal cost with corrected volume ($C_{R,k}^{400}$), and percent absolute relative error ($ \epsilon_{R,k} $) for the problems (R - k) of Test configuration 1 (Roads-C,D,E)	106

List of Tables

B.3	Required time in seconds ($t_{R,k}$), optimal cost ($C_{R,k}$), optimal cost with corrected volume ($C_{R,k}^{400}$), and percent absolute relative error ($ \epsilon_{R,k} $) for the problems ($R-k$) of Test configuration 1 (Roads-F,G)	107
B.4	Optimal cost (C_{R,n_g}), normalized cost (\bar{C}_{R,n_g}), required time (t_{R,n_g}), and normalized time (\bar{t}_{R,n_g}) for the problems ($R-n_g$) of Test configuration 2	108
B.5	Cost (C_{R,k,n_z,n_r}^b) and timing (t_{R,k,n_z,n_r}^b) of the basic MILP model, cost (C_{R,k,n_z,n_r}^n) and timing (t_{R,k,n_z,n_r}^n) of the quasi-network-flow model, percent difference in cost (δ_{R,k,n_z,n_r}), and percent improvement (η_{R,k,n_z,n_r}) for the problems ($R-k, n_z, n_r$) of Test configuration 3 (Part-1)	109
B.6	Cost (C_{R,k,n_z,n_r}^b) and timing (t_{R,k,n_z,n_r}^b) of the basic MILP model, cost (C_{R,k,n_z,n_r}^n) and timing (t_{R,k,n_z,n_r}^n) of the quasi-network-flow model, percent difference in cost (δ_{R,k,n_z,n_r}), and percent improvement (η_{R,k,n_z,n_r}) for the problems ($R-k, n_z, n_r$) of Test configuration 3 (Part-2)	110
B.7	Cost (C_{R,k,n_z,n_r}^b) and timing (t_{R,k,n_z,n_r}^b) of the basic MILP model, cost (C_{R,k,n_z,n_r}^n) and timing (t_{R,k,n_z,n_r}^n) of the quasi-network-flow model, percent difference in cost (δ_{R,k,n_z,n_r}), and percent improvement (η_{R,k,n_z,n_r}) for the problems ($R-k, n_z, n_r$) of Test configuration 3 (Part-3)	111
B.8	Cost (C_{R,k,n_z,n_r}^b) and timing (t_{R,k,n_z,n_r}^b) of the basic MILP model, cost (C_{R,k,n_z,n_r}^n) and timing (t_{R,k,n_z,n_r}^n) of the quasi-network-flow model, percent difference in cost (δ_{R,k,n_z,n_r}), and percent improvement (η_{R,k,n_z,n_r}) for the problems ($R-k, n_z, n_r$) of Test configuration 3 (Part-4)	112
B.9	Cost (C_{R,k,n_z,n_r}^b) and timing (t_{R,k,n_z,n_r}^b) of the basic MILP model, cost (C_{R,k,n_z,n_r}^n) and timing (t_{R,k,n_z,n_r}^n) of the quasi-network-flow model, percent difference in cost (δ_{R,k,n_z,n_r}), and percent improvement (η_{R,k,n_z,n_r}) for the problems ($R-k, n_z, n_r$) of Test configuration 3 (Part-5)	113
B.10	Cost (C_{R,k,n_z,n_r}^b) and timing (t_{R,k,n_z,n_r}^b) of the basic MILP model, cost (C_{R,k,n_z,n_r}^n) and timing (t_{R,k,n_z,n_r}^n) of the quasi-network-flow model, percent difference in cost (δ_{R,k,n_z,n_r}), and percent improvement (η_{R,k,n_z,n_r}) for the problems ($R-k, n_z, n_r$) of Test configuration 3 (Part-6)	114

List of Tables

B.11 Cost (C_{R,k,n_z,n_r}^b) and timing (t_{R,k,n_z,n_r}^b) of the basic MILP model, cost (C_{R,k,n_z,n_r}^n) and timing (t_{R,k,n_z,n_r}^n) of the quasi-network-flow model, percent difference in cost (δ_{R,k,n_z,n_r}), and percent improvement (η_{R,k,n_z,n_r}) for the problems ($R-k, n_z, n_r$) of Test configuration 3 (Part-7)	115
B.12 Cost (C_{R,k,n_z,n_r}^b) and timing (t_{R,k,n_z,n_r}^b) of the basic MILP model, cost (C_{R,k,n_z,n_r}^n) and timing (t_{R,k,n_z,n_r}^n) of the quasi-network-flow model, percent difference in cost (δ_{R,k,n_z,n_r}), and percent improvement (η_{R,k,n_z,n_r}) for the problems ($R-k, n_z, n_r$) of Test configuration 3 (Part-8)	116

List of Figures

1.1	Ground profile and corresponding road profile (in 2D)	3
1.2	Crest and sag vertical curves	5
2.1	Optimized vertical alignment for a hypothetical road example . . .	23
2.2	Optimized earthwork schedule for a hypothetical road example (with blocks).	31
3.1	The cross-section of a road with and without side-slopes (for a cut).	40
3.2	The cross-section of a road with and without side-slopes (for a fill).	40
3.3	Approximation of side-slopes (for a cut).	41
3.4	Approximation of side-slopes (for a fill).	41
3.5	Convexity of V_i^+ when $0 < A_i^1 < A_i^2 < A_i^3$	46
4.1	A typical section i at time-step t (where $\vartheta(j) = \varphi(k) = i$).	61
5.1	Percent absolute relative error for NGM and AM approaches of Moreb's model and the basic MILP model with different numbers of cut and fill entries in the lookup table	85
5.2	Normalized time for NGM and AM approaches of Moreb's model and the basic MILP model with different numbers of cut and fill entries in the lookup table	85
5.3	The change of normalized cost with respect to the number of sec- tions per segment for each of the roads	88
5.4	The change of normalized time with respect to the number of sec- tions per segment for each of the roads	88
5.5	A histogram showing the number of problems solved within differ- ent improvement range	91

List of Algorithms

3.1	DS	56
3.2	DSS	57
3.3	$Md_{\max\alpha}$	58
3.4	$MTd_{\max\alpha}$	59

Glossary of Notation

∞	Infinity.
\sum	Summation.
\int	Integral.
\emptyset	The empty set.
$ \mathcal{A} $	The number of elements of the set \mathcal{A} .
$a \in \mathcal{A}$	a is an element of the set \mathcal{A} .
$a \notin \mathcal{A}$	a is not an element of the set \mathcal{A} .
$a \in \mathcal{A} \cup \mathcal{B}$	a is an element of \mathcal{A} or \mathcal{B} .
$a \in \mathcal{A} \setminus \mathcal{B}$	a is an element of \mathcal{A} but not \mathcal{B} .
$\mathcal{A} \subseteq \mathcal{B}$	$a \in \mathcal{A}$ implies $a \in \mathcal{B}$.
$\mathcal{A} \times \mathcal{B}$	Cartesian product $\{(x, y) : x \in \mathcal{A}, y \in \mathcal{B}\}$.
f'	Derivative of a function f .
$\min f$	Minimum of a function f .
$\text{dom } f$	Domain of a function f .
$\omega : \mathcal{A} \rightarrow \mathcal{B}$	ω is a mapping function on the set $\text{dom } \omega \subseteq \mathcal{A}$ into the set \mathcal{B} .
$\Theta(\cdot)$	Asymptotic tight bound.
$\mathcal{O}(\cdot)$	Asymptotic upper bound.
$\Omega(\cdot)$	Asymptotic lower bound.

Acknowledgements

I would like to express my gratitude to all the individuals who were involved directly or indirectly with this work. Specially, I would like to thank my supervisor Dr. Yves Lucet for his guidance during the research. His knowledge and patience were essential for every aspects of this thesis.

I would also like to thank Dr. Warren Hare, who is a member of my graduate committee, for listening to my problems with patience, providing me with valuable guidelines, and helping me grow my interests in Mathematics.

I am very grateful to Dr. Heinz Bauschke, who is also a member of my graduate committee, for his cheerful encouragements and his constructive criticisms that helped me improve my presentation and writing skills.

I would like to express my appreciation to our industrial partner (who wishes to remain anonymous) for providing us with the valuable technical details and practical data for the models.

I would like to express my sincere gratitude to Dr. Donovan Hare for giving me ideas about different optimization techniques and helping me develop a new model to speedup the computational time.

It is my pleasure to thank Jessica Weeres for her contributions to this project and her help to overcome some of the early difficulties of this research, Kasra Bigdeli for helping me implement some previous work in this area, and Sunny Mehrotra for helping me with the ideas about the speedup of the models.

I am indebted to my parents for their prayers and moral support.

A special thanks goes to all my colleagues and friends for their inspiration.

This research is partially funded by an Engage Grants Program from the Natural Sciences and Engineering Research Council (NSERC) of Canada and an Accelerate internship program from the Mathematics of Information Technology and Complex Systems (MITACS).

Dedication

To my brother Fahim and sister Tamanna.

Chapter 1

Introduction

If you would understand anything,
observe its beginning and its devel-
opment.

Aristotle (384 BC - 322 BC)

The first roadways were the tracks made by travelers and their pack animals in about 10000 BC. But it was not before 5000 BC when men started constructing roads because of their socioeconomic needs due to the growth of civilization [Lay93]. Pierre Trésaguet (1716-1796) is recognized to be the first who thought of designing the road scientifically, although very sophisticated road designs were found in the Roman Empire. Since Trésaguet, the area of scientific road design has been enriched by many researchers. In this chapter, we will recall some of these research results and discuss the importance and the basic concepts of road design.

1.1 Road design

Good communication is the essence of economic development for any country. As a result, each year the government allocates a significant part of its annual budget to maintain a high quality road infrastructure. For example, in 2009/10 all levels of Canadian government spent about 28.9 billion dollars for maintaining the existing road infrastructure and constructing new roads [Min10b]. Table 1.1 shows the total expenses of different transportation sectors by all government levels of Canada from 2005/06 to 2009/10 [Min10a]. A good portion of the transportation budget is allocated to road construction and maintenance. It is therefore in the interest of the government to minimize road construction and maintenance cost as much as possible. At the same time, safety is also a very important factor. Every year about 142,376 collisions occur with about 2,739 fatalities (2004 – 2008 average [Min10a]). Table 1.2 shows the statistics of road collisions, injuries, and casualties from 2005 to 2009. Consequently, a scientifically well-designed road can save not only millions of dollars but also a lot of lives. At the same time, it can also consider the environmental impacts [Jha03, Aka03] and the socioeconomic

1.1. Road design

Table 1.1: Total transportation expenses (in millions of dollars) by all government levels of Canada, 2005/06-2009/10

Year	2005/06(R)	2006/07(R)	2007/08(R)	2008/09(R)	2009/10(R)
Air	849	890	897	927	1,183
Rail	264	241	326	387	435
Road	17,042	18,444	22,386	26,935	28,934

Notes: 1. R = Revised.

Table 1.2: Casualty collisions, fatalities, and injuries in roads of Canada, 2005-2009

Year	2005	2006	2007	2008(R)	2009(E)
Casualty collisions	148,154	145,130	141,094	129,816	126,360
Fatalities	2,898	2,884	2,761	2,419	2,200
Injuries	204,764	199,994	192,762	176,433	171,851

Notes:

1. R = Revised. E = Planned and/or estimated.
2. "Casualty collisions" exclude collisions involving property damage only.
3. "Fatalities" include those who died in traffic collision.
4. "Injuries" include all those who suffered any visible injury or complained of pain.

effects [DJ09] of the road construction process. For these reasons, scientific road design has become an important research topic.

Definition 1.1. *Ground profile and road profile.* The ground (respectively road) profile is the vertical profile of the ground before (respectively after) the road is constructed. See Figure 1.1.

The objective of the road construction procedure is to convert the ground profile into the road profile ideally using a 3D model. But due to computational limitations, most researchers divide the road design procedure into horizontal and vertical alignment design [JSJ06, GLA05, GCF88, Fwa89, Sch00].

1.1.1 Horizontal alignment design

Given any two points on earth, the purpose of the horizontal alignment design is to find the best route among all possible routes to build the road. The projection of the three-dimensional ground profile on a horizontal plane is analyzed in the horizontal alignment design procedure. The best route should have the lowest cost while maintaining all the design requirements. Some constraints of the horizontal

1.1. Road design

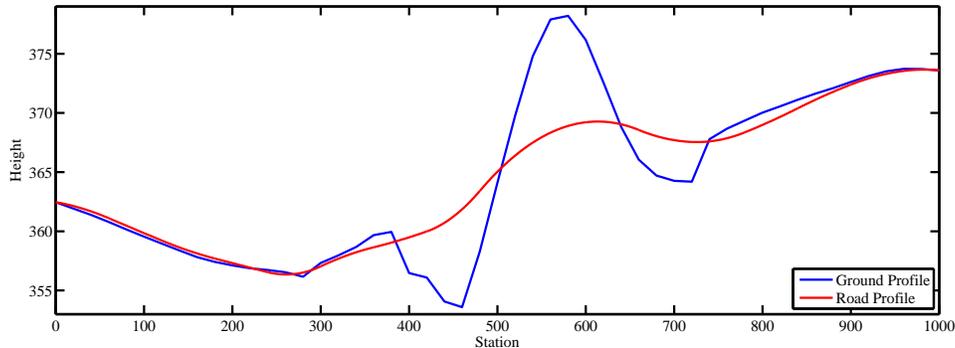


Figure 1.1: Ground profile and corresponding road profile (in 2D)

alignment design come from the laws of mechanics and are sometimes determined empirically. For the basic understanding of the theory of horizontal alignment design we refer to [AAS04].

1.1.2 Vertical alignment design

Vertical alignment is generally calculated for a predetermined horizontal alignment. The input of the vertical alignment is the projection of the three dimensional ground profile sliced by the horizontal alignment on a vertical plane. The output is the final road profile.

Cost components of the vertical alignment design

A major cost component of the vertical alignment design is the earthwork cost. It consists of the earth cutting cost (also known as the excavation cost), the earth filling cost (also known as the embankment cost), and the earth movement cost (also known as the hauling cost).

Pavement is defined to be the hard surface of a road. Pavement costs depend mainly on the thickness and quality of the pavement materials used, both of which depend on the roadbed soil [Fwa89].

Land acquisition costs may vary along the length of the road depending on the geographic location, current state of the location, and type of construction (cut or fill) [Fwa89].

Vehicle operating costs can be divided into primary and secondary vehicle operating costs. The primary vehicle operating cost is the cost of operating the vehicle during the road construction process, while the secondary vehicle operating

cost is the cost of operating vehicle after the road is constructed (mainly user and maintenance costs). Both of these costs depend mainly on the slope of the road.

Fwa [Fwa89] also studied the effects of earthwork cost on the overall construction cost for vertical alignment design. As expected, the overall construction cost linearly increases with the increase of the earth cutting cost, the earth filling cost, and the earth movement cost.

Design requirements for the vertical alignment

Fwa [Fwa89] discussed three major design requirements that must be maintained for vertical alignment, namely the maximum allowable gradient, curvature requirements, and the fixed point constraints.

The maximum allowable gradient mainly depends on the type of terrain of a road, type of vehicles to be operated on the road, and the maximum vehicle speed [Fwa89].

The curvature requirements depend on the minimal stopping sight distance, type of vertical curve (crest or sag), and the length of the vertical curve [AAS04].

Definition 1.2. *Sight distance.* The length of the road visible to the driver when he is driving is called the sight distance. The minimal sight distance required to stop a vehicle to prevent an accident is called the minimal stopping sight distance.

The minimal stopping sight distance depends on the maximum speed of a road and the transportation rules of the region.

Definition 1.3. *Crest and Sag vertical curves.* A crest vertical curve is a concave parabolic curve, whereas a sag vertical curve is a convex parabolic curve.

Figure 1.2 shows the two types of vertical curves. If A is the percentage slope change, L is the length of the vertical curve, S is the sight distance, h_1 is the height of the driver's eye above roadway surface, and h_2 is the height of an object above the roadway surface then the curvature requirements for the crest vertical curves are given by the following equations [AAS04, GLA05, Fwa89],

$$A \leq \frac{100 (\sqrt{2h_1} + \sqrt{2h_2})^2}{2S - L}, \quad \text{if } L \leq S, \quad (1.1a)$$

$$A \leq \frac{100 (\sqrt{2h_1} + \sqrt{2h_2})^2 L}{S^2}, \quad \text{if } L > S, \quad (1.1b)$$

and the curvature requirements for the sag vertical curves are

$$A \leq \frac{120 + 3.5S}{2S - L}, \quad \text{if } L \leq S, \quad (1.1c)$$

$$A \leq \frac{L(120 + 3.5S)}{S^2}, \quad \text{if } L > S. \quad (1.1d)$$

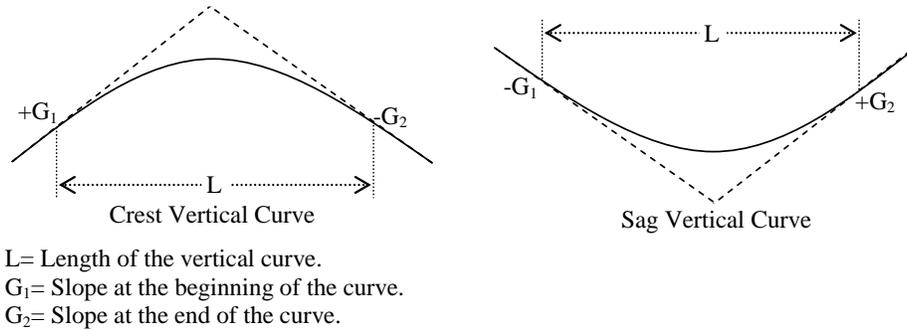


Figure 1.2: Crest and sag vertical curves

It should be noted that the values of h_1, h_2 are predetermined, and all the lengths and distances are given in meters. Equations (1.1a) to (1.1d) may slightly vary depending on the transportation rules of a specific region.

Fixed point constraints enforce that at some points along the road the height of the road must be fixed. Typically, the elevation at the start and end points of a new highway is fixed, and when a new highway intersects with an existing highway the elevation of the intersect points are also fixed.

1.2 Intelligent road design

Intelligent road design refers to the process of planning and designing roads with the aid of sophisticated computer algorithms. Many researchers are working in this area, and many techniques have been proposed over the years. In this section, we will briefly go through some of these techniques with their advantages and disadvantages.

1.2.1 Models for optimizing 3D alignment

Three dimensional (3D) alignment optimization is the process of optimizing horizontal and vertical alignment simultaneously. Because of the complexity of modeling 3D alignment and computational limitations, not many models of the 3D alignment optimization are found in the literature. Hogan [Hog73] presented a dynamic programming model named OPTLOC for optimizing 3D alignment, and later Nicholson, Elms, and Williman [NEW76] proposed a two stage approach where the first stage was similar to Hogan's model, and in the second stage refine-

ments were computed using discrete variational calculus. The problem with the dynamic programming approach is that it is very difficult to handle backtracking alignments and deal with both horizontal and vertical curvatures at the same time.

Chew, Goh, and Fwa [CGF89] proposed a numerical search method that simultaneously optimized horizontal and vertical alignment with a 3D cubic spline, and gave a smooth alignment. They used concepts from optimal control theory. This formulation does not always guarantee global optimality, because it requires a differentiable objective function that cannot incorporate discontinuous cost parameters like land acquisition cost.

Akay [Aka03] developed a model for 3D alignment of forest roads and used a simulated annealing algorithm to solve it. He considered environmental requirements and safety of the drivers in his model. Cheng and Lee [CL06] proposed a heuristic approach to solve the 3D alignment of the highways.

Aruga [Aru05] presented a tabu search method for optimizing 3D alignments of forest roads, and later Aruga et al. [ATSM06] used Dijkstra shortest path method, cubic splines, and tabu search to create an initial alignment, which could be used with Aruga's original model.

Jha [Jha03] employed a criteria-based decision support system for selecting highway alignments using genetic algorithms and geographic information systems (GIS) that considered environmental impacts. Jong and Schonfeld [JS03] used an evolutionary model for optimizing 3D alignment. Kim, Jha, and Son [KJS05] proposed a stepwise genetic algorithm to improve the computational efficiency and the quality of solutions. Kim et al. [KJSK07] further extended the research to incorporate bridges and tunnels in the 3D alignment. More recently, a genetic algorithm was used to solve a multi-objective optimization problem while considering the travel time, vehicle operation, accident, earthwork, land acquisition, and pavement construction costs [MJ09].

Models of the 3D alignment require a large computer memory to run, but most of the models cannot guarantee global optimality. Consequently, most researchers compute a horizontal alignment at first, then for a fixed horizontal alignment they optimize the vertical alignment.

1.2.2 Models for optimizing horizontal alignment

Modeling horizontal alignment is relatively more complex than modeling the vertical alignment, because it involves political, socioeconomic, and environmental issues. Calculus of variations, network optimization, and dynamic programming are the most popular techniques for optimizing horizontal alignments.

Calculus of variations tries to find a curve that connects two points while minimizing the integral of a cost function [Wan95]. So, the type of problems that

calculus of variations usually deals with is very similar to the horizontal alignment problem. Howard, Bramnick, and Shaw developed the Optimum Curvature Principle (OCP) [HBS68] for horizontal alignment using the ideas of calculus of variations. Shaw and Howard [SH81] proposed two integration methods, namely arc of circle algorithm and intrinsic equation procedure for applying OCP. Shaw and Howard [SH82] also applied OCP to find the horizontal alignment of an expressway. The OCP requires determining the local cost functions and needs some approximations and assumptions, but the optimal road derived by the OCP is continuous and globally optimum. Thomson and Skykes [TS88] also used this technique in transportation.

Network optimization is also a very popular technique for horizontal alignment optimization. In this approach, the horizontal alignment problem is modeled as a network problem, i.e., shortest path problem, max-flow min-cut problem, min cost flow problem, etc. Then well-developed network optimization algorithms like Dijkstra's algorithm [Dij59], Bellman-Ford algorithm [CLR01], Ford-Fulkerson Algorithm [LRFF62], etc., are used to compute the solution. Turner and Miles [TM71] developed the Generalized Computer Aid Route Selection (GCARS) system. The GCARS system has a grid network that uses a cost model matrix created by a linear combination of pavement costs and earthwork costs. The GCARS system was further improved by allowing diagonal movements of the network matrix [Tur78]. Later models that used network optimization techniques were developed by Athanassoulis and Calogero [AC73], Parker [Par77], and Trietsch [Tri87b, Tri87a]. Although the network models are simple, easy to implement, and have well-developed algorithms to solve them, the resulting solution is nonsmooth, and calculations require a large amount of computer memory to run.

Dynamic programming [Bel57] is a popular optimization technique developed by Richard Bellman. Although dynamic programming is mostly used for optimizing vertical alignments, some studies like [Tri87b], [OEC73], [JSJ06] discussed the usage of dynamic programming for horizontal alignment. Similar to network models, the dynamic programming approach is simple, easy to implement, and has well-developed algorithms to solve it, but the resulting alignment is nonsmooth.

1.2.3 Models for optimizing the vertical alignment

The vertical alignment problem is a well-defined problem, which does not depend on uncontrollable factors like political issues, socioeconomic concerns, etc. As a result, this problem is studied more comprehensively than the horizontal alignment problem, and there are more existing models.

To model this problem, the length of the potential road is divided into smaller units called sections. The earthwork allocation problem is the problem of finding

an optimal schedule for moving earth between sections. Since earthwork cost is a significant variable cost in the vertical alignment problem, often vertical alignment calculations are performed in two stages. In stage one, the model seeks a feasible vertical alignment, and in stage two, earthwork operations are optimized. Thus many researchers focused just on the earthwork allocation problem. The first attempts to combine both of these stages in a single model can be found in [Bak72] and later in [Ont84].

Popular techniques to model the vertical alignment problem include numerical search, dynamic programming, linear programming, mixed integer linear programming, and quadratic programming.

Numerical search

Earlier models for optimizing vertical alignments used numerical search. This approach exploits a continuous search space, and most of the cost components and constraints can be modeled directly. But the resulting model becomes very complex to solve due to the non-linearity and non-convexity of the model.

Hayman [Hay70] used continuous decision variables for the height at each section and employed a numerical search algorithm to find the optimal heights that satisfied all the constraints. Then straight line segments were used to connect these points to get a piecewise linear vertical alignment. Hayman's algorithm starts with an initial guess of the solution, then seeks a feasible search direction, and continues searching in that direction for a better solution until no such solution can be found. After that it seeks another feasible search direction and repeats the procedure. Obviously, the global optimum is not guaranteed by this algorithm, and the algorithm only gives satisfactory results if the initial guess is close to the optimal solution.

The Transportation and Road Research Laboratory (TRRL) in the UK developed a program named MINERVA [OEC73, Pea73] that utilizes numerical search to find an optimal vertical alignment. Unlike Hayman's algorithm, it gives a smoother solution, since the output alignment consists of straight line segments with a parabolic curve in between. But it also depends on the initial guess, and the algorithm can be stuck at a local optima. Pearman [Pea73] suggested a procedure to find an estimate of the quality of the current solution, compared with the global optimum.

Both Hayman's algorithm and MINERVA depend on a good initial estimate of the solution. Robinson [Rob73] presented a program called VENUS that gave a feasible vertical alignment which could be used as a good initial estimate for any algorithm. VENUS starts by smoothing the ground profile that is then converted into a preliminary road profile by fitting straight lines with the use of a least square process. The alignment is further adjusted to satisfy all the design requirements, and thus a feasible vertical alignment can be found.

Goh et al. [GCF88] presented a discrete and a continuous model for optimizing vertical alignment. In the continuous model, they first formulated the vertical alignment problem as a calculus of variations problem, which is then converted into an optimal control problem [GT88]. Their vertical alignment was approximated by a cubic spline, and the model became a general constrained non-linear optimization problem that could be solved by any known numerical search algorithm. The problem gets very complicated to solve, when earth movement costs are considered in the model. This property makes the model inapplicable for some roads.

Dynamic programming

Dynamic programming [Bel57] is a discrete optimization technique for a multi-stage decision making process developed by Richard Bellman and further improved by Bellman and Dreyfus [BD62]. The basis of the dynamic programming algorithm is the principle of optimality [Bel57].

Definition 1.4. *Admissible and optimal policies.* An admissible policy is a vector of decision that satisfies all the constraints. An admissible policy that minimizes the cost function is referred to as an optimal policy.

Theorem 1.5. *(Principle of optimality)* An optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision [Bel57].

Dynamic programming was first applied to the vertical alignment problem by Puy Huarte [Hua73]. He developed a two phase model, where in the first phase dynamic programming was used to generate an optimal vertical profile, but since the resulting profile is piecewise linear, the alignment is approximated by a set of cubic polynomial functions in phase two.

To improve the existing roads, Murchland [Mur73] developed a program called VALOR that minimized the earthwork cost. This program uses a dynamic programming algorithm to find the optimal vertical alignment, which is represented by a quadratic spline.

Goh et al. [GCF88], Fwa [Fwa89], and Goktepe et al. [GLA05] used a similar approach to optimize vertical alignment using dynamic programming. They discretized the search space by horizontal and vertical grid lines, where each of the horizontal grid lines stood for a state, and each of the vertical grid lines represented a stage in the multi-stage decision process. Intuitively, the output of this formulation gives the height along each vertical grid line, which represents a stage.

A recent approach of optimizing vertical alignment using dynamic programming can be found in [GAA09]. In the study, the authors integrated the Weighted Ground Line Method (WGLM) of earthwork optimization with the dynamic programming model of vertical alignment.

The advantage of dynamic programming is that it is very simple to implement, and there are existing well-developed algorithms to solve it. It can work with nonsmooth, nonconvex, or discontinuous objective functions. The disadvantage is that this approach gives a piecewise linear alignment rather than a smooth one and uses a discrete search space. For some formulations, dynamic programming does not give a realistic solution. In a study ([RL11]), we show that this approach is not appropriate to fulfill many requirements of the vertical alignment problem.

LP, MILP and QP models

Linear programming (LP) is a popular optimization technique. A linear program has a linear objective function and linear constraints in terms of the decision variables. If there exists integer decision variables in a linear program then it is called a mixed integer linear program (MILP). Quadratic programming (QP) is a special type of mathematical optimization problem that has a quadratic objective function subject to linear constraints.

The idea of using linear programming for earthwork optimization was first discussed in [SN71]. Later, Mayer and Stark [MS81] developed the model that minimizes the earth moving cost while balancing the earth quantities in any section. A similar model was developed by Nandgaonkar [Nan81]. Oglesby and Hicks [OH82] used a combination of a graphical method called the mass haul diagram and linear programming for earthwork optimization.

Easa [Eas88b] extended the model of Mayer and Stark [MS81] and proposed a numerical search algorithm that searches for a feasible vertical alignment while optimizing earthwork operations by linear programming as a sub-routine.

Easa [Eas87, Eas88a] suggested that the unit costs for the earthwork operation are not always constant, and for linear unit costs he proposed a quadratic programming model [Eas88a] of the earthwork operation.

Moreb [Mor96] modeled the vertical alignment and earthwork operation in a single linear program. Both Easa's model [Eas88b] and Moreb's model [Mor96] give piecewise linear alignments that require post-processing steps to smooth out the sharp connectivity resulting in a nonoptimal solution for some roads.

Many researchers used polynomials to approximate the road profile to get a smooth solution. ReVelle, Whitlatch, and Wright [RWW97] developed a model that approximated the road profile with a fifth order polynomial while using a linear program to find the optimal coefficients of the polynomial. Aljohani and Moreb

[AM03] gave a more flexible representation using polynomials of any degree. The problem with using polynomials is that it limits the number of ups and downs in the road profile depending on the degree of the polynomial, i.e., this approach will perform badly for long roads.

Moreb and Aljohani [MA04] found a quadratic spline (a piecewise quadratic) representation of the road profile that minimizes earthwork cost using linear programming to avoid the shortcomings of the polynomial approach. Moreb [Mor09] improved the previous model by adding some additional constraints that ensure smoothness at the connectivity of two spline pieces and generalized the technique with a piecewise polynomial of any degree. Koch and Lucet [KL10] further improved the accuracy of the spline model by changing two constraints and reported that up to quadratic spline could be used while maintaining the linear structure of the model. This model can represent the vertical alignment problem very well, but it does not account for the physical blocks like rivers, mountains, etc., that prevent earth movement, and at the same time, the model does not consider side-slopes of the road resulting in an inaccurate model for practical use.

Hare, Koch, and Lucet [HKL11] proposed a mixed integer linear programming model for earthwork optimization that considers blocks. They also proposed algorithms to speed up the computation.

The issue of finding the optimal number of sections and the optimal length of each section was addressed in Ji et al. [JSB⁺10] where a mixed integer linear programming technique was proposed to solve this problem. At the same time, they proposed a minimal cost flow formulation of the earthwork problem that can be solved by standard network flow algorithms.

Other models

Among the other models for the vertical alignment, Goktepe et al. [GLA09] proposed a genetic algorithm based on a constrained curve-fitting method to optimize vertical alignment. In the study, they also considered soil parameters. This study also integrates the Weighted Ground Line Method (WGLM) of earthwork optimization with the genetic algorithm. Garber and Hoel [GH09] used the mass haul diagram, which is a greedy algorithm, to calculate the earthwork operations without considering blocks. It is not known whether the mass haul diagram approach is optimal or not. It can also be solved as a transshipment problem, which is a special case of the network flow problem.

In our research, we extend the models of Moreb [Mor09], Koch and Lucet [KL10], and Hare, Koch, and Lucet [HKL11] to formulate the vertical alignment problem with blocks. We develop a technique to incorporate the side-slopes in the model and still formulate the model as a mixed integer linear program. We also

1.2. Intelligent road design

propose a new model using concepts from the network flow techniques. This new model shows a significant improvement on computational speed over our previous model.

Chapter 2

Models for Vertical Alignment and Earthwork Optimization

What is important is the gradual development of a theory, based on a careful analysis of the facts.

John von Neumann (1903 - 1957)

In this chapter, we present two existing models, namely the spline LP model for vertical alignment [Mor09] and the MILP model for earthwork optimization with blocks [HKL11]. These two models are the basis for our research.

For this thesis, we use the notations from [HKL11] with slight modifications.

2.1 Spline LP model for vertical alignment optimization

The idea of approximating the road profile by a spline was first proposed by Moreb and Aljohani [MA04]. The process was further improved by Moreb [Mor09], followed by Koch and Lucet [KL10]. The resulting linear programming (LP) model is explained in this section.

2.1.1 Definitions and model parameters

Definition 2.1. *Spline segment.* A spline is a piecewise polynomial function, where each polynomial piece is called a spline segment.

The model approximates the road with a spline. The spline has m spline segments, which are indexed by the set $\mathcal{G} = \{1, 2, \dots, m\}$. In [KL10], the authors show that up to a quadratic spline can be used to maintain the linearity of the model. For all $g \in \mathcal{G}$, the equation for a quadratic spline segment is

$$P_g(x) = a_{g,1} + a_{g,2}x + a_{g,3}x^2. \quad (2.1)$$

2.1. Spline LP model for vertical alignment optimization

Definition 2.2. *Section.* The length of a spline segment is further divided into smaller units called sections.

The g^{th} spline segment ($g \in \mathcal{G}$) has n_g sections. So, $n = \sum_{g \in \mathcal{G}} n_g$ is the total number of sections. We index these by the set $\mathcal{S} = \{1, 2, \dots, n\}$. The volume of earth moved from section i to section j is denoted by x_{ij} .

Definition 2.3. For all $g \in \mathcal{G}$, the set $\mathcal{S}_g = \{1, 2, \dots, n_g\}$ contains the indexes of sections for a segment g , and

$$\varkappa : (\mathcal{G}, \mathcal{S}_g) \rightarrow \mathcal{S} \quad (2.2)$$

maps the section index of a spline segment to the actual section index.

The starting point of the i^{th} section of the g^{th} spline segment is denoted by $s_{g,i}$, and the end point is denoted by $s_{g,i+1}$.

Remark 2.4. For all $g \in \mathcal{G} \setminus \{m\}$, $s_{g,n_g+1} = s_{g+1,1}$ by definition.

Definition 2.5. *Cut and fill.* A section is called a cut (respectively fill), if the volume difference between the ground profile and the road profile at the section is positive (respectively negative).

The volume of any section i is represented by $V_i = V_i^+ - V_i^-$, where $V_i^+ \geq 0$ is the cut volume, and $V_i^- \geq 0$ is the fill volume.

Remark 2.6. For any section i , one of V_i^+ and V_i^- must be zero. A cut has $V_i^- = 0$, $V_i^+ \neq 0$ making $V_i = V_i^+ > 0$ and a fill has $V_i^+ = 0$, $V_i^- \neq 0$ making $V_i = -V_i^- < 0$. If a section is neither a cut nor a fill then both V_i^+ and V_i^- are zero.

Definition 2.7. *Borrow and waste pits.* Borrow pits are external sections from which earth can be borrowed, and waste pits are external sections to which earth can be dumped for additional cost.

Remark 2.8. A borrow pit is always a cut, and a waste pit is always a fill.

There are n_b borrow pits, which are indexed by the set $\mathcal{B} = \{n+1, n+2, \dots, n+n_b\}$ and n_w waste pits, which are indexed by the set $\mathcal{W} = \{n+n_b+1, n+n_b+2, \dots, n+n_b+n_w\}$. The index set $\mathcal{N} = \mathcal{S} \cup \mathcal{B} \cup \mathcal{W} = \{1, 2, \dots, n+n_b+n_w\}$ is defined to be the set of all the sections, borrow pits, and waste pits indexes. Also

$$\vartheta : \mathcal{B} \rightarrow \mathcal{S} \quad (2.3)$$

maps the borrow pit index to the section index to which it is attached, and

$$\varphi : \mathcal{W} \rightarrow \mathcal{S} \quad (2.4)$$

2.1. Spline LP model for vertical alignment optimization

maps the waste pit index to the section index to which it is attached. The distance from a borrow or waste pit to the associated section is called the *dead haul distance* and is denoted by \tilde{d}_i where $i \in \mathcal{B} \cup \mathcal{W}$. The capacity of the i^{th} borrow pit (respectively waste pit) is denoted by C_i^b (respectively C_i^w).

Definition 2.9. For each $i \in \mathcal{N}$, the index set $\mathcal{N}_{\rightarrow}^i$ consists of all indexes j such that x_{ij} is a permitted move,

$$\mathcal{N}_{\rightarrow}^i = \left\{ j : \begin{array}{ll} j \in \mathcal{S} \cup \mathcal{W}, j \neq i & \text{if } i \in \mathcal{S} \\ j \in \mathcal{S} & \text{if } i \in \mathcal{B} \\ \emptyset & \text{if } i \in \mathcal{W} \end{array} \right\}, \quad (2.5)$$

and the index set $\mathcal{N}_{\leftarrow}^i$ consists of all indexes j such that x_{ji} is a permitted move,

$$\mathcal{N}_{\leftarrow}^i = \left\{ j : \begin{array}{ll} j \in \mathcal{S} \cup \mathcal{B}, j \neq i & \text{if } i \in \mathcal{S} \\ j \in \mathcal{S} & \text{if } i \in \mathcal{W} \\ \emptyset & \text{if } i \in \mathcal{B} \end{array} \right\}. \quad (2.6)$$

Proposition 2.10. For any $i, j \in \mathcal{N}$, $j \in \mathcal{N}_{\rightarrow}^i$ if, and only if, $i \in \mathcal{N}_{\leftarrow}^j$.

Proof. By definition of $\mathcal{N}_{\rightarrow}^i$ and $\mathcal{N}_{\leftarrow}^j$, i can either be a section or a borrow pit, and j can either be a section or a waste pit. Since movement from borrow pit to waste pit is not allowed we need to consider the following cases.

Case I (i is a section and j is a section): Assume, $j \in \mathcal{N}_{\rightarrow}^i$. Since $i, j \in \mathcal{S}$ and $i \neq j$,

$$\begin{aligned} & j \in (\mathcal{S} - \{i\}), \\ \Rightarrow & i \in (\mathcal{S} - \{j\}), \\ \Rightarrow & i \in (\mathcal{S} \cup \mathcal{B}) - \{j\}, \\ \Rightarrow & i \in \mathcal{N}_{\leftarrow}^j. \end{aligned}$$

Now assume, $i \in \mathcal{N}_{\leftarrow}^j$. Since $i, j \in \mathcal{S}$,

$$\begin{aligned} & i \in (\mathcal{S} - \{j\}), \\ \Rightarrow & j \in (\mathcal{S} - \{i\}), \\ \Rightarrow & j \in (\mathcal{S} \cup \mathcal{W}) - \{i\}, \\ \Rightarrow & j \in \mathcal{N}_{\rightarrow}^i. \end{aligned}$$

Case II (i is a section and j is a waste pit): Assume, $j \in \mathcal{N}_{\rightarrow}^i$. So, $i \in \mathcal{S}$ and $j \in \mathcal{W}$ implies $i \in \mathcal{N}_{\leftarrow}^j$ by the definition of $\mathcal{N}_{\leftarrow}^j$. Now assume, $i \in \mathcal{N}_{\leftarrow}^j$. Since

2.1. Spline LP model for vertical alignment optimization

$i \in \mathcal{S}$ and $j \in \mathcal{W}$,

$$\begin{aligned} & i \in (\mathcal{S} - \{j\}), \\ \Rightarrow & j \in (\mathcal{S} - \{i\}), \\ \Rightarrow & j \in (\mathcal{S} \cup \mathcal{W}) - \{i\}, \\ \Rightarrow & j \in \mathcal{N}_{\rightarrow}^i. \end{aligned}$$

Case III (i is a borrow pit and j is a section): Assume, $j \in \mathcal{N}_{\rightarrow}^i$. Since $i \in \mathcal{B}$ and $j \in \mathcal{S}$,

$$\begin{aligned} & j \in (\mathcal{S} - \{i\}), \\ \Rightarrow & i \in (\mathcal{S} - \{j\}), \\ \Rightarrow & i \in (\mathcal{S} \cup \mathcal{B}) - \{j\}, \\ \Rightarrow & i \in \mathcal{N}_{\leftarrow}^j. \end{aligned}$$

Now assume, $i \in \mathcal{N}_{\leftarrow}^j$. So, $i \in \mathcal{B}$ and $j \in \mathcal{S}$ implies $j \in \mathcal{N}_{\rightarrow}^i$ by the definition of $\mathcal{N}_{\rightarrow}^i$. \square

Definition 2.11. The set \mathcal{N}^2 consists of all pairs of indexes (i, j) such that x_{ij} is a permitted move, i.e.,

$$\begin{aligned} \mathcal{N}^2 &= \{(i, j) : j \in \mathcal{N}_{\rightarrow}^i, i \in \mathcal{N}_{\leftarrow}^j\}, \\ &= \{(i, j) : j \in \mathcal{N}_{\rightarrow}^i\}, \text{ [By Proposition 2.10].} \end{aligned} \quad (2.7)$$

Proposition 2.12. $\mathcal{N}^2 \subseteq \mathcal{N} \times \mathcal{N}$.

Proof. Definition 2.9 and 2.11 complete the proof. \square

Cost components

Definition 2.13. *Excavation, hauling, and embankment cost.* The cost for cutting earth from a section is called the excavation cost, the cost for moving earth from one section to another section is called the hauling cost, and the cost for filling a section with earth is called the embankment cost.

For any $i \in \mathcal{N}$, the per unit volume excavation cost is represented by p_i , the per unit volume embankment cost is represented by q_i , and c_{ij} represents the per unit volume hauling cost from section i to section j . If the constant cost c represents

2.1. Spline LP model for vertical alignment optimization

the cost for moving one unit of volume of earth by one unit distance, and d_{ij} is the distance between section i and section j then $c_{ij} = cd_{ij}$, where

$$d_{ij} = \begin{cases} \text{distance between } i \text{ and } j & \text{if } i, j \in \mathcal{S} \\ d_{\vartheta(i)j} + \tilde{d}_i & \text{if } i \in \mathcal{B} \text{ and } j \in \mathcal{S} \\ d_{i\varphi(j)} + \tilde{d}_j & \text{if } i \in \mathcal{S} \text{ and } j \in \mathcal{W}. \end{cases} \quad (2.8)$$

It should be noted that the model will work with a more sophisticated cost function. For the details of the cost function, we refer to [Koc10].

Decision variables

The decision variables represent the output of the optimization procedure. The decision variables for the problem are

- V_i^+ (for all $i \in \mathcal{S}$): the volume of the earth excavated from any section,
- V_i^- (for all $i \in \mathcal{S}$): the volume of earth embanked to any section,
- x_{ij} (for all $(i, j) \in \mathcal{N}^2$): the volume of earth moved from one section to another,
- $a_{g,k}$ (for all $g \in \mathcal{G}, k \in \{1, 2, 3\}$): the coefficient of the quadratic polynomials,
- u_i (for all $i \in \mathcal{S}$): the difference in height between the ground profile and the road profile for any section.

Design parameters

The design parameters represent the input of the optimization procedure, which will be provided by the end-user. The design parameters for the problem are

- m : the number of spline segments,
- n_g (for all $g \in \mathcal{G}$): the number of sections per spline segment,
- $s_{g,i}$ (for all $g \in \mathcal{G}, i \in \mathcal{S}_g$): the start and end points of the sections,
- $h_{g,i}$ (for all $g \in \mathcal{G}, i \in \mathcal{S}_g$): the average height of the ground for each section,
- \bar{u}_i and \underline{u}_i (for all $i \in \mathcal{S}$): the upper and the lower bound of the variable u_i ,
- A_i (for all $i \in \mathcal{S}$): the area of each section,

2.1. Spline LP model for vertical alignment optimization

- L (respectively U): the lower (respectively upper) bound of the slope,
- H_A (respectively H_B): the starting (respectively ending) elevation,
- H'_A (respectively H'_B): the starting (respectively ending) slope,
- p_i , q_j , and c_{ij} (for all $i \in \mathcal{S} \cup \mathcal{B}$, $j \in \mathcal{S} \cup \mathcal{W}$): excavation, embankment, and hauling costs, and
- the locations of the borrow and waste pits.

2.1.2 Objective and constraints

Using the definitions and model parameters explained in Section 2.1.1, we formulate the model of [Mor09] with modifications from [KL10].

Objective function

$$\min \sum_{i \in \mathcal{S}} (p_i V_i^+ + q_i V_i^-) + \sum_{i \in \mathcal{B}} \sum_{j \in \mathcal{N}_i^{\rightarrow}} p_i x_{ij} + \sum_{i \in \mathcal{W}} \sum_{j \in \mathcal{N}_i^{\leftarrow}} q_i x_{ji} + \sum_{(i,j) \in \mathcal{N}^2} c_{ij} x_{ij}. \quad (2.9a)$$

Balance constraints

$$\sum_{j \in \mathcal{N}_i^{\rightarrow}} x_{ij} = V_i^+, \quad \text{for all } i \in \mathcal{S}, \quad (2.9b)$$

$$\sum_{j \in \mathcal{N}_i^{\leftarrow}} x_{ji} = V_i^-, \quad \text{for all } i \in \mathcal{S}. \quad (2.9c)$$

Pit constraints

$$\sum_{j \in \mathcal{N}_i^{\rightarrow}} x_{ij} \leq C_i^b, \quad \text{for all } i \in \mathcal{B}, \quad (2.9d)$$

$$\sum_{j \in \mathcal{N}_i^{\leftarrow}} x_{ji} \leq C_i^w, \quad \text{for all } i \in \mathcal{W}. \quad (2.9e)$$

2.1. Spline LP model for vertical alignment optimization

Volume constraints

$$V_i^+ - V_i^- = A_i u_i, \quad \text{for all } i \in \mathcal{S}. \quad (2.9f)$$

Gap constraints

$$h_{g,i} - a_{g,3} \frac{(s_{g,i+1})^2 + s_{g,i+1}s_{g,i} + (s_{g,i})^2}{3} - a_{g,2} \frac{s_{g,i+1} + s_{g,i}}{2} - a_{g,1} = u_{\neq(g,i)},$$

for all $g \in \mathcal{G}, i \in \mathcal{S}_g$. (2.9g)

Slope constraints

$$L \leq a_{g,2} + 2a_{g,3}s_{g,i} \leq U, \quad \text{for all } g \in \mathcal{G}, i \in \{1, n_g\}. \quad (2.9h)$$

Smoothness constraints

$$a_{g,1} + a_{g,2}s_{g,1} + a_{g,3}s_{g,1}^2 = a_{(g-1),1} + a_{(g-1),2}s_{g,1} + a_{(g-1),3}s_{g,1}^2,$$

for all $g \in \mathcal{G} \setminus \{1\}$, (2.9i)

$$a_{g,2} + 2a_{g,3}s_{g,1} = a_{(g-1),2} + 2a_{(g-1),3}s_{g,1}, \quad \text{for all } g \in \mathcal{G} \setminus \{1\}. \quad (2.9j)$$

Fixed point constraints

$$a_{1,1} + a_{1,2}s_{1,1} + a_{1,3}s_{1,1}^2 = H_A, \quad (2.9k)$$

$$a_{m,1} + a_{m,2}s_{m,n_m} + a_{m,3}s_{m,n_m}^2 = H_B, \quad (2.9l)$$

$$a_{1,2} + 2a_{1,3}s_{1,1} = H'_A, \quad (2.9m)$$

$$a_{m,2} + 2a_{m,3}s_{m,n_m} = H'_B. \quad (2.9n)$$

Bounds

$$x_{ij} \geq 0, \quad \text{for all } (i, j) \in \mathcal{N}^2, \quad (2.9o)$$

$$V_i^+ \geq 0, \quad \text{for all } i \in \mathcal{S}, \quad (2.9p)$$

$$V_i^- \geq 0, \quad \text{for all } i \in \mathcal{S}, \quad (2.9q)$$

$$\underline{u}_i \leq u_i \leq \bar{u}_i, \quad \text{for all } i \in \mathcal{S}, \quad (2.9r)$$

$$a_{g,k} \in \mathbb{R}, \quad \text{for all } g \in \mathcal{G}, k \in \{1, 2, 3\}. \quad (2.9s)$$

2.1.3 Model discussion

The objective function, represented by Equation (2.9a), minimizes the total of excavation, embankment, and hauling costs.

The balance constraints, represented by Equations (2.9b) and (2.9c), ensure that the total volume moved from a section is equal to the volume excavated from that section, and the total volume moved to a section equals to the volume embanked to that section.

The pit constraints, represented by Equations (2.9d) and (2.9e), enforce the excavated volume or embanked volume of a borrow or waste pit to be within its capacity.

The volume constraints, represented by Equation (2.9f), force the total volume excavated from or embanked to a section to be equal to the volume difference between ground profile and road profile of that section. Equation (2.9f) also implies that $u_i > 0$ for a cut, $u_i < 0$ for a fill, and $u_i = 0$ for a section that is neither cut nor fill.

Proposition 2.14. *Remark 2.6 holds at the optimal solution.*

Proof. For eventual contradiction, assume that there exists a section i such that $V_i^+ = \delta^+$ and $V_i^- = \delta^-$ in the optimal solution, where $\delta^+ > 0$ and $\delta^- > 0$. If $\delta^+ \geq \delta^-$ (respectively $\delta^+ < \delta^-$), then $V_i^+ = \delta^+ - \delta^-$ and $V_i^- = 0$ (respectively $V_i^+ = 0$ and $V_i^- = \delta^- - \delta^+$) give the same u_i for Equation (2.9f) with a lower objective value. \square

For any section i , u_i is defined to be the difference between the average height of the ground and the height of the potential road at that section. This definition was written as the gap constraints in [Mor09]. For all $g \in \mathcal{G}, i \in \mathcal{S}_g$, the gap constraints are

$$h_{g,i} - P_g \left(\frac{s_{g,i+1} + s_{g,i}}{2} \right) = u_{\varkappa(g,i)}. \quad (2.10)$$

2.1. Spline LP model for vertical alignment optimization

Equation (2.10) checks the gap at the middle point of a section, which is a source of inaccuracy. The gap constraints were improved in [KL10] by considering area gap instead of the height gap. For all $g \in \mathcal{G}$, $i \in \mathcal{S}_g$, the improved gap constraints are

$$\int_{s_{g,i}}^{s_{g,i+1}} h_{g,i} dx - \int_{s_{g,i}}^{s_{g,i+1}} P_g(x) dx = \int_{s_{g,i}}^{s_{g,i+1}} u_{\varkappa(g,i)} dx, \quad (2.11)$$

resulting in

$$\begin{aligned} h_{g,i}(s_{g,i+1} - s_{g,i}) - a_{g,3} \frac{(s_{g,i+1})^3 - (s_{g,i})^3}{3} - a_{g,2} \frac{(s_{g,i+1})^2 - (s_{g,i})^2}{2} \\ - a_{g,1}(s_{g,i+1} - s_{g,i}) = u_{\varkappa(g,i)}(s_{g,i+1} - s_{g,i}). \end{aligned} \quad (2.12)$$

Equation (2.9g) is a simplification of Equation (2.12).

The slope constraints imply that the slope of the spline segments must be bounded. The slope of the polynomial $P_g(x)$ is calculated as

$$P'_g(x) = a_{g,2} + 2a_{g,3}x. \quad (2.13)$$

It should be noted that the lower (L) and the upper (U) bounds of the slope are calculated from the curvature requirements and maximum gradient constraints of the vertical alignment design specifications. Since the spline segments are quadratic, the slope of each segment is linear. The maximum and the minimum of a linear function occurs at the boundary points. So, checking the slope constraint at the beginning point and the end point of each segment is sufficient.

Smoothness constraints enforce the spline segments to be continuous (by Equation (2.9i)) and differentiable (by Equation (2.9j)) at their joining points. Equations (2.9i) and (2.9j) are also known as the C^0 and C^1 continuity constraints respectively.

The fixed point constraints fix the starting and ending elevations (by Equations (2.9k) and (2.9l)) as well as the starting and ending slope (by Equations (2.9m) and (2.9n)).

Equations (2.9o) to (2.9s) state the bounds of the decision variables.

Proposition 2.15. (Model size) *The LP has $\mathcal{O}(n^2)$ continuous variables and $\mathcal{O}(n)$ constraints.*

Proof. The total number of continuous variables is $2n + n_b + n_w + n^2 + n_b n + n_w n + 3m + n \leq 8n + 3n^2$ since $n \geq n_b$, $n \geq n_w$, and $n \geq m$. Each of the Equations from (2.9b) to (2.9g) creates $\mathcal{O}(n)$ constraints. Slope, Continuity, and Endpoint constraints are $\mathcal{O}(m)$ in number. The total number of constraints is $\mathcal{O}(n)$, since $n \geq m$. \square

2.1. Spline LP model for vertical alignment optimization

Example 2.16. We optimize the vertical alignment for a 100m long hypothetical road with no borrow or waste pits. The length of each section is 25m, and the road width is 3.5m. The cost components and design parameters are

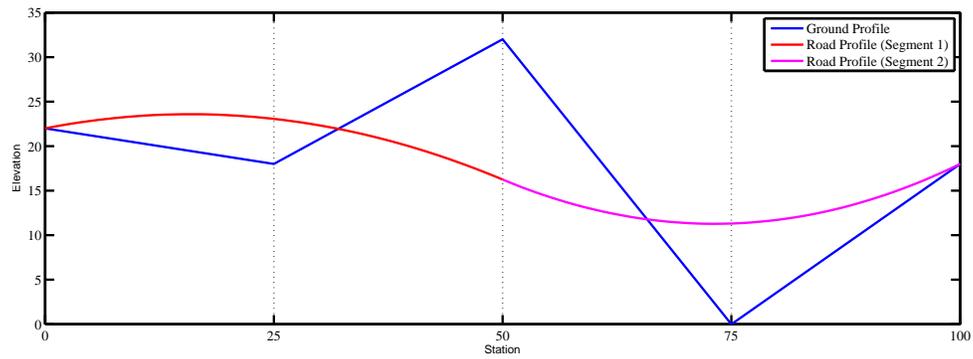
- excavation cost: \$3.13 per unit volume,
- embankment cost: \$7.05 per unit volume,
- hauling cost: \$0.008 per unit volume per unit distance,
- bounds for slope change: $L = -0.5, U = 0.5$,
- start and end elevations: $H_A = 22m, H_B = 18m$,
- station points: $s_{1,1} = 0m, s_{1,2} = 50m, s_{2,1} = 50m, s_{2,2} = 100m$, and
- height of the ground for each section: $h_{1,1} = 20m, h_{1,2} = 25m, h_{2,1} = 16m, h_{2,2} = 9m$.

The resulting LP is as follows.

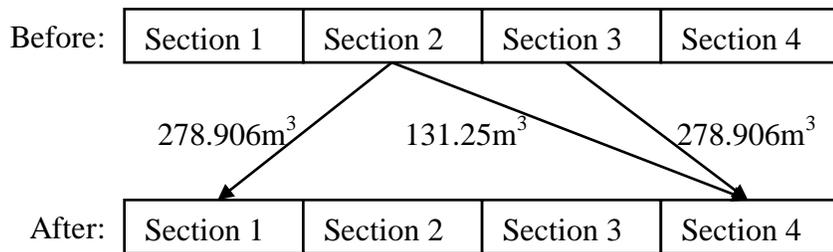
$$\begin{aligned}
 \min \quad & 3.13V_1^+ + 3.13V_2^+ + 3.13V_3^+ + 3.13V_4^+ + 7.05V_1^- + 7.05V_2^- \\
 & + 7.05V_3^- + 7.05V_4^- + 0.02x_{12} + 0.04x_{13} + 0.06x_{14} + 0.02x_{21} \\
 & + 0.02x_{23} + 0.04x_{24} + 0.04x_{31} + 0.02x_{32} + 0.02x_{34} + 0.06x_{41} \\
 & + 0.04x_{42} + 0.02x_{43}, \\
 \text{s.t.} \quad & x_{12} + x_{13} + x_{14} = V_1^+, x_{21} + x_{23} + x_{24} = V_2^+, x_{31} + x_{32} + x_{34} = V_3^+, \\
 & x_{41} + x_{42} + x_{43} = V_4^+, x_{21} + x_{31} + x_{41} = V_1^-, x_{12} + x_{32} + x_{42} = V_2^-, \\
 & x_{13} + x_{23} + x_{43} = V_3^-, x_{14} + x_{24} + x_{34} = V_4^-, V_1^+ - V_1^- = 87.5u_1, \\
 & V_2^+ - V_2^- = 87.5u_2, V_3^+ - V_3^- = 87.5u_3, V_4^+ - V_4^- = 87.5u_4, \\
 & 20 - a_{11} - 12.5a_{12} - 208.333a_{13} = u_1, \\
 & 25 - a_{11} - 37.5a_{12} - 1458.333a_{13} = u_2, \\
 & 16 - a_{21} - 62.5a_{22} - 3958.333a_{23} = u_3, \\
 & 9 - a_{21} - 87.5a_{22} - 7708.333a_{23} = u_4, -0.5 \leq a_{12} \leq 0.5, \\
 & -0.5 \leq a_{12} + 100a_{13} \leq 0.5, -0.5 \leq a_{22} + 100a_{23} \leq 0.5, \\
 & -0.5 \leq a_{22} + 200a_{23} \leq 0.5, a_{21} + 100a_{22} + 10000a_{23} = 18, \\
 & a_{11} = 22, a_{11} + 50a_{12} + 2500a_{13} = a_{21} + 50a_{22} + 2500a_{23}, \\
 & a_{12} + 100a_{13} = a_{22} + 100a_{23}, -10 \leq u_1, u_2, u_3, u_4 \leq 10 \\
 & V_1^+, V_2^+, V_3^+, V_4^+, V_1^-, V_2^-, V_3^-, V_4^- \geq 0.
 \end{aligned}$$

The LP has 27 variables and 36 constraints. The output is shown in Figure 2.1.

2.1. Spline LP model for vertical alignment optimization



(a) Ground profile and road profile.



(b) Earth movement between the sections.

Figure 2.1: Optimized vertical alignment for a hypothetical road example

2.2 MILP model for earthwork optimization with blocks

The mixed integer linear programming (MILP) model for earthwork optimization with blocks was proposed by Hare, Koch, and Lucet [HKL11]. In this model, the earthwork operations are optimized for a predetermined vertical alignment. The output is the optimal schedule for moving earth among the sections.

2.2.1 Definitions and model parameters

In this section, we discuss the definitions and model parameters of this model.

Remark 2.17. Borrow and waste pits are necessary to ensure feasible solutions exist [Koc10, HKL11] (except for the unlikely case where the sum of the volume difference between ground profile and road profile of the cut sections is equal to that of the fill sections). For vertical alignment calculation, borrow pits and waste pits are introduced to give more flexibility to the solution, whereas for earthwork optimization without them it is very unlikely a feasible solution exists.

Definition 2.18. *Block.* A block is an obstacle that needs to be dealt with before any earth can be moved over the location [Koc10, HKL11]. There are n_z blocks, which are indexed by the set $\mathcal{I} = \{1, 2, \dots, n_z\}$ and

$$\varsigma : \mathcal{I} \rightarrow \mathcal{S} \quad (2.14)$$

maps the block index to the section index.

The process is discretized into time-steps to deal with blocks. The set \mathcal{T} contains the indexes of the time-steps, and the binary variable y_{kt} represents whether a block $k \in \mathcal{I}$ is removed at time-step t ($y_{kt} = 1$) or not ($y_{kt} = 0$). It is ensured that at each time-step at least one block will be removed, i.e., the number of time-steps will be at most $n_z + 1$. It should be noted that the introduction of the binary variables y_{kt} , makes the model a mixed integer linear program.

Definition 2.19. *Access roads.* An access road is defined as a road that gives access for earthwork to occur to any section to which it is attached. The index set $\mathcal{R} = \{1, 2, \dots, n_r\}$ contains the indexes of the access roads and

$$\varrho : \mathcal{R} \rightarrow \mathcal{S} \quad (2.15)$$

maps the access road index to the section index.

Assumption 2.20. There is a borrow pit and a waste pit attached with each access road with infinite capacity.

2.2. MILP model for earthwork optimization with blocks

Remark 2.21. Assumption 2.20 is reasonable in a sense that an access road can be used to borrow earth from or dump earth to anywhere in the world.

For all $k \in \mathcal{I}$, the set \mathcal{N}_b^k consists of all the pairs (i, j) such that the block k is in between the sections i and j ,

$$\mathcal{N}_b^k = \{(i, j) : i \leq \varsigma(k) \leq j, i \in \mathcal{N}, j \in \mathcal{N}, i \neq j\}, \quad (2.16)$$

the set $\mathcal{N}_b^{k_1, k_2}$ consists of all indexes i such that the section i is in between the blocks k_1 and k_2 ,

$$\mathcal{N}_b^{k_1, k_2} = \{i \in \mathcal{N} : \varsigma(k_1) \leq i \leq \varsigma(k_2)\}, \quad (2.17)$$

the set $\mathcal{N}_b^{\leftarrow, k}$ consists of all indexes i such that the section i comes before block k ,

$$\mathcal{N}_b^{\leftarrow, k} = \{i \in \mathcal{N} : i \leq \varsigma(k)\}, \quad (2.18)$$

and the set $\mathcal{N}_b^{k, \rightarrow}$ consists of all indexes i such that the section i comes after block k ,

$$\mathcal{N}_b^{k, \rightarrow} = \{i \in \mathcal{N} : i \geq \varsigma(k)\}. \quad (2.19)$$

The set $\bar{\mathcal{I}}$ consists of pairs (k_1, k_2) such that there is no access road in between the blocks k_1 and k_2 ,

$$\bar{\mathcal{I}} = \{(k_1, k_2) : k_1, k_2 \in \mathcal{I}, r \in \mathcal{R}, \varsigma(k_1) \leq \varsigma(k_2), (\varrho(r) < \varsigma(k_1) \text{ or } \varsigma(k_2) > \varrho(r))\}, \quad (2.20)$$

the set $\bar{\mathcal{I}}_{\rightarrow}$ consists of blocks that have no access road after them,

$$\bar{\mathcal{I}}_{\rightarrow} = \{k \in \mathcal{I} : \varrho(r) < \varsigma(k), r \in \mathcal{R}\}, \quad (2.21)$$

and the set $\bar{\mathcal{I}}_{\leftarrow}$ consists of blocks that have no access road before them,

$$\bar{\mathcal{I}}_{\leftarrow} = \{k \in \mathcal{I} : \varsigma(k) > \varrho(r), r \in \mathcal{R}\}. \quad (2.22)$$

Finally, we define M to be the maximum of total cut volume and total fill volume,

$$M = \max\left\{\sum_{i \in \mathcal{S}} V_i^+, \sum_{i \in \mathcal{S}} V_i^-\right\}. \quad (2.23)$$

We also use some definitions and model parameters from Section 2.1.1. The earth movement variables x_{ij} are changed in this model with an additional index for time-step.

2.2. MILP model for earthwork optimization with blocks

- The total number of section is n , which are indexed by the set $\mathcal{S} = \{1, 2, \dots, n\}$. The volume of earth moved from section i to section j at time-step t is denoted by x_{ijt} . The cut and fill volume of a section i are denoted by V_i^+ and V_i^- respectively.
- There are n_b borrow pits, which are indexed by the set $\mathcal{B} = \{n + 1, n + 2, \dots, n + n_b\}$, n_w waste pits, which are indexed by the set $\mathcal{W} = \{n + n_b + 1, n + n_b + 2, \dots, n + n_b + n_w\}$,

$$\vartheta : \mathcal{B} \rightarrow \mathcal{S} \quad (2.24)$$

maps the borrow pit index to the section index to which it is attached,

$$\varphi : \mathcal{W} \rightarrow \mathcal{S} \quad (2.25)$$

maps the waste pit index to the section index to which it is attached. The *dead haul distance* of the i^{th} borrow or waste pit is denoted by \tilde{d}_i , the capacity of the i^{th} borrow pit (respectively waste pit) is denoted by C_i^b (respectively C_i^w), and the set of all the sections, borrow pits, and waste pits indexes is denoted by $\mathcal{N} = \mathcal{S} \cup \mathcal{B} \cup \mathcal{W} = \{1, 2, \dots, n + n_b + n_w\}$.

Cost components

Cost components for this model are identical to the previous model.

Decision variables

The decision variables are

- x_{ijt} (for all $(i, j) \in \mathcal{N}^2, t \in \mathcal{T}$): the volume of earth moved from one section to another at a particular time step,
- y_{kt} (for all $t \in \mathcal{T}, k \in \mathcal{I}$): the block removal indicators.

Design parameters

The design parameters for the MILP are

- n : the number of sections,
- n_z : the number of blocks,
- V_i^+ (for all $i \in \mathcal{S} \cup \mathcal{B}$): the volume of earth excavated from any section,

2.2. MILP model for earthwork optimization with blocks

- V_i^- (for all $i \in \mathcal{S} \cup \mathcal{W}$): the volume of earth embanked to any section,
- ϵ_i^{s+} (respectively ϵ_i^{s-}): the stockpiling tolerance of a cut (respectively fill) section,
- p_i , q_j , and c_{ij} (for all $i \in \mathcal{S} \cup \mathcal{B}$, $j \in \mathcal{S} \cup \mathcal{W}$): excavation, embankment, and hauling costs, and
- the locations of the borrow pits, waste pits, blocks, and access roads.

2.2.2 Objective and constraints

Objective function

$$\begin{aligned} \min \sum_{i \in \mathcal{S}} (p_i V_i^+ + q_i V_i^-) &+ \sum_{i \in \mathcal{B}} \sum_{j \in \mathcal{N}_{\rightarrow}^i} \sum_{t \in \mathcal{T}} p_i x_{ijt} \\ &+ \sum_{i \in \mathcal{W}} \sum_{j \in \mathcal{N}_{\leftarrow}^i} \sum_{t \in \mathcal{T}} q_i x_{jit} + \sum_{(i,j) \in \mathcal{N}^2} \sum_{t \in \mathcal{T}} c_{ij} x_{ijt}. \end{aligned} \quad (2.26a)$$

Balance constraints

$$\sum_{t \in \mathcal{T}} \sum_{j \in \mathcal{N}_{\rightarrow}^i} x_{ijt} = V_i^+, \quad \text{for all } i \in \mathcal{S}, \quad (2.26b)$$

$$\sum_{t \in \mathcal{T}} \sum_{j \in \mathcal{N}_{\leftarrow}^i} x_{jit} = V_i^-, \quad \text{for all } i \in \mathcal{S}. \quad (2.26c)$$

Pit constraints

$$\sum_{t \in \mathcal{T}} \sum_{j \in \mathcal{N}_{\rightarrow}^i} x_{ijt} \leq C_i^b, \quad \text{for all } i \in \mathcal{B}, \quad (2.26d)$$

$$\sum_{t \in \mathcal{T}} \sum_{j \in \mathcal{N}_{\leftarrow}^i} x_{jit} \leq C_i^w, \quad \text{for all } i \in \mathcal{W}. \quad (2.26e)$$

Volume constraints

$$V_i^+ - V_i^- = V_i, \quad \text{for all } i \in \mathcal{S}. \quad (2.26f)$$

Stockpiling constraints

$$\sum_{t=1}^u \sum_{j \in \mathcal{N}_{\rightarrow}^i} x_{ijt} - \sum_{t=1}^u \sum_{j \in \mathcal{N}_{\leftarrow}^i} x_{jit} \leq V_i + \epsilon_i^{s+},$$

$$\text{for all } i \in \mathcal{S}, u \in \mathcal{T} \setminus \{n_z + 1\}, V_i \geq 0, \quad (2.26g)$$

$$\sum_{t=1}^u \sum_{j \in \mathcal{N}_{\rightarrow}^i} x_{ijt} - \sum_{t=1}^u \sum_{j \in \mathcal{N}_{\leftarrow}^i} x_{jit} \geq V_i - \epsilon_i^{s-},$$

$$\text{for all } i \in \mathcal{S}, u \in \mathcal{T} \setminus \{n_z + 1\}, V_i < 0. \quad (2.26h)$$

Block constraints

$$x_{ijt} \leq My_{kt}, \quad \text{for all } t \in \mathcal{T}, k \in \mathcal{I}, (i, j) \in \mathcal{N}^k, \quad (2.26i)$$

$$x_{ijt} \leq My_{k_1 t} + My_{k_2 t}, \quad \text{for all } t \in \mathcal{T}, (k_1, k_2) \in \bar{\mathcal{I}}, (i, j) \in \mathcal{N}^{k_1, k_2}, \quad (2.26j)$$

$$\sum_{t=0}^u \sum_{j \in \mathcal{N}_{\rightarrow}^{\zeta(k)}} x_{\zeta(k)jt} - \sum_{t=0}^u \sum_{j \in \mathcal{N}_{\leftarrow}^{\zeta(k)}} x_{j\zeta(k)t} \geq y_{ku} V_{\zeta(k)},$$

$$\text{for all } k \in \mathcal{I}, u \in \mathcal{T}, V_{\zeta(k)} \geq 0, \quad (2.26k)$$

$$\sum_{t=0}^u \sum_{j \in \mathcal{N}_{\rightarrow}^{\zeta(k)}} x_{\zeta(k)jt} - \sum_{t=0}^u \sum_{j \in \mathcal{N}_{\leftarrow}^{\zeta(k)}} x_{j\zeta(k)t} \leq y_{ku} V_{\zeta(k)},$$

$$\text{for all } k \in \mathcal{I}, u \in \mathcal{T}, V_{\zeta(k)} < 0, \quad (2.26l)$$

$$\sum_{k \in \mathcal{I}} y_{kt} \geq t, \quad \text{for all } t \in \mathcal{T} \setminus \{n_z + 1\}, \quad (2.26m)$$

$$y_{k,(t+1)} \geq y_{k,t}, \quad \text{for all } k \in \mathcal{I}, t \in \mathcal{T} \setminus \{n_z + 1\}. \quad (2.26n)$$

Bounds

$$x_{ijt} \geq 0, \quad \text{for all } (i, j) \in \mathcal{N}^2, t \in \mathcal{T}, \quad (2.26o)$$

$$y_{kt} \in \{0, 1\}. \quad \text{for all } k \in \mathcal{I}, t \in \mathcal{T}, \quad (2.26p)$$

2.2.3 Model discussion

In the objective function, represented by Equation (2.26a), the sum of the excavation, embankment, and hauling cost is minimized.

The balance constraints, represented by Equations (2.26b) and (2.26c), and the pit constraints, represented by Equations (2.26d) and (2.26e), have the additional sum over the time-steps.

Unlike the vertical alignment model, the right-hand side of the equation representing the volume constraints (2.26f) for the earthwork model is a constant, i.e., the total volume of a section is predetermined.

Introducing time-steps to the earthwork model generates an interesting phenomenon. In the intermediate time-steps temporary stockpiling of earth in different sections occurs. If the stockpiling is not controlled, then temporary blocks may be created. Stockpiling constraints, represented by Equations (2.26h) and (2.26h), are introduced to deal with this phenomenon. These constraints allow a cut section to store an excess of ϵ_i^{s+} volume and a fill section to excavate an excess of ϵ_i^{s-} volume.

If there is a block between two sections then no earth movement is allowed between those sections until the block is removed, which is ensured by Equation (2.26i). No earth movement is allowed among sections that are in between two blocks without an access road until one of the blocks is removed, which is enforced by Equation (2.26j). Constraints (2.26k) and (2.26l) ensure that when the required amount of earth is cut or filled for any section with a block, then the block is considered removed for that section. These constraints are also called the block removal indicators. Constraints (2.26m) certify that at least one block will be removed at each time-step. After a block is removed at any time-step, it must stay removed at all the later time-steps. These redundant constraints are enforced by Equation (2.26n).

Equations (2.26o) to (2.26p) state the bounds for the decision variables.

Proposition 2.22. *(Sufficient conditions for feasibility without blocks) Assume $V_s = \sum_{i \in \mathcal{S}} (V_i^+ - V_i^-)$, $V_b = \sum_{i \in \mathcal{B}} V_i^+$, and $V_w = \sum_{i \in \mathcal{W}} V_i^-$. If there are no blocks, $V_w - V_s \geq 0$, and $V_b + V_s \geq 0$, then the MILP always has a solution.*

2.2. MILP model for earthwork optimization with blocks

Proof. Since the conditions imply that the waste pits and the borrow pits provide sufficient slack, the MILP is always feasible with no blocks. \square

Proposition 2.23. *(Sufficient conditions for feasibility with blocks) Assume $V_s = \sum_{i \in \mathcal{S}} (V_i^+ - V_i^-)$. If there exists an access road with a borrow pit having volume V_b and waste pit having volume V_w such that $V_w - V_s \geq 0$ and $V_b + V_s \geq 0$ holds, then the MILP always has a solution.*

Proof. A feasible solution is to remove the blocks sequentially by borrow and waste pits attached with the access roads. Then by Proposition 2.22, the MILP is always feasible. \square

Corollary 2.24. *If there is at least one access road, then Assumption 2.20 ensures that the MILP always has a solution.*

Proposition 2.25. *(Model size) The MILP has $\mathcal{O}(n^2 n_z)$ continuous variables, $\mathcal{O}(n_z^2)$ binary variables, and $\mathcal{O}(n^2 n_z)$ constraints.*

Proof. The total number of continuous variables is $(n^2 + n_b n + n_w n)(n_z + 1) \leq 3n^2 \times 2n_z$, since $n > n_b$ and $n > n_w$. The total number of binary variables is $n_z(n_z + 1) \leq 2n_z^2$. Equations (2.26b), (2.26c), (2.26d), (2.26e), and (2.26f) create $3n + n_b + n_w = \mathcal{O}(n)$ constraints. Equations (2.26g) and (2.26h) create $n_z n = \mathcal{O}(n_z n)$ stockpiling constraints. Equations (2.26i) and (2.26j) create $\mathcal{O}(n^2 n_z)$, Equations (2.26k) and (2.26l) create $\mathcal{O}(n_z^2)$, Equation (2.26m) creates $\mathcal{O}(n_z)$, and Equation (2.26m) creates $\mathcal{O}(n_z^2)$ block constraints. The total number of constraints is $\mathcal{O}(n^2 n_z)$. \square

Example 2.26. We add a block in Section 3 and an access road in Section 1 to the vertical alignment calculated in Example 2.16. Then we solve the earthwork optimization. The output is shown in Figure 2.2.

2.2. MILP model for earthwork optimization with blocks

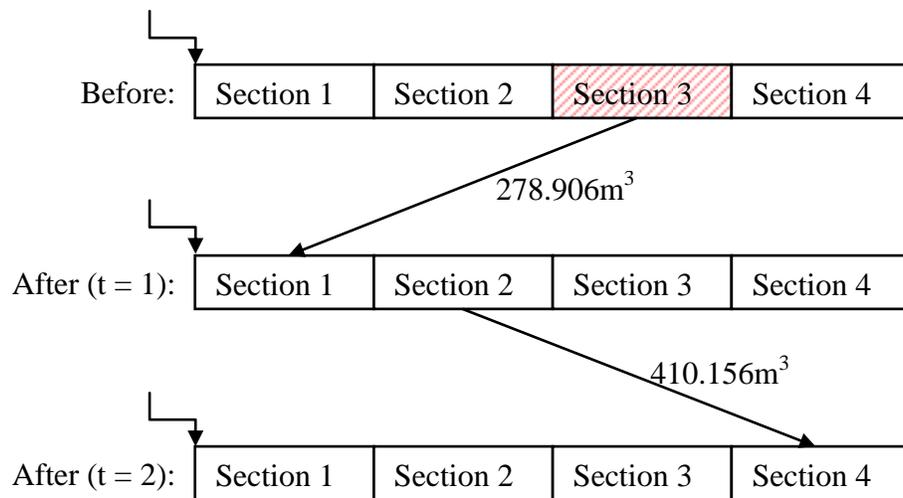


Figure 2.2: Optimized earthwork schedule for a hypothetical road example (with blocks).

Chapter 3

A Basic MILP Model for Vertical Alignment Considering Blocks and Side-slopes¹

If I were again beginning my studies, I would follow the advice of Plato and start with mathematics.

Galileo Galilei (1564 - 1642)

The spline LP model presented in Section 2.1.2 is an improvement over other models of vertical alignment, because it approximates the road profile with a piecewise polynomial and thus results in a smoother solution. But the model does not consider blocks or side-slopes of the road.

The MILP model described in Section 2.2.2 considers blocks, but it only optimizes earthwork for a predetermined vertical alignment. The structure of this model leads to automatic stockpiling or roadside stockpiling. Although roadside stockpiling has theoretical significance, in practice the engineers want to avoid it as much as possible. The model also assumes that the cost for stockpiling is the same as the cost for embankment, which is not true since stockpiling cost is less. Incorporating this in the model, makes it more complex, and since that scenario occurs rarely in practice, the model introduced in this chapter will assume that no roadside stockpiling occurs. We leave the inclusion of roadside stockpiling for future research. The most common form of stockpiling is called pit stockpiling, which uses borrow and waste pits for stockpiling. This feature can be easily incorporated by changing only the input of the model.

In this chapter, we will combine the spline LP model for vertical alignment optimization and the MILP model for earthwork optimization with blocks to describe a spline MILP model for vertical alignment that considers blocks. For this model, we assume that only pit stockpiling occurs, and we add additional constraints to

¹A preliminary patent application was filed in Canada for part of the contents of the current chapter.

prevent roadside stockpiling. We further extend the model to incorporate side-slopes.

3.1 Definitions and model parameters

For this model, we use the definitions and model parameters from Sections 2.1.1 and 2.2.1 that are described briefly in this section with some additional definitions.

- The quadratic spline that represents the road profile has m segments, which are indexed by the set $\mathcal{G} = \{1, 2, \dots, m\}$. For all $g \in \mathcal{G}$ the equation for each segment is

$$P_g(x) = a_{g,1} + a_{g,2}x + a_{g,3}x^2. \quad (3.1)$$

- The g^{th} spline segment has n_g sections, which are indexed by the set $\mathcal{S}_g = \{1, 2, \dots, n_g\}$, $n = \sum_{g \in \mathcal{G}} n_g$ is the total number of sections, which are indexed by the set $\mathcal{S} = \{1, 2, \dots, n\}$,

$$\varkappa : (\mathcal{G}, \mathcal{S}_g) \rightarrow \mathcal{S} \quad (3.2)$$

maps the section index of a spline segment to the actual section index. The volume of earth moved from section i to section j at time-step t is denoted by x_{ijt} , the starting point of the i^{th} section of the g^{th} spline segment is denoted by $s_{g,i}$, and the end point is denoted by $s_{g,i+1}$. The cut and fill volume of a section i are denoted by V_i^+ and V_i^- respectively.

- There are n_b borrow pits, which are indexed by the set $\mathcal{B} = \{n + 1, n + 2, \dots, n + n_b\}$, n_w waste pits, which are indexed by the set $\mathcal{W} = \{n + n_b + 1, n + n_b + 2, \dots, n + n_b + n_w\}$,

$$\vartheta : \mathcal{B} \rightarrow \mathcal{S} \quad (3.3)$$

maps the borrow pit index to the section index to which it is attached,

$$\varphi : \mathcal{W} \rightarrow \mathcal{S} \quad (3.4)$$

maps the waste pit index to the section index to which it is attached. The *dead haul distance* of the i^{th} borrow or waste pit is denoted by \tilde{d}_i , the capacity of the i^{th} borrow pit (respectively waste pit) is denoted by C_i^b (respectively C_i^w), and the set of all the sections, borrow pits, and waste pits indexes is denoted by $\mathcal{N} = \mathcal{S} \cup \mathcal{B} \cup \mathcal{W} = \{1, 2, \dots, n + n_b + n_w\}$.

3.1. Definitions and model parameters

- For each $i \in \mathcal{N}$, the index set $\mathcal{N}_{\rightarrow}^i$ consists of all indexes j such that x_{ijt} is a permitted move, the index set $\mathcal{N}_{\leftarrow}^i$ consists of all indexes j such that x_{jit} is a permitted move, and the set \mathcal{N}^2 consists of all pair of indexes (i, j) such that x_{ijt} is a permitted move for all $t \in \mathcal{T}$. The set \mathcal{H} consists of the fixed points $H_{g,i}$ for the i^{th} section of segment g .
- There are n_z blocks, which are indexed by the set $\mathcal{I} = \{1, 2, \dots, n_z\}$, n_r access roads, which are indexed by the set $\mathcal{R} = \{1, 2, \dots, n_r\}$,

$$\varsigma : \mathcal{I} \rightarrow \mathcal{S} \quad (3.5)$$

maps the block index to the section index,

$$\varrho : \mathcal{R} \rightarrow \mathcal{S} \quad (3.6)$$

maps the access road index to the section index. The set \mathcal{T} contains the indexes of the time-steps, and the binary variable y_{kt} is the block removal indicator for block k at time-step t . At each time step at least one block will be removed, i.e., the number of time steps will be at most $n_z + 1$. Assumption 2.20 states that there is a borrow pit and a waste pit attached with each access road with infinite capacity.

- For all $k \in \mathcal{I}$, the set \mathcal{N}_b^k consists of all the pairs (i, j) such that the block k is in between the sections i and j , the set $\mathcal{N}_b^{k_1, k_2}$ consists of all the indexes i such that the section i is in between the blocks k_1 and k_2 , the set $\mathcal{N}_b^{\leftarrow, k}$ consists of all the indexes i such that the section i is before block k , the set $\mathcal{N}_b^{k, \rightarrow}$ consists of all the indexes i such that the section i is after block k , the set $\overline{\mathcal{I}}$ consists of all the pairs (k_1, k_2) such that there is no access road in between the blocks k_1 and k_2 , the set $\overline{\mathcal{I}}_{\rightarrow}$ consists of blocks that have no access road after them, and the set $\overline{\mathcal{I}}_{\leftarrow}$ consists of blocks that have no access road before them.

3.1.1 Cost components

For this model, we will use the cost components from Section 2.1.1 that are described briefly in this section.

- For a section i , the per unit volume excavation cost is p_i , and the per unit volume embankment cost is q_i .
- The per unit volume hauling cost from section i to section j is $c_{ij} = cd_{ij}$, where c represents the cost of moving one unit volume of earth by one unit

distance, and

$$d_{ij} = \begin{cases} \text{distance between } i \text{ and } j & \text{if } i, j \in \mathcal{S} \\ d_{\vartheta(i)j} + \tilde{d}_i & \text{if } i \in \mathcal{B} \text{ and } j \in \mathcal{S} \\ d_{i\varphi(j)} + \tilde{d}_j & \text{if } i \in \mathcal{S} \text{ and } j \in \mathcal{W}. \end{cases} \quad (3.7)$$

3.1.2 Decision variables

The decision variables are

- V_i^+ (for all $i \in \mathcal{S}$): the volume of earth excavated from any section,
- V_i^- (for all $i \in \mathcal{S}$): the volume of earth embanked to any section,
- x_{ijt} (for all $(i, j) \in \mathcal{N}^2, t \in \mathcal{T}$): the volume of earth moved from one section to another at a particular time step,
- $a_{g,k}$ (for all $g \in \mathcal{G}, k \in \{1, 2, 3\}$): the coefficient of the quadratic polynomials,
- u_i (for all $i \in \mathcal{S}$): the height difference of the ground profile from the road profile for any section,
- y_{kt} (for all $t \in \mathcal{T}, k \in \mathcal{I}$): the block removal indicators,
- b_i (for all $i \in \mathcal{S}$): binary variables to ensure that no roadside stockpiling occurs.

3.1.3 Design parameters

The design parameters for the MILP are

- m : the number of spline segments,
- n_g (for all $g \in \mathcal{G}$): the number of sections per spline segment,
- $s_{g,i}$ (for all $g \in \mathcal{G}, i \in \mathcal{S}_g$): the start and end points of the sections,
- $h_{g,i}$ (for all $g \in \mathcal{G}, i \in \mathcal{S}_g$): the average height of the ground for each section,
- \bar{u}_i and \underline{u}_i (for all $i \in \mathcal{S}$): the upper and lower bound of the variable u_i ,
- A_i (for all $i \in \mathcal{S}$): the area of each section,
- L (respectively U): the lower (respectively upper) bound of the slope,

3.2. Model description

- H_A (respectively H_B): the starting (respectively ending) elevation,
- H'_A (respectively H'_B): the starting (respectively ending) slope,
- $H_{g,i} \in \mathcal{H}$: the fixed points,
- n_z : the number of blocks,
- $A_i^{k_a}, B_i^{k_b}$ (for all $i \in \mathcal{S}$, $k_a \in \{1, 2, \dots, k_i^+\}$, $k_b \in \{1, 2, \dots, k_i^-\}$): area list for side-slope approximation,
- p_i , q_j , and c_{ij} (for all $i \in \mathcal{S} \cup \mathcal{B}$, $j \in \mathcal{S} \cup \mathcal{W}$): excavation, embankment, and hauling costs, and
- the locations of the borrow pits, the waste pits, the blocks, and the access roads.

3.2 Model description

3.2.1 The objective function

The objective of this problem is to find the vertical alignment while minimizing the total excavation cost, embankment cost, and hauling cost. So, the objective function can be written as the following,

$$\begin{aligned} \min \sum_{i \in \mathcal{S}} (p_i V_i^+ + q_i V_i^-) &+ \sum_{i \in \mathcal{B}} \sum_{j \in \mathcal{N}_\rightarrow^i} \sum_{t \in \mathcal{T}} p_i x_{ijt} \\ &+ \sum_{i \in \mathcal{W}} \sum_{j \in \mathcal{N}_\leftarrow^i} \sum_{t \in \mathcal{T}} q_i x_{jit} + \sum_{(i,j) \in \mathcal{N}^2} \sum_{t \in \mathcal{T}} c_{ij} x_{ijt}. \end{aligned} \quad (3.8)$$

3.2.2 Balance constraints

The cut balance constraints ensure that for all sections the total volume moved from a section in all time-steps must be equal to the volume excavated from that section, for all $i \in \mathcal{S}$,

$$\sum_{t \in \mathcal{T}} \sum_{j \in \mathcal{N}_\rightarrow^i} x_{ijt} = V_i^+. \quad (3.9a)$$

3.2. Model description

The fill balance constraints ensures that for all sections the total volume moved to a section in all time-steps must be equal to the volume embanked to that section, for all $i \in \mathcal{S}$,

$$\sum_{t \in \mathcal{T}} \sum_{j \in \mathcal{N}_{\leftarrow}^i} x_{jit} = V_i^-. \quad (3.9b)$$

3.2.3 Pit constraints

The borrow pit constraints ensure that the total excavated volume from a borrow pit does not exceed its limit. So, for all $i \in \mathcal{B}$,

$$\sum_{t \in \mathcal{T}} \sum_{j \in \mathcal{N}_{\rightarrow}^i} x_{ijt} \leq C_i^b. \quad (3.10)$$

Similarly, the waste pit constraints ensure that the total dumped volume to a waste pit does not exceed its capacity. So, for all $i \in \mathcal{W}$,

$$\sum_{t \in \mathcal{T}} \sum_{j \in \mathcal{N}_{\leftarrow}^i} x_{jit} \leq C_i^w. \quad (3.11)$$

3.2.4 Block constraints

If there is a block between two sections then no earth movement is allowed among those two sections until the block is removed. So, for all $t \in \mathcal{T} \setminus \{1\}$, $k \in \mathcal{I}$, and $(i, j) \in \mathcal{N}_b^k$ we have,

$$x_{ijt} \leq My_{k,t-1}. \quad (3.12a)$$

Also, no earth movement is allowed among sections that are in between two blocks without an access road until one of the blocks is removed. So, for all $t \in \mathcal{T} \setminus \{1\}$, $(k_1, k_2) \in \mathcal{I}^2$, and $i, j \in \mathcal{N}_b^{k_1, k_2}$ we have,

$$x_{ijt} \leq My_{k_1, t-1} + My_{k_2, t-1}. \quad (3.12b)$$

If there is no access road before any block then no earth movement is allowed among sections before that block until the block is removed. So, for all $t \in \mathcal{T} \setminus \{1\}$, $k \in \overline{\mathcal{I}}_{\leftarrow}$, and $i, j \in \mathcal{N}_b^{\leftarrow, k}$ we have,

$$x_{ijt} \leq My_{k,t-1}. \quad (3.12c)$$

3.2. Model description

If there is no access road after any block then no earth movement is allowed among sections after that block until the block is removed. So, for all $t \in \mathcal{T} \setminus \{1\}$, $k \in \overline{\mathcal{I}}_{\rightarrow}$, and $i, j \in \mathcal{N}_b^{k, \rightarrow}$ we have,

$$x_{ijt} \leq My_{k,t-1}. \quad (3.12d)$$

The difficulty of combining models of Sections 2.1.2 and 2.2.2 appears in constraints (2.26k) and (2.26l). Both y_{ku} and $V_{\varsigma(k)}$ (for all $k \in \mathcal{I}, u \in \mathcal{T}$) are decision variables in the vertical alignment problem with blocks, so constraints (2.26k) and (2.26l) become quadratic with respect to the decision variables, i.e., the model is no longer a linear program. We use Lemma 3.1 as described in [Bis09] to solve this issue.

Lemma 3.1. *For any optimization problem, if $y \in \{0, 1\}$, $0 \leq x \leq m$, and $w \in \mathbb{R}$ are the decision variables then the following constraints ensure that $w = xy$,*

$$w \leq my, \quad (3.12e)$$

$$w \leq x, \quad (3.12f)$$

$$w \geq x - m(1 - y), \quad (3.12g)$$

$$w \geq 0. \quad (3.12h)$$

Proof. For $y = 0$, Equations (3.12e) and (3.12h) imply $w = 0$, and Equations (3.12f) and (3.12g) imply $0 \leq x \leq m$. For $y = 1$, Equations (3.12f) and (3.12g) imply $w = x$, and Equations (3.12e) and (3.12h) imply $0 \leq x \leq m$. So, for both cases Equations (3.12e), (3.12f), (3.12g) and (3.12h) imply $w = xy$ and $0 \leq x \leq m$. \square

Lemma 3.1 allows us to rewrite the quadratic terms as linear terms by introducing additional continuous variables. For all $k \in \mathcal{I}$ and $t \in \mathcal{T}$, we introduce continuous decision variables W_{kt}^+ and W_{kt}^- such that $W_{kt}^+ = y_{kt}V_{\varsigma(k)}^+$ for the cut sections with blocks and $W_{kt}^- = y_{kt}V_{\varsigma(k)}^-$ for the fill sections with blocks. For all $k \in \mathcal{I}, t \in \mathcal{T}$, the following constraints ensures that $W_{kt}^+ = y_{kt}V_{\varsigma(k)}^+$,

$$W_{kt}^+ \leq M_{\varsigma(k)}^+ y_{kt}, \quad (3.12i)$$

$$W_{kt}^+ \leq V_{\varsigma(k)}^+, \quad (3.12j)$$

$$W_{kt}^+ \geq V_{\varsigma(k)}^+ - M_{\varsigma(k)}^+ (1 - y_{kt}), \quad (3.12k)$$

$$W_{kt}^+ \geq 0, \quad (3.12l)$$

3.2. Model description

and the following constraints ensures that $W_{kt}^- = y_{kt}V_{\varsigma(k)}^-$,

$$W_{kt}^- \leq M_{\varsigma(k)}^- y_{kt}, \quad (3.12m)$$

$$W_{kt}^- \leq V_{\varsigma(k)}^-, \quad (3.12n)$$

$$W_{kt}^- \geq V_{\varsigma(k)}^- - M_{\varsigma(k)}^- (1 - y_{kt}), \quad (3.12o)$$

$$W_{kt}^- \geq 0. \quad (3.12p)$$

The block removal indicator constraints ensure that when the required amount of earth is cut or filled for any section with blocks then the block is removed for that section. For all $k \in \mathcal{I}, u \in \mathcal{T}$, these constraints are written as follows,

$$\sum_{t=0}^u \sum_{j \in \mathcal{N}_{\rightarrow}^{\varsigma(k)}} x_{\varsigma(k)jt} \geq W_{ku}^+, \quad (3.12q)$$

$$\sum_{t=0}^u \sum_{j \in \mathcal{N}_{\leftarrow}^{\varsigma(k)}} x_{j\varsigma(k)t} \geq W_{ku}^-. \quad (3.12r)$$

At least one block must be removed at each time-step. So, for all $t \in \mathcal{T} \setminus \{n_z + 1\}$,

$$\sum_{k \in \mathcal{I}} y_{kt} \geq t. \quad (3.12s)$$

The monotonicity constraints are optional constraints that ensure that if a block is removed at any time-step then it will stay removed for all the later time-steps. So, for all $k \in \mathcal{I}, t \in \mathcal{T} \setminus \{n_z + 1\}$,

$$y_{k,(t+1)} \geq y_{k,t}. \quad (3.12t)$$

3.2.5 Volume constraints

The volume constraints ensure that the total volume excavated from a section or embanked to a section must be equal to the volume difference between the ground profile and the road profile for that section. For all $i \in \mathcal{S}$, if the area for is A_i then the volume constraints can be written as

$$V_i^+ - V_i^- = A_i u_i. \quad (3.13a)$$

Equation (3.13a) also implies that $u_i > 0$ for a cut, $u_i < 0$ for a fill, and $u_i = 0$ for a section that is neither cut nor fill.

3.2. Model description

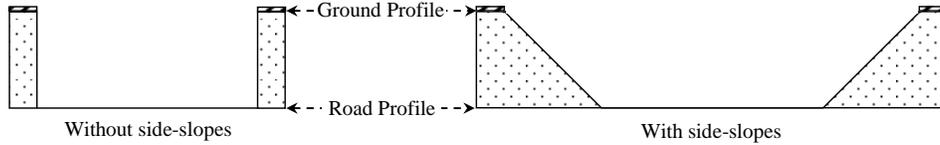


Figure 3.1: The cross-section of a road with and without side-slopes (for a cut).

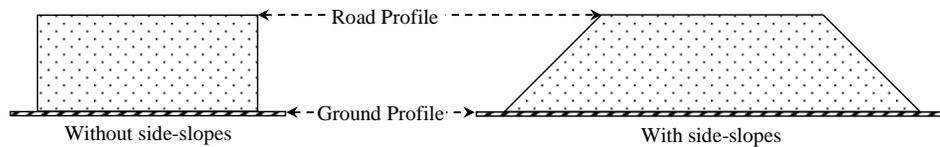


Figure 3.2: The cross-section of a road with and without side-slopes (for a fill).

Side-slopes

Constructed roads must have side-slopes for stability reasons. Not considering side-slopes in the model introduces significant errors in the volume.

Definition 3.2. *Side-slopes.* Side-slopes are defined as the gradual decrease (respectively increase) of height from the road profile to the ground profile for a fill (respectively cut) section.

Figures 3.1 and 3.2 show the cross-section of a road with and without side-slopes for a cut and a fill respectively. Without side-slopes the cross-section of a road is a rectangle, and with side-slopes it is typically a trapezoid. So, for a fixed section length and width the volume of a section is a linear function of the height of the road when side-slopes are not considered. However, when side-slopes are considered, the volume becomes a quadratic function of the height for typical roads, and thus the linear structure of the model is violated. Some real world examples exist where the constructed road must have very complicated side-slopes. So, we seek a general way to approximate side-slopes that not only performs very well for the typical roads but also performs reasonably for worst case scenarios. Since the cost parameters for this model are estimations with a 5% expected error, the model will be practical if the approximation of side-slopes is within the error margin.

3.2. Model description

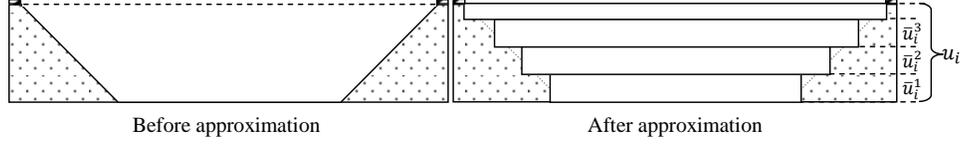


Figure 3.3: Approximation of side-slopes (for a cut).

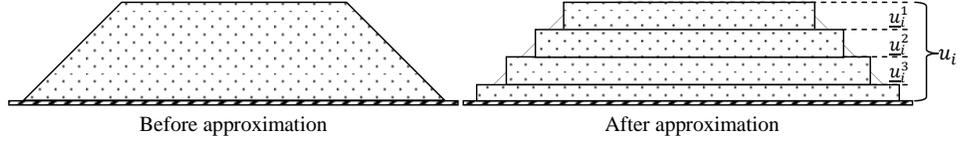


Figure 3.4: Approximation of side-slopes (for a fill).

Approximation of side-slopes

For typical roads with side-slopes, we propose to approximate the trapezoid shaped cross-sections with k rectangles. We call the height of such a rectangle the slab-height. Figures 3.3 and 3.4 show the approximation of side-slopes for a cut and a fill respectively.

The cross-section of a cut section i is approximated by k_i^+ rectangles having slab-height $\bar{u}_i^1, \bar{u}_i^2, \dots, \bar{u}_i^{k_i^+}$ and width $w_i^1, w_i^2, \dots, w_i^{k_i^+}$ where

$$0 < w_i^1 < w_i^2 < \dots < w_i^{k_i^+}. \quad (3.13b)$$

So, for the cut section i with section length of L , the volume of these rectangular bars are $A_i^1 \bar{u}_i^1, A_i^2 \bar{u}_i^2, \dots, A_i^{k_i^+} \bar{u}_i^{k_i^+}$ with

$$0 < A_i^1 < A_i^2 < \dots < A_i^{k_i^+} \quad (3.13c)$$

where $A_i^k = w_i^k \times L$ for all $k \in \{1, 2, \dots, k_i^+\}$.

Similarly, if the side slope of a fill section i with section length of L is approximated by k_i^- rectangular bars having slab-height $\underline{u}_i^1, \underline{u}_i^2, \dots, \underline{u}_i^{k_i^-}$ and area $B_i^1, B_i^2, \dots, B_i^{k_i^-}$ with

$$0 < B_i^1 < B_i^2 < \dots < B_i^{k_i^-} \quad (3.13d)$$

3.2. Model description

then the volume of these bars are $B_i^1 u_i^1, B_i^2 u_i^2, \dots, B_i^{k_i^-} u_i^{k_i^-}$. It should be noted that u_i^k are negative numbers for all $k \in \{1, 2, \dots, k_i^-\}$, and Equations (3.13c) and (3.13d) are the assumptions of the model.

For all $i \in \mathcal{S}$, $k_a \in \{1, 2, \dots, k_i^+\}$, and $k_b \in \{1, 2, \dots, k_i^-\}$, the cut and fill area lists, $A_i^{k_a}$ and $B_i^{k_b}$, are given to the model as a table, which is called a lookup table. So, a lookup table has k_i^+ entries for cut area information and k_i^- entries for fill area information.

First, we will develop the side-slope equations for the cut sections and then extend the same idea for the fill sections. For a cut section i , $V_i^- = 0$, so the volume constraint (3.13a) becomes

$$V_i^+ = A_i u_i. \quad (3.13e)$$

If the cut section is approximated by k_i^+ entries from the lookup table then the volume constraint considering side-slopes can be written as

$$V_i^+ = A_i^1 u_i^1 + A_i^2 u_i^2 + \dots + A_i^{k_i^+} u_i^{k_i^+} \quad (3.13f)$$

where $0 \leq u_i^1 \leq \bar{u}_i^1, 0 \leq u_i^2 \leq \bar{u}_i^2, \dots, 0 \leq u_i^{k_i^+} \leq \bar{u}_i^{k_i^+}$, and $u_i = u_i^1 + u_i^2 + \dots + u_i^{k_i^+}$.

Proposition 3.3. *For any optimal solution to the MILP, if i is a cut section and $\bar{u}_i^1 + \bar{u}_i^2 + \dots + \bar{u}_i^k \leq u_i \leq \bar{u}_i^1 + \bar{u}_i^2 + \dots + \bar{u}_i^{k+1}$ then $u_i^1 = \bar{u}_i^1, u_i^2 = \bar{u}_i^2, \dots, u_i^k = \bar{u}_i^k, u_i^{k+1} = u_i - (\bar{u}_i^1 + \bar{u}_i^2 + \dots + \bar{u}_i^k), u_i^{k+2} = 0, u_i^{k+3} = 0, \dots, u_i^{k_i^+} = 0$.*

Proof. Since $V_i^+ = A_i^1 u_i^1 + A_i^2 u_i^2 + \dots + A_i^{k_i^+} u_i^{k_i^+}$ is being minimized in the objective function and $0 < A_i^1 < A_i^2 < \dots < A_i^{k_i^+}$, u_i^1 must be equal to \bar{u}_i^1 if u_i^2 has a value, otherwise there is another solution with a lower objective value. With similar logic, $u_i^2, u_i^3, \dots, u_i^k$ must go to their upper bounds, u_i^{k+1} must equal to the remaining height, and rest must be 0. \square

Remark 3.4. The above implementation of side slopes introduces $k_i^+ + k_i^-$ more continuous variables for each section i .

Lemma 3.5. *Given $0 < A_1 < A_2$, consider*

3.2. Model description

$$\begin{aligned}
 \text{(LP1)} \quad & \min_{u_1, u_2} V, \\
 \text{s.t.} \quad & V = A_1 u_1 + A_2 u_2, \\
 & u = u_1 + u_2, \\
 & 0 \leq u_1 \leq \bar{u}_1, \\
 & 0 \leq u_2 \leq \bar{u}_2,
 \end{aligned}$$

$$\begin{aligned}
 \text{(LP2)} \quad & \min_u V, \\
 \text{s.t.} \quad & V \geq A_1 u, \\
 & V \geq A_1 \bar{u}_1 + A_2(u - \bar{u}_1),
 \end{aligned}$$

Then (LP2) will give the same result as (LP1).

Proof. (LP2) can be written as,

$$\begin{aligned}
 \min \quad & V, \\
 \text{s.t.} \quad & V \geq \max\{A_1 u, A_1 \bar{u}_1 + A_2(u - \bar{u}_1)\}, \\
 & 0 < A_1 < A_2.
 \end{aligned}$$

Since it is a minimization problem, the optimized output would be

$$V_{\text{opt}} = \max\{A_1 u, A_1 \bar{u}_1 + A_2(u - \bar{u}_1)\}.$$

Case I ($u \in [0, \bar{u}_1]$): For (LP1), since $u \in [0, \bar{u}_1]$ by Proposition 3.3 we have $u_1 = u$ and $u_2 = 0$. So,

$$V_{\text{opt}} = A_1 u.$$

For (LP2), since $A_1 < A_2$ and $u \in [0, \bar{u}_1]$ we have,

$$\begin{aligned}
 V_{\text{opt}} &= \max\{A_1 u, A_1 \bar{u}_1 + A_2(u - \bar{u}_1)\}, \\
 &= A_1 u.
 \end{aligned}$$

Case II ($u \in [\bar{u}_1, \bar{u}_1 + \bar{u}_2]$): For (LP1), since $u \in [\bar{u}_1, \bar{u}_1 + \bar{u}_2]$ by Proposition 3.3 we have $u_1 = \bar{u}_1$ and $u_2 = u - \bar{u}_1$. So,

$$V_{\text{opt}} = A_1 \bar{u}_1 + A_2(u - \bar{u}_1).$$

For (LP2), since $A_1 < A_2$ and $u \in [\bar{u}_1, \bar{u}_1 + \bar{u}_2]$ we have

$$\begin{aligned}
 V_{\text{opt}} &= \max\{A_1 u, A_1 \bar{u}_1 + A_2(u - \bar{u}_1)\}, \\
 &= A_1 \bar{u}_1 + A_2(u - \bar{u}_1).
 \end{aligned}$$

□

3.2. Model description

Theorem 3.6. *Given $0 < A_1 < A_2 < \dots < A_n$, the following LP*

$$\begin{aligned}
 \min \quad & V \\
 \text{s.t.} \quad & V \geq A_1 h \\
 & V \geq A_1 \bar{u}_1 + A_2 (h - \bar{u}_1) \\
 & V \geq A_1 \bar{u}_1 + A_2 \bar{u}_2 + A_3 (h - \bar{u}_1 - \bar{u}_2) \\
 & V \geq A_1 \bar{u}_1 + A_2 \bar{u}_2 + A_3 \bar{u}_3 + A_4 (h - \bar{u}_1 - \bar{u}_2 - \bar{u}_3) \\
 & \cdot \\
 & \cdot \\
 & \cdot \\
 & V \geq A_1 \bar{u}_1 + A_2 \bar{u}_2 + \dots + A_{n-1} \bar{u}_{n-1} + A_n (h - \bar{u}_1 - \bar{u}_2 - \dots - \bar{u}_{n-1})
 \end{aligned}$$

is equivalent to

$$\begin{aligned}
 \min_{u_1, u_2, \dots, u_n} \quad & V, \\
 \text{s.t.} \quad & V = A_1 u_1 + A_2 u_2 + \dots + A_n u_n, \\
 & u = u_1 + u_2 + \dots + u_n, \\
 & 0 \leq u_1 \leq \bar{u}_1, \\
 & 0 \leq u_2 \leq \bar{u}_2, \\
 & \cdot \\
 & \cdot \\
 & \cdot \\
 & 0 \leq u_n \leq \bar{u}_n.
 \end{aligned}$$

Proof. If $n = 2$, then Lemma 3.5 completes the proof. Similar logic can be applied by using Proposition 3.3. \square

Remark 3.7. Implementation of side slopes by Theorem 3.6 reduces the number of continuous variables for side-slopes from $k_i^+ + k_i^-$ to zero for each section i .

Lemma 3.8. *If no roadside stockpiling occurs then the volume constraints (3.13a) are equivalent to*

$$V_i^+ \geq A_i u_i \text{ for all } i \in \mathcal{S}, \quad (3.13g)$$

$$V_i^- \geq -A_i u_i \text{ for all } i \in \mathcal{S}. \quad (3.13h)$$

3.2. Model description

Proof. If i is a cut, $V_i^- = 0$ when no roadside stockpiling occurs, which also satisfies Equation (3.13h) since its right hand side is a negative number, and $V_i^+ = A_i u_i$ since the objective function minimizes V_i^+ and $V_i^+ \geq A_i u_i$ by Equation (3.13g). So, $V_i^+ - V_i^- = A_i u_i - 0 = A_i u_i$. If i is a fill, $V_i^+ = 0$ when no roadside stockpiling occurs, which also satisfies Equation (3.13g) since its right hand side is a negative number, and $V_i^- = -A_i u_i$ since the objective function minimizes V_i^- and $V_i^- \geq -A_i u_i$ by Equation (3.13h). So, $V_i^+ - V_i^- = 0 - (-A_i u_i) = A_i u_i$. If i is neither cut nor fill, $V_i^+ = 0$ and $V_i^- = 0$ when no roadside stockpiling occurs, which also satisfy Equations (3.13g) and (3.13h) since their right hand sides are zero. So, $V_i^+ - V_i^- = 0 = A_i u_i$ since $u_i = 0$. \square

By Lemma 3.8 and Corollary 3.6, we replace the volume constraints (3.13a) by the following constraints to incorporate side slope, given $0 < A_i^1 < A_i^2 < \dots < A_i^{k_i^+}$ and $0 < B_i^1 < B_i^2 < \dots < B_i^{k_i^-}$, for all $i \in \mathcal{S}$,

$$V_i^+ \geq A_i^1 u_i \quad (3.13i)$$

$$V_i^+ \geq A_i^1 \bar{u}_i^1 + A_i^2 (u_i - \bar{u}_i^1) \quad (3.13j)$$

$$V_i^+ \geq A_i^1 \bar{u}_i^1 + A_i^2 \bar{u}_i^2 + A_i^3 (u_i - \bar{u}_i^1 - \bar{u}_i^2) \quad (3.13k)$$

.

$$V_i^+ \geq A_i^1 \bar{u}_i^1 + A_i^2 \bar{u}_i^2 + \dots + A_i^{k_i^+} (u_i - \bar{u}_i^1 - \dots - \bar{u}_i^{k_i^+ - 1}) \quad (3.13l)$$

$$V_i^- \geq -B_i^1 u_i \quad (3.13m)$$

$$V_i^- \geq -B_i^1 \underline{u}_i^1 - B_i^2 (u_i - \underline{u}_i^1) \quad (3.13n)$$

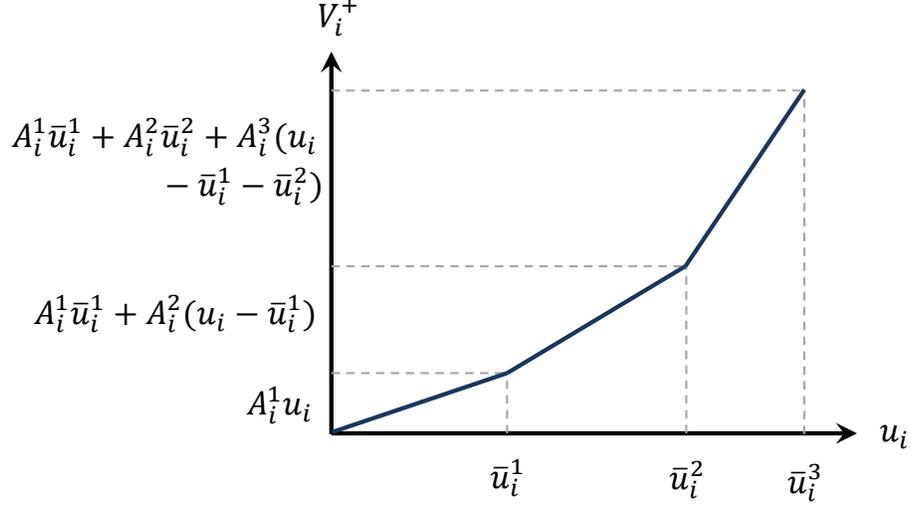
$$V_i^- \geq -B_i^1 \underline{u}_i^1 - B_i^2 \underline{u}_i^2 - B_i^3 (u_i - \underline{u}_i^1 - \underline{u}_i^2) \quad (3.13o)$$

.

$$V_i^- \geq -B_i^1 \underline{u}_i^1 - B_i^2 \underline{u}_i^2 - \dots - B_i^{k_i^-} (u_i - \underline{u}_i^1 - \dots - \underline{u}_i^{k_i^- - 1}). \quad (3.13p)$$

The compact form of these equations are, for $k_a \in \{1, 2, \dots, k_i^+\}$

$$V_i^+ \geq \sum_{k=1}^{k_a-1} A_i^k \bar{u}_i^k + A_i^{k_a} \left(u_i - \sum_{k=1}^{k_a-1} \bar{u}_i^k \right), \quad (3.13q)$$


 Figure 3.5: Convexity of V_i^+ when $0 < A_i^1 < A_i^2 < A_i^3$.

and for $k_b \in \{1, 2, \dots, k_i^-\}$

$$V_i^- \geq - \sum_{k=1}^{k_b-1} B_i^k \underline{u}_i^k - B_i^{k_b} \left(u_i - \sum_{k=1}^{k_b-1} \underline{u}_i^k \right). \quad (3.13r)$$

Remark 3.9. Both V_i^+ and V_i^- are represented as a piecewise linear function of u_i . Modeling a piecewise linear function requires binary variables to indicate the piece to be used for calculation, unless the function is convex. The assumptions $0 < A_i^1 < A_i^2 < \dots < A_i^{k_i^+}$ and $0 < B_i^1 < B_i^2 < \dots < B_i^{k_i^-}$ make V_i^+ and V_i^- piecewise linear convex functions (see Figure 3.5 where A_i^1 , A_i^2 , and A_i^3 are the slopes of the linear pieces). For non-convex data, approximation is required to convexify. Our numerical experiments show that less than 1% entries of the lookup table are non-convex for most of the problems of our test set.

3.2.6 Roadside stockpiling constraints

Roadside stockpiling constraints ensure that no roadside stockpiling occurs. In that case, for any section $i \in \mathcal{S}$ at least one of V_i^+ and V_i^- must be zero, which holds for the spline LP model of vertical alignment as shown in the previous chapter. But it may not hold for this model, since both V_i^+ and V_i^- of section i depend

3.2. Model description

on the earth movement variables and sometimes the earth movement is restricted by the presence of blocks. We call such sections block dependent sections.

We introduce additional binary variables b_i for all $i \in \mathcal{S}$ to ensure at least one of V_i^+ and V_i^- must be zero. If a section i is a cut then $b_i = 1$ otherwise $b_i = 0$. So, the roadside stockpiling constraints can be written as

$$V_i^+ \leq Mb_i, \quad (3.14a)$$

$$V_i^+ \geq \delta_i^+ b_i, \quad (3.14b)$$

$$V_i^- \leq M(1 - b_i), \quad (3.14c)$$

where M is a big number, and δ_i^+ is the smallest possible cut volume for a section i .

An alternate representation of Equations (3.14), which ensure $b_i = 1$ for a cut, $b_i = 0$ for a fill, and a section must be either a cut or fill with at least a small δ volume, can be the following

$$V_i^+ \leq Mb_i, \quad (3.15a)$$

$$V_i^+ \geq \delta b_i, \quad (3.15b)$$

$$V_i^- \leq M(1 - b_i), \quad (3.15c)$$

$$V_i^- \geq \delta(1 - b_i). \quad (3.15d)$$

We show that even with the additional n binary variables the performance of the model will not decrease significantly.

Lemma 3.10. *If i is not a block dependent section, $0 \leq b_i \leq 1$ is continuous (instead of binary), and the roadside stockpiling constraints are represented by Equation (3.15), then b_i will either be 0 or 1 in the optimal solution.*

Proof. If i is not a block dependent section, then either V_i^+ or V_i^- or both will be zero in the optimal solution otherwise we can prove there is another solution with a lower objective value. If V_i^+ is zero, then b_i must be zero by Equation (3.15b) since $\delta > 0$. If V_i^- is zero, then b_i must be 1 by Equation (3.15d) since $\delta > 0$. So, b_i will never take a fractional value. \square

Corollary 3.11. *If i is not a block dependent section, $b_i \in \{0, 1\}$, and the roadside stockpiling constraints are represented by Equation (3.15), then the MILP algorithm will never branch on any b_i .*

Proof. The branch and bound algorithm for solving MILP make branching decision based on LP relaxation of the MILP and the algorithm will branch on only those binary variables which has fractional value on the LP relaxation. By Lemma 3.10, the branch and bound algorithm will never branch on b_i . The MILP algorithm will run as if the binary variable b_i never existed. \square

3.2. Model description

Theorem 3.12. *If the roadside stockpiling constraints are represented by Equation (3.15), then the MILP algorithm may only branch on b_i if i is a block dependent section.*

Proof. By Corollary 3.11 this theorem holds. □

Lemma 3.13. *If i is not a block dependent section, $0 \leq b_i \leq 1$ is continuous (instead of binary), and the roadside stockpiling constraints are represented by Equation (3.14), then b_i will be 0 when $V_i^+ = 0$ and can take any value within the range $0 < b_i \leq 1$ when $V_i^+ \neq 0$ in the optimal solution.*

Proof. If i is not a block dependent section, then either V_i^+ or V_i^- will be zero in the optimal solution otherwise we can prove there is another solution with a lower objective value. If V_i^+ is zero, then b_i must be zero by Equation (3.14b) since $\delta > 0$. If $V_i^+ \neq 0$, then b_i cannot be zero by Equation (3.14b) since $\delta > 0$. So, b_i may take a fractional value if i is a cut. □

Corollary 3.14. *If i is not a block dependent section, $b_i \in \{0, 1\}$, and the roadside stockpiling constraints are represented by Equation (3.14), then the MILP algorithm may branch on b_i when i is a cut.*

Proof. By Lemma 3.13, the branch and bound algorithm may branch on b_i when i is a cut. □

Theorem 3.15. *If the roadside stockpiling constraints are represented by Equation (3.14), then the MILP algorithm may only branch on b_i if i is a block dependent section or a cut.*

Proof. By Corollary 3.14 this theorem holds. □

Remark 3.16. If the roadside stockpiling constraints are represented by Equation (3.15), from Theorem 3.12, we see that if a section i is not a block dependent section the model will be solved as if b_i never existed, and if the roadside stockpiling constraints are represented by Equation (3.14), from Theorem 3.15, we see that if a section i is not a block dependent section and a fill the model will be solved as if b_i never existed.

Theorem 3.17. *If \mathcal{S}_0 is the set of sections that are neither cuts nor fills, then the solution time of a model with Equation (3.15) will be less than or equal to that of a model with Equation (3.15) with error value $\sum_{i \in \mathcal{S}_0} \delta \times \min\{p_i, q_i\}$.*

3.2. Model description

Proof. The more binary variables the branch and bound algorithm branches on, the longer it takes to solve. By Theorems 3.12 and 3.15, the roadside stockpiling constraints represented by Equation (3.15) can always be solved as fast as or faster than the same model with roadside stockpiling constraints represented by Equation (3.14). If Equations (3.14) are used as the roadside stockpiling constraints then both V_i^+ and V_i^- cannot be zero at the same time. In the situation where both V_i^+ and V_i^- should be zero, the optimizer will choose $V_i^+ = \delta$ if $p_i > q_i$ and $V_i^- = \delta$ if $p_i < q_i$ and thus will introduce $\delta \times \min\{p_i, q_i\}$ error in the objective function. \square

Remark 3.18. If $S_0 = \emptyset$, then Equations (3.14) and (3.15) are logically equivalent.

Remark. The option of choosing between Equations (3.14) and (3.15) will be exposed to the end-users who can decide whether they want a faster solution or a more accurate solution.

3.2.7 Gap constraints

The gap constraints ensure that the area gap between the ground profile and the road profile for any section i , must be equal to the area created by section length and the height difference between the ground profile and the road profile. For all $g \in \mathcal{G}, i \in S_g$ the gap constraints can be written as

$$h_{g,i} - a_{g,3} \frac{(s_{g,i+1})^2 + s_{g,i+1}s_{g,i} + (s_{g,i})^2}{3} - a_{g,2} \frac{s_{g,i+1} + s_{g,i}}{2} - a_{g,1} = u_{\neq(g,i)}. \quad (3.16)$$

3.2.8 Slope constraints

The slope constraints (3.17) implies that the slope of the spline segments must not go beyond a maximum value U or a minimum value L , and since the spline segments are quadratic checking the slope bounds at the beginning and end point of the spline segments is sufficient. For all $g \in \mathcal{G}, i \in \{1, n_g\}$ the slope constraints can be written as

$$L \leq a_{g,2} + 2a_{g,3}s_{g,i} \leq U. \quad (3.17)$$

3.2.9 Smoothness constraints

The C^0 continuity constraints force the spline segments to be continuous at their joining points. So, for all $g \in \mathcal{G} \setminus \{1\}$

$$a_{g,1} + a_{g,2}s_{g,1} + a_{g,3}s_{g,1}^2 = a_{(g-1),1} + a_{(g-1),2}s_{g,1} + a_{(g-1),3}s_{g,1}^2, \quad (3.18a)$$

3.2. Model description

and the C^1 continuity constraints force the spline segments to be differentiable or smooth at their joining points i.e., for all $g \in \mathcal{G} \setminus \{1\}$

$$a_{g,2} + 2a_{g,3}s_{g,1} = a_{(g-1),2} + 2a_{(g-1),3}s_{g,1}. \quad (3.18b)$$

3.2.10 Fixed point constraints

The fixed point constraints fix the starting and ending elevations,

$$a_{1,1} + a_{1,2}s_{1,1} + a_{1,3}s_{1,1}^2 = H_A, \quad (3.19a)$$

$$a_{m,1} + a_{m,2}s_{m,n_m} + a_{m,3}s_{m,n_m}^2 = H_B, \quad (3.19b)$$

the elevations of the fixed points, for all $H_{g,i} \in \mathcal{H}$

$$a_{g,1} + a_{g,2}s_{g,i} + a_{g,3}s_{g,i}^2 = H_{g,i}, \quad (3.19c)$$

as well as the starting and ending slopes,

$$a_{1,2} + 2a_{1,3}s_{1,1} = H'_A, \quad (3.19d)$$

$$a_{m,2} + 2a_{m,3}s_{m,n_m} = H'_B. \quad (3.19e)$$

It should be noted that Equations (3.19d) and (3.19e) are optional constraints for most of the problems.

3.2.11 Bounds

The bounds of the decision variables are as follows, for all $(i, j) \in \mathcal{N}^2, t \in \mathcal{T}$,

$$x_{ijt} \geq 0, \quad (3.20a)$$

for all $k \in \mathcal{I}, t \in \mathcal{T}$,

$$y_{kt} \in \{0, 1\}, \quad (3.20b)$$

for all $i \in \mathcal{S}$,

$$\underline{u}_i \leq u_i \leq \bar{u}_i, \quad (3.20c)$$

$$0 \leq V_i^+ \leq M_i^+, \quad (3.20d)$$

$$0 \leq V_i^- \leq M_i^-, \quad (3.20e)$$

for all $g \in \mathcal{G}, k \in \{1, 2, 3\}$,

$$a_{g,k} \in \mathbb{R}. \quad (3.20f)$$

3.3 Model summary

The summary of the MILP model for the vertical alignment design considering blocks and side-slopes is stated below.

3.3.1 The objective function

$$\begin{aligned} \min \sum_{i \in \mathcal{S}} (p_i V_i^+ + q_i V_i^-) &+ \sum_{i \in \mathcal{B}} \sum_{j \in \mathcal{N}_{\rightarrow}^i} \sum_{t \in \mathcal{T}} p_i x_{ijt} \\ &+ \sum_{i \in \mathcal{W}} \sum_{j \in \mathcal{N}_{\leftarrow}^i} \sum_{t \in \mathcal{T}} q_i x_{jit} + \sum_{(i,j) \in \mathcal{N}^2} \sum_{t \in \mathcal{T}} c_{ij} x_{ijt}. \end{aligned} \quad (3.8)$$

3.3.2 Balance constraints

For all $i \in \mathcal{S}$,

$$\sum_{t \in \mathcal{T}} \sum_{j \in \mathcal{N}_{\rightarrow}^i} x_{ijt} = V_i^+, \quad (3.9a)$$

and for all $i \in \mathcal{S}$,

$$\sum_{t \in \mathcal{T}} \sum_{j \in \mathcal{N}_{\leftarrow}^i} x_{jit} = V_i^-. \quad (3.9b)$$

3.3.3 Pit constraints

For all $i \in \mathcal{B}$,

$$\sum_{t \in \mathcal{T}} \sum_{j \in \mathcal{N}_{\rightarrow}^i} x_{ijt} \leq C_i^b, \quad (3.10)$$

and for all $i \in \mathcal{W}$,

$$\sum_{t \in \mathcal{T}} \sum_{j \in \mathcal{N}_{\leftarrow}^i} x_{jit} \leq C_i^w. \quad (3.11)$$

3.3.4 Block constraints

For all $t \in \mathcal{T} \setminus \{1\}$, $k \in \mathcal{I}$, and $(i, j) \in \mathcal{N}_b^k$,

$$x_{ijt} \leq My_{k,t-1}, \quad (3.12a)$$

3.3. Model summary

for all $t \in \mathcal{T} \setminus \{1\}$, $(k_1, k_2) \in \mathcal{I}^2$, and $i, j \in \mathcal{N}_b^{k_1, k_2}$,

$$x_{ijt} \leq M y_{k_1, t-1} + M y_{k_2, t-1}, \quad (3.12b)$$

for all $t \in \mathcal{T} \setminus \{1\}$, $k \in \bar{\mathcal{I}}_{\leftarrow}$, and $i, j \in \mathcal{N}_b^{\leftarrow, k}$,

$$x_{ijt} \leq M y_{k, t-1}, \quad (3.12c)$$

for all $t \in \mathcal{T} \setminus \{1\}$, $k \in \bar{\mathcal{I}}_{\rightarrow}$, and $i, j \in \mathcal{N}_b^{k, \rightarrow}$,

$$x_{ijt} \leq M y_{k, t-1}, \quad (3.12d)$$

for all $k \in \mathcal{I}$, $u \in \mathcal{T}$,

$$W_{ku}^+ \leq M_{\zeta(k)}^+ y_{ku}, \quad (3.12i)$$

$$W_{ku}^+ \leq V_{\zeta(k)}^+, \quad (3.12j)$$

$$W_{ku}^+ \geq V_{\zeta(k)}^+ - M_{\zeta(k)}^+ (1 - y_{ku}), \quad (3.12k)$$

$$W_{ku}^- \leq M_{\zeta(k)}^- y_{ku}, \quad (3.12m)$$

$$W_{ku}^- \leq V_{\zeta(k)}^-, \quad (3.12n)$$

$$W_{ku}^- \geq V_{\zeta(k)}^- - M_{\zeta(k)}^- (1 - y_{ku}), \quad (3.12o)$$

$$\sum_{t=0}^u \sum_{j \in \mathcal{N}_{\rightarrow}^{\zeta(k)}} x_{\zeta(k)jt} \geq W_{ku}^+, \quad (3.12q)$$

$$\sum_{t=0}^u \sum_{j \in \mathcal{N}_{\leftarrow}^{\zeta(k)}} x_{j\zeta(k)t} \geq W_{ku}^-, \quad (3.12r)$$

for all $t \in \mathcal{T} \setminus \{n_z + 1\}$,

$$\sum_{k \in \mathcal{I}} y_{kt} \geq t, \quad (3.12s)$$

and for all $k \in \mathcal{I}$, $t \in \mathcal{T} \setminus \{n_z + 1\}$,

$$(y_{k, (t+1)} \geq y_{k, t}). \quad (3.12t)$$

3.3.5 Volume constraints

Given $0 < A_i^1 < A_i^2 < \dots < A_i^{k_i^+}$ and $0 < B_i^1 < B_i^2 < \dots < B_i^{k_i^-}$, for $k_a \in \{1, 2, \dots, k_i^+\}$

$$V_i^+ \geq \sum_{k=1}^{k_a-1} A_i^k \bar{u}_i^k + A_i^{k_a} \left(u_i - \sum_{k=1}^{k_a-1} \bar{u}_i^k \right), \quad (3.13q)$$

3.3. Model summary

and for $k_b \in \{1, 2, \dots, k_i^-\}$

$$V_i^- \geq - \sum_{k=1}^{k_b-1} B_i^k \underline{u}_i^k - B_i^{k_b} \left(u_i - \sum_{k=1}^{k_b-1} \underline{u}_i^k \right). \quad (3.13r)$$

3.3.6 Roadside stockpiling constraints

If the end-user prefers accuracy over speed, then for all $i \in \mathcal{S}$,

$$V_i^+ \leq Mb_i, \quad (3.14a)$$

$$V_i^+ \geq \delta_i^+ b_i, \quad (3.14b)$$

$$V_i^- \leq M(1 - b_i), \quad (3.14c)$$

otherwise, for all $i \in \mathcal{S}$,

$$V_i^+ \leq Mb_i, \quad (3.15a)$$

$$V_i^+ \geq \delta b_i, \quad (3.15b)$$

$$V_i^- \leq M(1 - b_i), \quad (3.15c)$$

$$V_i^- \geq \delta(1 - b_i). \quad (3.15d)$$

3.3.7 Gap constraints

For all $g \in \mathcal{G}, i \in \mathcal{S}_g$,

$$h_{g,i} - a_{g,3} \frac{(s_{g,i+1})^2 + s_{g,i+1}s_{g,i} + (s_{g,i})^2}{3} - a_{g,2} \frac{s_{g,i+1} + s_{g,i}}{2} - a_{g,1} = u_{\neq(g,i)}. \quad (3.16)$$

3.3.8 Slope constraints

For all $g \in \mathcal{G}, i \in \{1, n_g\}$,

$$L \leq a_{g,2} + 2a_{g,3}s_{g,i} \leq U. \quad (3.17)$$

3.3.9 Smoothness constraints

For all $g \in \mathcal{G} \setminus \{1\}$,

$$a_{g,1} + a_{g,2}s_{g,1} + a_{g,3}s_{g,1}^2 = a_{(g-1),1} + a_{(g-1),2}s_{g,1} + a_{(g-1),3}s_{g,1}^2, \quad (3.18a)$$

$$a_{g,2} + 2a_{g,3}s_{g,1} = a_{(g-1),2} + 2a_{(g-1),3}s_{g,1}. \quad (3.18b)$$

3.3.10 Fixed point constraints

$$a_{1,1} + a_{1,2}s_{1,1} + a_{1,3}s_{1,1}^2 = H_A, \quad (3.19a)$$

$$a_{m,1} + a_{m,2}s_{m,n_m} + a_{m,3}s_{m,n_m}^2 = H_B, \quad (3.19b)$$

for all $H_{g,i} \in \mathcal{H}$

$$a_{g,1} + a_{g,2}s_{g,i} + a_{g,3}s_{g,i}^2 = H_{g,i}, \quad (3.19c)$$

and

$$\left(a_{1,2} + 2a_{1,3}s_{1,1} = H'_A \right), \quad (3.19d)$$

$$\left(a_{m,2} + 2a_{m,3}s_{m,n_m} = H'_B \right). \quad (3.19e)$$

3.3.11 Bounds

For all $(i, j) \in \mathcal{N}^2, t \in \mathcal{T}$,

$$x_{ijt} \geq 0, \quad (3.20a)$$

for all $k \in \mathcal{I}, t \in \mathcal{T}$,

$$y_{kt} \in \{0, 1\}, \quad (3.20b)$$

$$W_{kt}^+ \geq 0, \quad (3.12l)$$

$$W_{kt}^- \geq 0, \quad (3.12p)$$

for all $i \in \mathcal{S}$,

$$\underline{u}_i \leq u_i \leq \bar{u}_i, \quad (3.20c)$$

$$0 \leq V_i^+ \leq M_i^+, \quad (3.20d)$$

$$0 \leq V_i^- \leq M_i^-, \quad (3.20e)$$

and for all $g \in \mathcal{G}, k \in \{1, 2, 3\}$,

$$a_{g,k} \in \mathbb{R}. \quad (3.20f)$$

3.4 Model evaluation

Proposition 3.19. *The MILP of Section 3.3 has $\mathcal{O}(n^2n_z)$ continuous variables.*

3.5. Algorithms

Proof. The MILP has n variables for excavation volume, n variables for embankment volume, $(n^2 + n_b n + n_w n)(n_z + 1)$ variables for hauling volume, $2n_z(n_z + 1)$ number of W_{ku}^+ and W_{ku}^- variables, $3m$ variables for the spline coefficients, and n variables for section height. So, the total number of continuous variables is

$$\begin{aligned}
 & n + n + (n^2 + n_b n + n_w n)(n_z + 1) + 2n_z(n_z + 1) + 3m + n \\
 = & 3n + 3m + (n^2 + n_b n + n_w n + 2n_z)(n_z + 1) \\
 \leq & 6n + (n^2 + n^2 + n^2 + 2n_z)(n_z + n_z) \\
 = & 6n + 6n^2 n_z + 4n_z^2
 \end{aligned}$$

since $n \geq n_b$, $n \geq n_w$, $n \geq m$, and $n \geq n_z$ and assuming $n_z \geq 1$. □

Proposition 3.20. *The MILP of Section 3.3 has $\mathcal{O}(n_z^2 + n)$ binary variables.*

Proof. The total number of binary variables is $n_z(n_z + 1) + n \leq 2n_z^2 + n$ assuming $n_z \geq 1$. □

Proposition 3.21. *The MILP of Section 3.3 has $\mathcal{O}(n^2 n_z^3)$ constraints.*

Proof. The number of constraints created by different equations of this model are shown in Table 3.1. The total number of constraints is $\mathcal{O}(n^2 n_z^3)$. □

3.5 Algorithms

In this section, we recall some algorithms to speed up the solution of the model. These algorithms are studied in more detail in [HKL11, Koc10].

3.5.1 The direct solve (DS) algorithm

The direct solve algorithm is the most basic algorithm that reads the data and build the model depending on the solver specification. Then it solves the model and writes the solution. But some models can be ill conditioned. A linear program or a mixed integer program is called ill conditioned or numerically unstable when a small change in the input results in a big change in the output. Having very large or very small numbers as coefficients in the constraint matrix is a source of numerical instability. Constraints (3.16), (3.18a), and (3.19a) make our model numerically unstable because of the quadratic coefficients. A ill conditioned model can cause trouble in the pivot selection process of the simplex algorithm. So, for this sort of numerical instability we have to either scale the constraint matrix or make the solver use a more conservative pivoting scheme during the basis factorization.

3.5. Algorithms

Equation	Number of Constraints (worst case)
(3.9a),(3.9b)	$2n$
(3.10),(2.26e)	$n_b + n_w$
(3.12a)	$\frac{1}{4}n^2(n_z + 1)n_z$
(3.12b)	$(n - 1)^2(n_z + 1)\frac{n_z(n_z - 1)}{2}$
(3.12c), (3.12d)	$2(n - 1)^2(n_z + 1)n_z$
From (3.12i) to (3.12r)	$2n_z(n - 1)(n_z + 1)$
(3.12s)	n_z
(3.12t)	n_z^2
(3.13q), (3.13r)	$\sum_{i \in \mathcal{S}} (k_i^+ + k_i^-)$
From (3.14a) to (3.14c)	$3n$
From (3.14a) to (3.14c)	$4n$
(3.16)	n
(3.17)	$2m$
(3.18a), (3.18a)	$2(m - 1)$
From (3.19a) to (3.19e)	$4 + \mathcal{H} $

Table 3.1: Number of constraints created by different equations of the basic MILP model

Most commercial solvers have built-in functionality to handle ill conditioned models. For example, CPLEX has a scale factor that can be used for aggressive scaling and another parameter named Markowitz tolerance for a more conservative pivoting scheme. A function can be written to scale the constraint matrix if the solver does not have this functionality. Our preliminary result shows that if the maximum station number gets bigger than 1000 then the model starts to become ill conditioned.

Algorithm 3.1 DS

Step 0. **READ** the data and **BUILD** the model

Step 1. $\left\{ \begin{array}{l} \mathbf{IF} \text{ the model is an ill conditioned model} \\ \quad \mathbf{CALL} \text{ the ill conditioned model handling function} \\ \mathbf{ENDIF} \end{array} \right.$

Step 2. **SOLVE** the model

Step 3. **WRITE** the solution and **RETURN** the status of the solution

Algorithm 3.2 DSS

- Step 0. **READ** the data and **BUILD** the model
- Step 1. { **IF** the model is an ill conditioned model
 CALL the ill conditioned model handling function
 ENDIF
- Step 2. { **SELECT** the left most access road
 SET y_{kt} such that it removes
 the block closest to the access road at time-step 1
 the block second closest to the access road at time-step 2
 ...
 the block farthest to the access road at time-step n_b
 SOLVE the model with the added constraints
- Step 3. **USE** the solution as a starting point for the original model
- Step 4. **SOLVE** the model
- Step 5. **WRITE** the solution and **RETURN** the status of the solution

3.5.2 The direct solve with starting point (DSS) algorithm

The direct solve with starting point algorithm starts with a feasible starting point. While generating a very good feasible starting point is still an active research area, we use a heuristic to generate a reasonable feasible starting point. The left most access road is chosen, then variables y_{kt} are set in such a way that the block closest to the access road is removed first, then the next closest is removed and so on. With these values of y_{kt} the model is solved and the solution is used as a feasible starting point.

3.5.3 The maximal movement distance ($Md_{\max}\alpha$) algorithm

Heuristic 3.22. *It is better to move earth to a closer section than a farther section.*

The idea behind this algorithm is Heuristic 3.22. The algorithm starts with an initial d_{\max} and sets the values of $x_{ijt} = 0$ for the section that are d_{\max} further from each other. Then it solves the model and continues to use the solution as a starting point of the next iteration until d_{\max} is large enough so no x_{ijt} is set to zero. At that iteration, we get the optimal solution.

Algorithm 3.3 $Md_{\max}\alpha$

Step 0. **INPUT** $d_{\max} \geq 2, \alpha > 1$, **READ** the data and **BUILD** the model

Step 1. { **IF** the model is an ill conditioned model
 CALL the ill conditioned model handling function
 ENDIF

Step 2. { **IF** there is a starting point
 ADD the starting point to the model
 ENDIF
 FOR EACH section i , section j , and time-step t
 IF $|i - j| > d_{\max}$
 SET $x_{ijt} = 0$
 ENDIF
 ENDFOR
 SOLVE the model with the added constraints

Step 3. { **IF** $d_{\max} > |\mathcal{S}|$
 WRITE the solution and **RETURN** the status of the solution
 ELSE
 SET d_{\max} as $\alpha \times d_{\max}$
 ENDIF
 GO TO Step 2

Algorithm 3.4 $MTd_{\max}\alpha$

Step 0. **INPUT** $d_{\max} \geq 2, \alpha > 1, \epsilon > 0$, **READ** the data and **BUILD** the model

Step 1. $\left\{ \begin{array}{l} \text{IF the model is an ill conditioned model} \\ \quad \text{CALL the ill conditioned model handling function} \\ \text{ENDIF} \end{array} \right.$

Step 2. **SOLVE** the LP relaxation and **SAVE** the objective value

Step 3. $\left\{ \begin{array}{l} \text{IF there is a starting point} \\ \quad \text{ADD the starting point to the model} \\ \text{ENDIF} \\ \text{FOR EACH section } i, \text{ section } j, \text{ and time-step } t \\ \quad \text{IF } |i - j| > d_{\max} \\ \quad \quad \text{SET } x_{ijt} = 0 \\ \quad \text{ENDIF} \\ \text{ENDFOR} \\ \text{SOLVE the model with the added constraints} \\ \text{CALCULATE the mipgap as } \left| \frac{\text{Step 3 Obj} - \text{Step 2 Obj}}{\text{Step 2 Obj}} \right| \times 100 \end{array} \right.$

Step 4. $\left\{ \begin{array}{l} \text{IF } d_{\max} > |S| \text{ OR mipgap} < \epsilon \\ \quad \text{WRITE the solution and RETURN the status of the solution} \\ \text{ELSE} \\ \quad \text{SET } d_{\max} \text{ as } \alpha \times d_{\max} \\ \text{ENDIF} \\ \text{GO TO Step 2} \end{array} \right.$

3.5.4 The maximal movement distance with mipgap tolerance ($MTd_{\max}\alpha$) algorithm

The maximal movement distance with mipgap tolerance algorithm is an improvement over the $Md_{\max}\alpha$ algorithm. The optimal solution may occur in any of the intermediate steps of the $Md_{\max}\alpha$ algorithm. Setting $x_{ijt} = 0$ gives the lower bound for the optimal solution of the original model and the LP relaxation gives the upper bound. Consequently, this algorithm also checks the mipgap and if it is within the tolerance then the algorithm terminates.

Chapter 4

A Quasi-network-flow Model for Vertical Alignment Considering Blocks and Side-slopes

Take what you need, act as you must,
and you will obtain that for which
you wish.

René Descartes (1596 - 1650)

In general, a mixed integer linear program is *NP*-hard. As a result, some MILPs require smart modeling techniques to solve them in a reasonable time-frame. Hare, Koch, and Lucet [HKL11] found that the $Md_{\max}\alpha$ and $MTd_{\max}\alpha$ algorithms solve the problems with large number of sections noticeably faster than other algorithms. Both of these algorithms are based on Heuristic 3.22, which suggests that it is more probable to have earth movements among nearby sections than among further sections in an optimal solution. So, these algorithms start with a small maximum distance for earth movement, and continue to increase the maximum distance at each iteration until the optimality conditions are satisfied. These algorithms can be very efficient, if a good value for the initial maximum distance (d_{\max}) and the maximum distance increase factor (α) can be calculated, which usually varies depending on the problem.

Heuristic 3.22 motivates us to seek an alternate modeling technique. The model introduced in this chapter can be visualized as a graph where the sections and pits are nodes and the arcs represent feasible moves. Additional virtual nodes called transit nodes are also introduced. The amount of earth moved through an arc is called flow. There are different types of flows in the model.

- Each section has a load and an unload flow representing fill and cut amount respectively. Each of the flows are further divided into two for east and west direction.

4.1. Definitions and model parameters

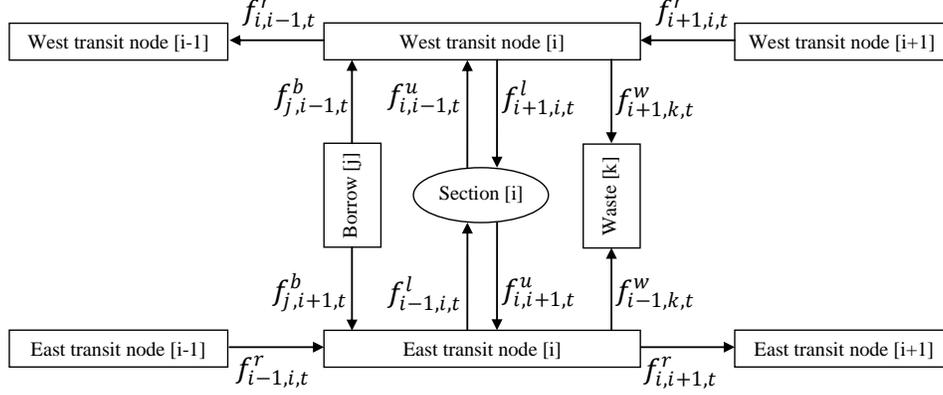


Figure 4.1: A typical section i at time-step t (where $\vartheta(j) = \varphi(k) = i$).

- Each borrow pit has an unload flow representing its cut amount for both directions. These flows are called borrow flows.
- Each waste pit has a load flow representing its fill amount for both directions. These flows are called waste flows.
- Each section has an east and a west transit node, each of which has a east transit flow and a west transit flow representing earth movement from one section to another.

All these flows are the decision variables of the model. A typical section is shown in Figure 4.1. This formulation reduces the earth movement variables from $\mathcal{O}(n^2n_z)$ to $\mathcal{O}(nn_z)$.

4.1 Definitions and model parameters

For this model, we use the definitions and model parameters from Section 3.1.

4.1.1 Cost components

The cost components are as follow.

- For a section i , the per unit volume excavation cost is p_i , and the per unit volume embankment cost is q_i . The per unit volume hauling cost from section i to section $i - 1$ (respectively $i + 1$) is represented as $c_{i,i-1}^r = cd_{i,i-1}$

4.1. Definitions and model parameters

(respectively $c_{i,i+1}^r = cd_{i,i+1}$), where c represents the cost of moving one unit volume of earth to one unit distance, and d_{ij} is the distance between section i and j .

- For a borrow pit i , the borrow cost for either direction is $c_i^b = p_i + c\tilde{d}_i$.
- For a waste pit i , the waste cost for either direction is $c_i^w = q_i + c\tilde{d}_i$.

It should be noted that the model requires the described structure of the cost function.

Remark 4.1. Borrow pits normally have higher quality materials than other sections. As a result the embankment cost with the materials from a borrow pit is lower than other embankment costs. But, after the borrow flows reach the transit nodes, the model cannot differentiate between borrow materials and other materials. To incorporate this in the model, the difference between general embankment cost and the borrow embankment cost is subtracted from the borrow excavation cost, so that the borrowed materials can be treated as normal materials.

4.1.2 Decision variables

The decision variables are

- V_i^+ (for all $i \in \mathcal{S} \cup \mathcal{B}$): the volume of earth excavated from any section,
- V_i^- (for all $i \in \mathcal{S} \cup \mathcal{W}$): the volume of earth embanked to any section,
- $f_{i,i-1,t}^r, f_{i,i+1,t}^r$ (for all $i \in \mathcal{N}, t \in \mathcal{T}$): the flow of earth from one transit node to the next in both directions,
- $f_{j,\vartheta(j)-1,t}^b, f_{j,\vartheta(j)+1,t}^b$ (for all $j \in \mathcal{B}, t \in \mathcal{T}$): the flow of earth to both east and waste transit nodes from a borrow pit,
- $f_{\varphi(k)-1,k,t}^w, f_{\varphi(k)+1,k,t}^w$ (for all $k \in \mathcal{W}, t \in \mathcal{T}$): the flow of earth from both east and west transit nodes to a waste pit,
- $f_{i-1,i,t}^l, f_{i+1,i,t}^l$ (for all $i \in \mathcal{N}, t \in \mathcal{T}$): the flow of earth from both east and west transit nodes to a section (also known as the load flow),
- $f_{i,i-1,t}^u, f_{i,i+1,t}^u$ (for all $i \in \mathcal{N}, t \in \mathcal{T}$): the flow of earth from a section to both east and west transit nodes (also known as the unload flow),
- $a_{g,k}$ (for all $g \in \mathcal{G}, k \in \{1, 2, 3\}$): the coefficient of the quadratic polynomials,

4.1. Definitions and model parameters

- u_i (for all $i \in \mathcal{S}$): the height difference of the ground profile from the road profile for any section,
- y_{kt} (for all $t \in \mathcal{T}, k \in \mathcal{I}$): the block removal indicators,
- b_i (for all $i \in \mathcal{S}$): binary variables that ensure that no roadside stockpiling will occur.

4.1.3 Design parameters

The design parameters for the MILP are

- m : the number of spline segments,
- n_g (for all $g \in \mathcal{G}$): the number of sections per spline segment,
- $s_{g,i}$ (for all $g \in \mathcal{G}, i \in \mathcal{S}_g$): the start and end points of the sections,
- $h_{g,i}$ (for all $g \in \mathcal{G}, i \in \mathcal{S}_g$): the average height of the ground for each section,
- \bar{u}_i and \underline{u}_i (for all $i \in \mathcal{S}$): the upper and lower bound of the variable u_i ,
- L (respectively U): the lower (respectively upper) bound of the slope,
- H_A (respectively H_B): the starting (respectively ending) elevation,
- H'_A (respectively H'_B): the starting (respectively ending) slope,
- $H_{g,i} \in \mathcal{H}$: the fixed points,
- n_z : the number of blocks,
- $A_i^{k_a}, B_i^{k_b}$ (for all $i \in \mathcal{S}, k_a \in \{1, 2, \dots, k_i^+\}, k_b \in \{1, 2, \dots, k_i^-\}$): area list for side-slope approximation,
- p_i, q_j , and c_{ij}^r (for all $i \in \mathcal{S}, j \in \{i-1, i+1\}$): excavation, embankment, and hauling costs for a section,
- c_i^b and c_j^w (for all $i \in \mathcal{B}, j \in \mathcal{W}$): borrow and waste costs, and
- the locations of the borrow pits, the waste pits, the blocks, and the access roads.

4.2 Model description

4.2.1 Objective function

The objective of this problem is to find a vertical alignment such that total excavation, embankment, and hauling cost will be minimized. The objective function can be written as

$$\begin{aligned} \min \sum_{i \in \mathcal{S}} (p_i V_i^+ + q_i V_i^-) + \sum_{t \in \mathcal{T}} \sum_{i \in \mathcal{S}} (c_{i,i-1}^r f_{i,i-1,t}^r + c_{i,i+1}^r f_{i,i+1,t}^r) \\ + \sum_{j \in \mathcal{B}} c_j^b \sum_{t \in \mathcal{T}} (f_{j,\vartheta(j)-1,t}^b + f_{j,\vartheta(j)+1,t}^b) + \sum_{k \in \mathcal{W}} c_k^w \sum_{t \in \mathcal{T}} (f_{\varphi(k)-1,k,t}^w + f_{\varphi(k)+1,k,t}^w). \end{aligned} \quad (4.1)$$

4.2.2 Flow constraints

The transit nodes are virtual points that provide transits for the flows. Therefore, the sum of the flows coming to a transit node must be equal to the sum of the flows leaving the node, i.e., for all $t \in \mathcal{T}$, $i \in \mathcal{S}$, the flow constraints for the east transit nodes are

$$f_{i-1,i,t}^r + f_{i,i+1,t}^u + \sum_{\substack{j \in \mathcal{B} \\ \vartheta(j)=i}} f_{j,i+1,t}^b = f_{i,i+1,t}^r + f_{i-1,i,t}^l + \sum_{\substack{k \in \mathcal{W} \\ \varphi(k)=i}} f_{i-1,k,t}^w, \quad (4.2a)$$

and for the west transit nodes are

$$f_{i+1,i,t}^r + f_{i,i-1,t}^u + \sum_{\substack{j \in \mathcal{B} \\ \vartheta(j)=i}} f_{j,i-1,t}^b = f_{i,i-1,t}^r + f_{i+1,i,t}^l + \sum_{\substack{k \in \mathcal{W} \\ \varphi(k)=i}} f_{i+1,k,t}^w. \quad (4.2b)$$

4.2.3 Balance constraints

The section nodes act as the sources and sinks for the flows. So, the total of unload flows from a section is equal to the cut volume of that section, i.e., for all $i \in \mathcal{S}$,

$$\sum_{t \in \mathcal{T}} (f_{i,i-1,t}^u + f_{i,i+1,t}^u) = V_i^+, \quad (4.3a)$$

and the total of load flows to a section is equal to the fill volume i.e., for all $i \in \mathcal{S}$,

$$\sum_{t \in \mathcal{T}} (f_{i-1,i,t}^l + f_{i+1,i,t}^l) = V_i^-. \quad (4.3b)$$

4.2.4 Pit constraints

The borrow pits act as the sources of the flows with additional costs, and the waste pits act as the sinks of the flows with additional costs. The total of the borrow flows from a borrow pit must not exceed its capacity, i.e., for all $j \in \mathcal{B}$,

$$\sum_{t \in \mathcal{T}} (f_{j, \vartheta(j)-1, t}^b + f_{j, \vartheta(j)+1, t}^b) \leq C_j^b, \quad (4.4a)$$

and the total of waste flows to a waste pit must not exceed its capacity, i.e., for all $k \in \mathcal{W}$,

$$\sum_{t \in \mathcal{T}} (f_{\varphi(k)-1, k, t}^w + f_{\varphi(k)+1, k, t}^w) \leq C_k^w. \quad (4.4b)$$

4.2.5 Block constraints

The block constraints must certify that no earth movement will occur over a block until the block is removed. We consider two cases for this constraints, namely a block with no borrow pit or waste pit with it and a block with a borrow and a waste pit with it. Using the two cases we write the block constraints for a general case.

Theorem 4.2. (Case I) *If a section i is a block with no borrow or waste pits associated with it, then for all $t \in \mathcal{T}$, $f_{i-1, i, t}^r = f_{i-1, i, t}^l$ and $f_{i+1, i, t}^r = f_{i+1, i, t}^l$ will ensure that no earth movement will occur over section i .*

Proof. Since $f_{i-1, i, t}^r = f_{i-1, i, t}^l$ and $f_{i+1, i, t}^r = f_{i+1, i, t}^l$, by Equations (4.2a) and (4.2a), $f_{i, i+1, t}^r = f_{i, i+1, t}^u$ and $f_{i, i-1, t}^r = f_{i, i-1, t}^u$ hold. Roadside stockpiling constraints ensure that one of V_i^+ and V_i^- will be 0. By Equations (4.3a) and (4.3b), one of $f_{i-1, i, t}^r = f_{i-1, i, t}^l = f_{i+1, i, t}^r = f_{i+1, i, t}^l = 0$ and $f_{i, i-1, t}^r = f_{i, i-1, t}^u = f_{i, i+1, t}^r = f_{i, i+1, t}^u = 0$ must hold. This means either a block will be embanked from both directions or it will be excavated to both directions, i.e., no earth movement will occur over the block. \square

Since no earth movement occurs until the block is removed, by Theorem 4.2, for all $t \in \mathcal{T}$, $k \in \mathcal{I}$, the constraints can be written (for this special case) as,

$$- \left(f_{\varsigma(k)-1, \varsigma(k), t}^r - f_{\varsigma(k)-1, \varsigma(k), t}^l \right) \leq My_{k, t-1}, \quad (4.5a)$$

$$f_{\varsigma(k)-1, \varsigma(k), t}^r - f_{\varsigma(k)-1, \varsigma(k), t}^l \leq My_{k, t-1}, \quad (4.5b)$$

$$- \left(f_{\varsigma(k)+1, \varsigma(k), t}^r - f_{\varsigma(k)+1, \varsigma(k), t}^l \right) \leq My_{k, t-1}, \quad (4.5c)$$

4.2. Model description

$$f_{\zeta(k)+1,\zeta(k),t}^r - f_{\zeta(k)+1,\zeta(k),t}^l \leq My_{k,t-1}, \quad (4.5d)$$

Theorem 4.3. (Case II) *If a section i is a block with a borrow pit j and a waste pit k associated with it, then for all $t \in \mathcal{T}$,*

$$\begin{aligned} f_{i-1,i,t}^r + f_{j,i+1,t}^{b,1} &= f_{i-1,i,t}^l + f_{i-1,k,t}^{w,1}, \\ f_{i,i+1,t}^r + f_{i-1,k,t}^{w,2} &= f_{i,i+1,t}^u + f_{j,i+1,t}^{b,2}, \\ f_{i+1,i,t}^r + f_{j,i-1,t}^{b,1} &= f_{i+1,i,t}^l + f_{i+1,k,t}^{w,1}, \text{ and} \\ f_{i,i-1,t}^r + f_{i+1,k,t}^{w,2} &= f_{i,i-1,t}^u + f_{j,i-1,t}^{b,2} \end{aligned}$$

will ensure that no earth movement will occur over section i where

$$\begin{aligned} f_{j,i+1,t}^{b,1} + f_{j,i+1,t}^{b,2} &= f_{j,i+1,t}^b, \\ f_{i-1,k,t}^{w,1} + f_{i-1,k,t}^{w,2} &= f_{i-1,k,t}^w, \\ f_{j,i-1,t}^{b,1} + f_{j,i-1,t}^{b,2} &= f_{j,i-1,t}^b, \text{ and} \\ f_{i+1,k,t}^{w,1} + f_{i+1,k,t}^{w,2} &= f_{i+1,k,t}^w. \end{aligned}$$

Proof. For a block with a borrow pit and a waste pit, the borrow and waste flows are divided into two flows. Since roadside stockpiling constraints ensure that one of V_i^+ and V_i^- will be 0, by Equations (4.3a) and (4.3b), one of $f_{i-1,i,t}^l = f_{i+1,i,t}^l = 0$ and $f_{i,i-1,t}^u = f_{i,i+1,t}^u = 0$ must hold. So, either $f_{i-1,i,t}^r + f_{j,i+1,t}^{b,1} = f_{i-1,k,t}^{w,1}$ and $f_{i+1,i,t}^r + f_{j,i-1,t}^{b,1} = f_{i+1,k,t}^{w,1}$ or $f_{i,i+1,t}^r + f_{i-1,k,t}^{w,2} = f_{j,i+1,t}^{b,2}$ and $f_{i,i-1,t}^r + f_{i+1,k,t}^{w,2} = f_{j,i-1,t}^{b,2}$ must hold. This means either a block can be embanked from both directions or excavated to both directions, i.e., no earth movement will occur over the block. \square

Since no earth movement occurs until the block is removed, by Theorem 4.3, for all $t \in \mathcal{T}$, $k \in \mathcal{I}$, the constraints can be written (for this special case) as,

$$f_{\zeta(k)-1,\zeta(k),t}^r + f_{j,\zeta(k)+1,t}^{b,1} - f_{\zeta(k)-1,\zeta(k),t}^l - f_{\zeta(k)-1,j,t}^{w,1} \geq -My_{k,t-1}, \quad (4.5e)$$

$$f_{\zeta(k)-1,\zeta(k),t}^r + f_{j,\zeta(k)+1,t}^{b,1} - f_{\zeta(k)-1,\zeta(k),t}^l - f_{\zeta(k)-1,j,t}^{w,1} \leq My_{k,t-1}, \quad (4.5f)$$

$$f_{\zeta(k),\zeta(k)+1,t}^r + f_{\zeta(k)-1,j,t}^{w,2} - f_{\zeta(k),\zeta(k)+1,t}^l - f_{j,\zeta(k)+1,t}^{b,2} \geq -My_{k,t-1}, \quad (4.5g)$$

4.2. Model description

$$f_{\varsigma(k),\varsigma(k)+1,t}^r + f_{\varsigma(k)-1,j,t}^{w,2} - f_{\varsigma(k),\varsigma(k)+1,t}^u - f_{j,\varsigma(k)+1,t}^{b,2} \leq My_{k,t-1}, \quad (4.5h)$$

$$f_{\varsigma(k)+1,\varsigma(k),t}^r + f_{j,\varsigma(k)-1,t}^{b,1} - f_{\varsigma(k)+1,\varsigma(k),t}^l - f_{\varsigma(k)+1,j,t}^{w,1} \geq -My_{k,t-1}, \quad (4.5i)$$

$$f_{\varsigma(k)+1,\varsigma(k),t}^r + f_{j,\varsigma(k)-1,t}^{b,1} - f_{\varsigma(k)+1,\varsigma(k),t}^l - f_{\varsigma(k)+1,j,t}^{w,1} \leq My_{k,t-1}, \quad (4.5j)$$

$$f_{\varsigma(k),\varsigma(k)-1,t}^r + f_{\varsigma(k)+1,j,t}^{w,2} - f_{\varsigma(k),\varsigma(k)-1,t}^u - f_{j,\varsigma(k)-1,t}^{b,2} \geq -My_{k,t-1}, \quad (4.5k)$$

$$f_{\varsigma(k),\varsigma(k)-1,t}^r + f_{\varsigma(k)+1,j,t}^{w,2} - f_{\varsigma(k),\varsigma(k)-1,t}^u - f_{j,\varsigma(k)-1,t}^{b,2} \leq My_{k,t-1}. \quad (4.5l)$$

By Theorems 4.2 and 4.3, for all $t \in \mathcal{T}$, $k \in \mathcal{I}$, the constraints can be written (for the general case) as,

$$f_{\varsigma(k)-1,\varsigma(k),t}^r + \sum_{\substack{j \in \mathcal{B} \\ \vartheta(j)=\varsigma(k)}} f_{j,\varsigma(k)+1,t}^{b,1} - f_{\varsigma(k)-1,\varsigma(k),t}^l - \sum_{\substack{j \in \mathcal{W} \\ \varphi(j)=\varsigma(k)}} f_{\varsigma(k)-1,j,t}^{w,1} \geq -My_{k,t-1}, \quad (4.5m)$$

$$f_{\varsigma(k)-1,\varsigma(k),t}^r + \sum_{\substack{j \in \mathcal{B} \\ \vartheta(j)=\varsigma(k)}} f_{j,\varsigma(k)+1,t}^{b,1} - f_{\varsigma(k)-1,\varsigma(k),t}^l - \sum_{\substack{j \in \mathcal{W} \\ \varphi(j)=\varsigma(k)}} f_{\varsigma(k)-1,j,t}^{w,1} \leq My_{k,t-1}, \quad (4.5n)$$

$$f_{\varsigma(k),\varsigma(k)+1,t}^r + \sum_{\substack{j \in \mathcal{W} \\ \varphi(j)=\varsigma(k)}} f_{\varsigma(k)-1,j,t}^{w,2} - f_{\varsigma(k),\varsigma(k)+1,t}^u - \sum_{\substack{j \in \mathcal{B} \\ \vartheta(j)=\varsigma(k)}} f_{j,\varsigma(k)+1,t}^{b,2} \geq -My_{k,t-1}, \quad (4.5o)$$

$$f_{\varsigma(k),\varsigma(k)+1,t}^r + \sum_{\substack{j \in \mathcal{W} \\ \varphi(j)=\varsigma(k)}} f_{\varsigma(k)-1,j,t}^{w,2} - f_{\varsigma(k),\varsigma(k)+1,t}^u - \sum_{\substack{j \in \mathcal{B} \\ \vartheta(j)=\varsigma(k)}} f_{j,\varsigma(k)+1,t}^{b,2} \leq My_{k,t-1}, \quad (4.5p)$$

$$f_{\varsigma(k)+1,\varsigma(k),t}^r + \sum_{\substack{j \in \mathcal{B} \\ \vartheta(j)=\varsigma(k)}} f_{j,\varsigma(k)-1,t}^{b,1} - f_{\varsigma(k)+1,\varsigma(k),t}^l - \sum_{\substack{j \in \mathcal{W} \\ \varphi(j)=\varsigma(k)}} f_{\varsigma(k)+1,j,t}^{w,1} \geq -My_{k,t-1}, \quad (4.5q)$$

$$f_{\varsigma(k)+1,\varsigma(k),t}^r + \sum_{\substack{j \in \mathcal{B} \\ \vartheta(j)=\varsigma(k)}} f_{j,\varsigma(k)-1,t}^{b,1} - f_{\varsigma(k)+1,\varsigma(k),t}^l - \sum_{\substack{j \in \mathcal{W} \\ \varphi(j)=\varsigma(k)}} f_{\varsigma(k)+1,j,t}^{w,1} \leq My_{k,t-1}, \quad (4.5r)$$

$$f_{\varsigma(k),\varsigma(k)-1,t}^r + \sum_{\substack{j \in \mathcal{W} \\ \varphi(j)=\varsigma(k)}} f_{\varsigma(k)+1,j,t}^{w,2} - f_{\varsigma(k),\varsigma(k)-1,t}^u - \sum_{\substack{j \in \mathcal{B} \\ \vartheta(j)=\varsigma(k)}} f_{j,\varsigma(k)-1,t}^{b,2} \geq -My_{k,t-1}, \quad (4.5s)$$

$$f_{\varsigma(k),\varsigma(k)-1,t}^r + \sum_{\substack{j \in \mathcal{W} \\ \varphi(j)=\varsigma(k)}} f_{\varsigma(k)+1,j,t}^{w,2} - f_{\varsigma(k),\varsigma(k)-1,t}^u - \sum_{\substack{j \in \mathcal{B} \\ \vartheta(j)=\varsigma(k)}} f_{j,\varsigma(k)-1,t}^{b,2} \leq My_{k,t-1}. \quad (4.5t)$$

4.2. Model description

Since the flows from borrow pits and waste pits, which are associated with blocks, are divided into two flows each, for all $t \in \mathcal{T}$, $j \in \mathcal{B}$, $k \in \mathcal{I}$, $\vartheta(j) = \varsigma(k)$,

$$f_{j,i+1,t}^{b,1} + f_{j,i+1,t}^{b,2} = f_{j,i+1,t}^b, \quad (4.5u)$$

$$f_{j,i-1,t}^{b,1} + f_{j,i-1,t}^{b,2} = f_{j,i-1,t}^b, \quad (4.5v)$$

and for all $t \in \mathcal{T}$, $j \in \mathcal{W}$, $k \in \mathcal{I}$, $\varphi(j) = \varsigma(k)$,

$$f_{i-1,k,t}^{w,1} + f_{i-1,k,t}^{w,2} = f_{i-1,k,t}^w, \quad (4.5w)$$

$$f_{i+1,k,t}^{w,1} + f_{i+1,k,t}^{w,2} = f_{i+1,k,t}^w. \quad (4.5x)$$

The block constraints must also ensure that no earth movement will occur among sections, borrow pits, and waste pits between two blocks, before the first block, or after the last block with no access roads until the block is removed. So, there will be no transit flow between two blocks with no access roads until one of the blocks is removed, i.e., for all $t \in \mathcal{T}$, $(k_1, k_2) \in \overline{\mathcal{I}}^2$, $\varsigma(k_1) \leq i, i + 1 \leq \varsigma(k_2)$,

$$f_{i,i+1,t}^r \leq M(y_{k_1,t-1} + y_{k_2,t-1}), \quad (4.6a)$$

$$f_{i+1,i,t}^r \leq M(y_{k_1,t-1} + y_{k_2,t-1}). \quad (4.6b)$$

There will also be no transit flow after the last block with no access road until the block is removed, i.e., for all $t \in \mathcal{T}$, $\varsigma(k) \leq i, i + 1 \leq n, k \in \overline{\mathcal{I}}_{\rightarrow}$,

$$f_{i,i+1,t}^r \leq M y_{k,t-1}, \quad (4.6c)$$

$$f_{i+1,i,t}^r \leq M y_{k,t-1}, \quad (4.6d)$$

and before the first block with no access road until the block is removed, i.e., for all $t \in \mathcal{T}$, $1 \leq i, i + 1 \leq \varsigma(k), k \in \overline{\mathcal{I}}_{\leftarrow}$,

$$f_{i,i+1,t}^r \leq M y_{k,t-1}, \quad (4.6e)$$

$$f_{i+1,i,t}^r \leq M y_{k,t-1}. \quad (4.6f)$$

There will be no borrow flow between two blocks with no access roads until one of the blocks is removed, i.e., for all $t \in \mathcal{T}$, $(k_1, k_2) \in \overline{\mathcal{I}}^2$, $\varsigma(k_1) \leq \vartheta(i) - 1, \vartheta(i), \vartheta(i) + 1 \leq \varsigma(k_2)$,

$$f_{i,\vartheta(i)+1,t}^b \leq M(y_{k_1,t-1} + y_{k_2,t-1}). \quad (4.6g)$$

$$f_{i,\vartheta(i)-1,t}^b \leq M(y_{k_1,t-1} + y_{k_2,t-1}). \quad (4.6h)$$

4.2. Model description

There will also be no borrow flow after the last block with no access road until the block is removed, i.e., for all $t \in \mathcal{T}$, $\varsigma(k) \leq \vartheta(i) - 1$, $\vartheta(i)$, $\vartheta(i) + 1 \leq n$, $k \in \bar{\mathcal{I}}_{\rightarrow}$,

$$f_{i,\vartheta(i)+1,t}^b \leq M y_{k,t-1}, \quad (4.6i)$$

$$f_{i,\vartheta(i)-1,t}^b \leq M y_{k,t-1}, \quad (4.6j)$$

and before the first block with no access road until the block is removed, i.e., for all $t \in \mathcal{T}$, $1 \leq \vartheta(i) - 1$, $\vartheta(i)$, $\vartheta(i) + 1 \leq \varsigma(k)$, $k \in \bar{\mathcal{I}}_{\leftarrow}$,

$$f_{i,\vartheta(i)+1,t}^b \leq M y_{k,t-1}, \quad (4.6k)$$

$$f_{i,\vartheta(i)-1,t}^b \leq M y_{k,t-1}. \quad (4.6l)$$

There will be no waste flow between two blocks with no access roads until one of the blocks is removed, i.e., for all $t \in \mathcal{T}$, $(k_1, k_2) \in \bar{\mathcal{I}}^2$, $\varsigma(k_1) \leq \varphi(i) - 1$, $\varphi(i)$, $\varphi(i) + 1 \leq \varsigma(k_2)$,

$$f_{\varphi(i)+1,i,t}^w \leq M(y_{k_1,t-1} + y_{k_2,t-1}), \quad (4.6m)$$

$$f_{\varphi(i)-1,i,t}^w \leq M(y_{k_1,t-1} + y_{k_2,t-1}). \quad (4.6n)$$

There will also be no waste flow after the last block with no access road until the block is removed, i.e., for all $t \in \mathcal{T}$, $\varsigma(k) \leq \varphi(i) - 1$, $\varphi(i)$, $\varphi(i) + 1 \leq n$, $k \in \bar{\mathcal{I}}_{\rightarrow}$,

$$f_{\varphi(i)+1,i,t}^w \leq M y_{k,t-1}, \quad (4.6o)$$

$$f_{\varphi(i)-1,i,t}^w \leq M y_{k,t-1}, \quad (4.6p)$$

and before the first block with no access road until the block is removed, i.e., for all $t \in \mathcal{T}$, $1 \leq \varphi(i) - 1$, $\varphi(i)$, $\varphi(i) + 1 \leq \varsigma(k)$, $k \in \bar{\mathcal{I}}_{\leftarrow}$,

$$f_{\varphi(i)+1,i,t}^w \leq M y_{k,t-1}, \quad (4.6q)$$

$$f_{\varphi(i)-1,i,t}^w \leq M y_{k,t-1}, \quad (4.6r)$$

The block removal indicator constraints ensure that when the required amount of earth is excavated from or embanked to a section with a block, then the block is considered removed. Similar to the model of Section 3.3, for all $k \in \mathcal{I}$, $t \in \mathcal{T}$, the following constraints ensure that $W_{kt}^+ = y_{kt} V_{\varsigma(k)}^+$,

$$W_{kt}^+ \leq M_{\varsigma(k)}^+ y_{kt}, \quad (3.12i)$$

$$W_{kt}^+ \leq V_{\varsigma(k)}^+, \quad (3.12j)$$

$$W_{kt}^+ \geq V_{\varsigma(k)}^+ - M_{\varsigma(k)}^+ (1 - y_{kt}), \quad (3.12k)$$

$$W_{kt}^+ \geq 0, \quad (3.12l)$$

4.2. Model description

and the following constraints assure that $W_{kt}^- = y_{kt}V_{\varsigma(k)}^-$,

$$W_{kt}^- \leq M_{\varsigma(k)}^- y_{kt}, \quad (3.12m)$$

$$W_{kt}^- \leq V_{\varsigma(k)}^-, \quad (3.12n)$$

$$W_{kt}^- \geq V_{\varsigma(k)}^- - M_{\varsigma(k)}^- (1 - y_{kt}), \quad (3.12o)$$

$$W_{kt}^- \geq 0. \quad (3.12p)$$

So, for all $k \in \mathcal{I}, t \in \mathcal{T}$, the block removal indicator constraints can be written as

$$\sum_{t=1}^u \left(f_{\varsigma(k), \varsigma(k)-1, t}^u + f_{\varsigma(k), \varsigma(k)+1, t}^u + \sum_{\substack{j \in \mathcal{W} \\ \varphi(j) = \varsigma(k)}} f_{j, t}^w \right) \geq W_{kt}^+, \quad (4.7)$$

$$\sum_{t=1}^u \left(f_{\varsigma(k)-1, \varsigma(k), t}^l + f_{\varsigma(k)+1, \varsigma(k), t}^l + \sum_{\substack{j \in \mathcal{B} \\ \vartheta(j) = \varsigma(k)}} f_{j, t}^b \right) \geq W_{kt}^-. \quad (4.8)$$

In this model, we redefine these constraints so that the W_{kt}^+ and W_{kt}^- (for all $k \in \mathcal{I}, t \in \mathcal{T}$) variables are no longer needed. For all $k \in \mathcal{I}, u \in \mathcal{T}$, these constraints can be written as,

$$\sum_{t=1}^u \left(f_{\varsigma(k), \varsigma(k)-1, t}^u + f_{\varsigma(k), \varsigma(k)+1, t}^u + \sum_{\substack{j \in \mathcal{W} \\ \varphi(j) = \varsigma(k)}} f_{j, t}^w \right) + M_{\varsigma(k)}^+ (1 - y_{ku}) \geq V_{\varsigma(k)}^+, \quad (4.9a)$$

$$\sum_{t=1}^u \left(f_{\varsigma(k)-1, \varsigma(k), t}^l + f_{\varsigma(k)+1, \varsigma(k), t}^l + \sum_{\substack{j \in \mathcal{B} \\ \vartheta(j) = \varsigma(k)}} f_{j, t}^b \right) + M_{\varsigma(k)}^- (1 - y_{ku}) \geq V_{\varsigma(k)}^-. \quad (4.9b)$$

It should be noted that Equations (4.9a) and (4.9b) are logically equivalent to Equations (3.12i) to (3.12p) and (4.7) to (4.8). The remaining block constraints are the same as Section 3.3, i.e., for all $t \in \mathcal{T} \setminus \{n_z + 1\}$,

$$\sum_{k \in \mathcal{I}} y_{kt} \geq t, \quad (3.12s)$$

and for all $k \in \mathcal{I}, t \in \mathcal{T} \setminus \{n_z + 1\}$,

$$(y_{k, (t+1)} \geq y_{k, t}). \quad (3.12t)$$

4.2.6 Other constraints

The following constraints are exactly the same as in Section 3.3: roadside stockpiling constraints, volume constraints, gap constraints, slope constraints, smoothness constraints, and fixed-point constraints.

4.2.7 Bounds

The bounds for the decision variables are, for all $i \in \mathcal{N}, t \in \mathcal{T}$,

$$f_{i,i-1,t}^r \geq 0, \quad (4.10)$$

$$f_{i,i+1,t}^r \geq 0, \quad (4.11)$$

$$f_{i-1,i,t}^l \geq 0, \quad (4.12)$$

$$f_{i+1,i,t}^l \geq 0, \quad (4.13)$$

$$f_{i,i-1,t}^u \geq 0, \quad (4.14)$$

$$f_{i,i+1,t}^u \geq 0, \quad (4.15)$$

for all $j \in \mathcal{B}, t \in \mathcal{T}$,

$$f_{j,\vartheta(j)-1,t}^b \geq 0, \quad (4.16)$$

$$f_{j,\vartheta(j)+1,t}^b \geq 0, \quad (4.17)$$

for all $k \in \mathcal{W}, t \in \mathcal{T}$

$$f_{\varphi(k)-1,k,t}^w \geq 0, \quad (4.18)$$

$$f_{\varphi(k)+1,k,t}^w \geq 0, \quad (4.19)$$

for all $k \in \mathcal{I}, t \in \mathcal{T}$,

$$y_{kt} \in \{0, 1\}, \quad (3.20b)$$

for all $i \in \mathcal{S}$,

$$\underline{u}_i \leq u_i \leq \bar{u}_i, \quad (3.20c)$$

$$0 \leq V_i^+ \leq M_i^+, \quad (3.20d)$$

$$0 \leq V_i^- \leq M_i^-, \quad (3.20e)$$

and for all $g \in \mathcal{G}, k \in \{1, 2, 3\}$,

$$a_{g,k} \in \mathbb{R}. \quad (3.20f)$$

4.3 Model summary

The summary of the quasi-network-flow model for the vertical alignment design considering blocks and side-slopes is stated below.

4.3.1 Objective function

$$\begin{aligned} \min \sum_{i \in \mathcal{S}} (p_i V_i^+ + q_i V_i^-) &+ \sum_{t \in \mathcal{T}} \sum_{i \in \mathcal{S}} (c_{i,i-1}^r f_{i,i-1,t}^r + c_{i,i+1}^r f_{i,i+1,t}^r) \\ &+ \sum_{j \in \mathcal{B}} c_j^b \sum_{t \in \mathcal{T}} (f_{j,\vartheta(j)-1,t}^b + f_{j,\vartheta(j)+1,t}^b) + \sum_{k \in \mathcal{W}} c_k^w \sum_{t \in \mathcal{T}} (f_{\varphi(k)-1,k,t}^w + f_{\varphi(k)+1,k,t}^w). \end{aligned} \quad (4.1)$$

4.3.2 Flow constraints

For all $t \in \mathcal{T}$, $i \in \mathcal{S}$,

$$f_{i-1,i,t}^r + f_{i,i+1,t}^u + \sum_{\substack{j \in \mathcal{B} \\ \vartheta(j)=i}} f_{j,i+1,t}^b = f_{i,i+1,t}^r + f_{i-1,i,t}^l + \sum_{\substack{k \in \mathcal{W} \\ \varphi(k)=i}} f_{i-1,k,t}^w, \quad (4.2a)$$

$$f_{i+1,i,t}^r + f_{i,i-1,t}^u + \sum_{\substack{j \in \mathcal{B} \\ \vartheta(j)=i}} f_{j,i-1,t}^b = f_{i,i-1,t}^r + f_{i+1,i,t}^l + \sum_{\substack{k \in \mathcal{W} \\ \varphi(k)=i}} f_{i+1,k,t}^w, \quad (4.2b)$$

4.3.3 Balance constraints

For all $i \in \mathcal{S}$,

$$\sum_{t \in \mathcal{T}} (f_{i,i-1,t}^u + f_{i,i+1,t}^u) = V_i^+, \quad (4.3a)$$

$$\sum_{t \in \mathcal{T}} (f_{i-1,i,t}^l + f_{i+1,i,t}^l) = V_i^-. \quad (4.3b)$$

4.3.4 Pit constraints

For all $j \in \mathcal{B}$,

$$\sum_{t \in \mathcal{T}} (f_{j,\vartheta(j)-1,t}^b + f_{j,\vartheta(j)+1,t}^b) \leq C_j^b, \quad (4.4a)$$

and for all $k \in \mathcal{W}$,

$$\sum_{t \in \mathcal{T}} (f_{\varphi(k)-1,k,t}^w + f_{\varphi(k)+1,k,t}^w) \leq C_k^w. \quad (4.4b)$$

4.3. Model summary

4.3.5 Block constraints

For all $t \in \mathcal{T}, k \in \mathcal{I}$,

$$f_{\varsigma(k)-1, \varsigma(k), t}^r + \sum_{\substack{j \in \mathcal{B} \\ \vartheta(j) = \varsigma(k)}} f_{j, \varsigma(k)+1, t}^{b,1} - f_{\varsigma(k)-1, \varsigma(k), t}^l - \sum_{\substack{j \in \mathcal{W} \\ \varphi(j) = \varsigma(k)}} f_{\varsigma(k)-1, j, t}^{w,1} \geq -My_{k, t-1}, \quad (4.5m)$$

$$f_{\varsigma(k)-1, \varsigma(k), t}^r + \sum_{\substack{j \in \mathcal{B} \\ \vartheta(j) = \varsigma(k)}} f_{j, \varsigma(k)+1, t}^{b,1} - f_{\varsigma(k)-1, \varsigma(k), t}^l - \sum_{\substack{j \in \mathcal{W} \\ \varphi(j) = \varsigma(k)}} f_{\varsigma(k)-1, j, t}^{w,1} \leq My_{k, t-1}, \quad (4.5n)$$

$$f_{\varsigma(k), \varsigma(k)+1, t}^r + \sum_{\substack{j \in \mathcal{W} \\ \varphi(j) = \varsigma(k)}} f_{\varsigma(k)-1, j, t}^{w,2} - f_{\varsigma(k), \varsigma(k)+1, t}^u - \sum_{\substack{j \in \mathcal{B} \\ \vartheta(j) = \varsigma(k)}} f_{j, \varsigma(k)+1, t}^{b,2} \geq -My_{k, t-1}, \quad (4.5o)$$

$$f_{\varsigma(k), \varsigma(k)+1, t}^r + \sum_{\substack{j \in \mathcal{W} \\ \varphi(j) = \varsigma(k)}} f_{\varsigma(k)-1, j, t}^{w,2} - f_{\varsigma(k), \varsigma(k)+1, t}^u - \sum_{\substack{j \in \mathcal{B} \\ \vartheta(j) = \varsigma(k)}} f_{j, \varsigma(k)+1, t}^{b,2} \leq My_{k, t-1}, \quad (4.5p)$$

$$f_{\varsigma(k)+1, \varsigma(k), t}^r + \sum_{\substack{j \in \mathcal{B} \\ \vartheta(j) = \varsigma(k)}} f_{j, \varsigma(k)-1, t}^{b,1} - f_{\varsigma(k)+1, \varsigma(k), t}^l - \sum_{\substack{j \in \mathcal{W} \\ \varphi(j) = \varsigma(k)}} f_{\varsigma(k)+1, j, t}^{w,1} \geq -My_{k, t-1}, \quad (4.5q)$$

$$f_{\varsigma(k)+1, \varsigma(k), t}^r + \sum_{\substack{j \in \mathcal{B} \\ \vartheta(j) = \varsigma(k)}} f_{j, \varsigma(k)-1, t}^{b,1} - f_{\varsigma(k)+1, \varsigma(k), t}^l - \sum_{\substack{j \in \mathcal{W} \\ \varphi(j) = \varsigma(k)}} f_{\varsigma(k)+1, j, t}^{w,1} \leq My_{k, t-1}, \quad (4.5r)$$

$$f_{\varsigma(k), \varsigma(k)-1, t}^r + \sum_{\substack{j \in \mathcal{W} \\ \varphi(j) = \varsigma(k)}} f_{\varsigma(k)+1, j, t}^{w,2} - f_{\varsigma(k), \varsigma(k)-1, t}^u - \sum_{\substack{j \in \mathcal{B} \\ \vartheta(j) = \varsigma(k)}} f_{j, \varsigma(k)-1, t}^{b,2} \geq -My_{k, t-1}, \quad (4.5s)$$

$$f_{\varsigma(k), \varsigma(k)-1, t}^r + \sum_{\substack{j \in \mathcal{W} \\ \varphi(j) = \varsigma(k)}} f_{\varsigma(k)+1, j, t}^{w,2} - f_{\varsigma(k), \varsigma(k)-1, t}^u - \sum_{\substack{j \in \mathcal{B} \\ \vartheta(j) = \varsigma(k)}} f_{j, \varsigma(k)-1, t}^{b,2} \leq My_{k, t-1}. \quad (4.5t)$$

for all $t \in \mathcal{T}, (k_1, k_2) \in \overline{\mathcal{I}}^2, \varsigma(k_1) \leq i, i+1 \leq \varsigma(k_2)$,

$$f_{i, i+1, t}^r \leq M(y_{k_1, t-1} + y_{k_2, t-1}), \quad (4.6a)$$

$$f_{i+1, i, t}^r \leq M(y_{k_1, t-1} + y_{k_2, t-1}), \quad (4.6b)$$

for all $t \in \mathcal{T}, \varsigma(k) \leq i, i+1 \leq n, k \in \overline{\mathcal{I}}_{\rightarrow}$,

$$f_{i, i+1, t}^r \leq My_{k, t-1}, \quad (4.6c)$$

$$f_{i+1, i, t}^r \leq My_{k, t-1}, \quad (4.6d)$$

4.3. Model summary

for all $t \in \mathcal{T}$, $1 \leq i, i+1 \leq \varsigma(k)$, $k \in \overline{\mathcal{I}}_{\leftarrow}$,

$$f_{i,i+1,t}^r \leq My_{k,t-1}, \quad (4.6e)$$

$$f_{i,i+1,t}^r \leq My_{k,t-1}, \quad (4.6f)$$

for all $t \in \mathcal{T}$, $(k_1, k_2) \in \overline{\mathcal{I}}^2$, $\varsigma(k_1) \leq \vartheta(i) - 1$, $\vartheta(i)$, $\vartheta(i) + 1 \leq \varsigma(k_2)$,

$$f_{i,\vartheta(i)+1,t}^b \leq M(y_{k_1,t-1} + y_{k_2,t-1}), \quad (4.6g)$$

$$f_{i,\vartheta(i)-1,t}^b \leq M(y_{k_1,t-1} + y_{k_2,t-1}), \quad (4.6h)$$

for all $t \in \mathcal{T}$, $\varsigma(k) \leq \vartheta(i) - 1$, $\vartheta(i)$, $\vartheta(i) + 1 \leq n$, $k \in \overline{\mathcal{I}}_{\rightarrow}$,

$$f_{i,\vartheta(i)+1,t}^b \leq My_{k,t-1}, \quad (4.6i)$$

$$f_{i,\vartheta(i)-1,t}^b \leq My_{k,t-1}, \quad (4.6j)$$

for all $t \in \mathcal{T}$, $1 \leq \vartheta(i) - 1$, $\vartheta(i)$, $\vartheta(i) + 1 \leq \varsigma(k)$, $k \in \overline{\mathcal{I}}_{\leftarrow}$,

$$f_{i,\vartheta(i)+1,t}^b \leq My_{k,t-1}, \quad (4.6k)$$

$$f_{i,\vartheta(i)-1,t}^b \leq My_{k,t-1}, \quad (4.6l)$$

for all $t \in \mathcal{T}$, $(k_1, k_2) \in \overline{\mathcal{I}}^2$, $\varsigma(k_1) \leq \varphi(i) - 1$, $\varphi(i)$, $\varphi(i) + 1 \leq \varsigma(k_2)$,

$$f_{\varphi(i)+1,i,t}^w \leq M(y_{k_1,t-1} + y_{k_2,t-1}). \quad (4.6m)$$

$$f_{\varphi(i)-1,i,t}^w \leq M(y_{k_1,t-1} + y_{k_2,t-1}). \quad (4.6n)$$

for all $t \in \mathcal{T}$, $\varsigma(k) \leq \varphi(i) - 1$, $\varphi(i)$, $\varphi(i) + 1 \leq n$, $k \in \overline{\mathcal{I}}_{\rightarrow}$,

$$f_{\varphi(i)+1,i,t}^w \leq My_{k,t-1}, \quad (4.6o)$$

$$f_{\varphi(i)-1,i,t}^w \leq My_{k,t-1}, \quad (4.6p)$$

for all $t \in \mathcal{T}$, $1 \leq \varphi(i) - 1$, $\varphi(i)$, $\varphi(i) + 1 \leq \varsigma(k)$, $k \in \overline{\mathcal{I}}_{\leftarrow}$,

$$f_{\varphi(i)+1,i,t}^w \leq My_{k,t-1}, \quad (4.6q)$$

$$f_{\varphi(i)+1,i,t}^w \leq My_{k,t-1}, \quad (4.6r)$$

for all $t \in \mathcal{T}$, $j \in \mathcal{B}$, $k \in \mathcal{I}$, $\vartheta(j) = \varsigma(k)$,

$$f_{j,i+1,t}^{b,1} + f_{j,i+1,t}^{b,2} = f_{j,i+1,t}^b, \quad (4.5u)$$

4.3. Model summary

$$f_{j,i-1,t}^{b,1} + f_{j,i-1,t}^{b,2} = f_{j,i-1,t}^b, \quad (4.5v)$$

for all $t \in \mathcal{T}$, $j \in \mathcal{W}$, $k \in \mathcal{I}$, $\varphi(j) = \varsigma(k)$,

$$f_{i-1,k,t}^{w,1} + f_{i-1,k,t}^{w,2} = f_{i-1,k,t}^w, \quad (4.5w)$$

$$f_{i+1,k,t}^{w,1} + f_{i+1,k,t}^{w,2} = f_{i+1,k,t}^w, \quad (4.5x)$$

for all $k \in \mathcal{I}$, $u \in \mathcal{T}$,

$$\sum_{t=1}^u (f_{\varsigma(k),\varsigma(k)-1,t}^u + f_{\varsigma(k),\varsigma(k)+1,t}^u + \sum_{\substack{j \in \mathcal{W} \\ \varphi(j)=\varsigma(k)}} f_{j,t}^w) + M_{\varsigma(k)}^+ (1 - y_{ku}) \geq V_{\varsigma(k)}^+, \quad (4.9a)$$

$$\sum_{t=1}^u (f_{\varsigma(k)-1,\varsigma(k),t}^l + f_{\varsigma(k)+1,\varsigma(k),t}^l + \sum_{\substack{j \in \mathcal{B} \\ \theta(j)=\varsigma(k)}} f_{j,t}^b) + M_{\varsigma(k)}^- (1 - y_{ku}) \geq V_{\varsigma(k)}^-, \quad (4.9b)$$

for all $t \in \mathcal{T} \setminus \{n_z + 1\}$,

$$\sum_{k \in \mathcal{I}} y_{kt} \geq t, \quad (3.12s)$$

and for all $k \in \mathcal{I}$, $t \in \mathcal{T} \setminus \{n_z + 1\}$,

$$(y_{k,(t+1)} \geq y_{k,t}). \quad (3.12t)$$

4.3.6 Volume constraints

Given $0 < A_i^1 < A_i^2 < \dots < A_i^{k_i^+}$ and $0 < B_i^1 < B_i^2 < \dots < B_i^{k_i^-}$, for $k_a \in \{1, 2, \dots, k_i^+\}$

$$V_i^+ \geq \sum_{k=1}^{k_a-1} A_i^k \bar{u}_i^k + A_i^{k_a} \left(u_i - \sum_{k=1}^{k_a-1} \bar{u}_i^k \right), \quad (3.13q)$$

and for $k_b \in \{1, 2, \dots, k_i^-\}$

$$V_i^+ \geq - \sum_{k=1}^{k_b-1} B_i^k \underline{u}_i^k - B_i^{k_b} \left(u_i - \sum_{k=1}^{k_b-1} \underline{u}_i^k \right). \quad (3.13r)$$

4.3.7 Roadside stockpiling constraints

If the end-user prefers accuracy over speed, then for all $i \in \mathcal{S}$,

$$V_i^+ \leq Mb_i, \quad (3.14a)$$

$$V_i^+ \geq \delta_i^+ b_i, \quad (3.14b)$$

$$V_i^- \leq M(1 - b_i), \quad (3.14c)$$

otherwise, for all $i \in \mathcal{S}$,

$$V_i^+ \leq Mb_i, \quad (3.15a)$$

$$V_i^+ \geq \delta b_i, \quad (3.15b)$$

$$V_i^- \leq M(1 - b_i), \quad (3.15c)$$

$$V_i^- \geq \delta(1 - b_i). \quad (3.15d)$$

4.3.8 Gap constraints

For all $g \in \mathcal{G}, i \in \mathcal{S}_g$,

$$h_{g,i} - a_{g,3} \frac{(s_{g,i+1})^2 + s_{g,i+1}s_{g,i} + (s_{g,i})^2}{3} - a_{g,2} \frac{s_{g,i+1} + s_{g,i}}{2} - a_{g,1} = u_{\neq(g,i)}. \quad (3.16)$$

4.3.9 Slope constraints

For all $g \in \mathcal{G}, i \in \{1, n_g\}$,

$$L \leq a_{g,2} + 2a_{g,3}s_{g,i} \leq U. \quad (3.17)$$

4.3.10 Smoothness constraints

For all $g \in \mathcal{G} \setminus \{1\}$,

$$a_{g,1} + a_{g,2}s_{g,1} + a_{g,3}s_{g,1}^2 = a_{(g-1),1} + a_{(g-1),2}s_{g,1} + a_{(g-1),3}s_{g,1}^2, \quad (3.18a)$$

$$a_{g,2} + 2a_{g,3}s_{g,1} = a_{(g-1),2} + 2a_{(g-1),3}s_{g,1}. \quad (3.18b)$$

4.3.11 Fixed point constraints

$$a_{1,1} + a_{1,2}s_{1,1} + a_{1,3}s_{1,1}^2 = y_A, \quad (3.19a)$$

$$a_{m,1} + a_{m,2}s_{m,n_m} + a_{m,3}s_{m,n_m}^2 = y_B, \quad (3.19b)$$

4.3. Model summary

for all $H_{g,i} \in \mathcal{H}$

$$a_{g,1} + a_{g,2}s_{g,i} + a_{g,3}s_{g,i}^2 = H_{g,i}, \quad (3.19c)$$

and

$$\left(a_{1,2} + 2a_{1,3}s_{1,1} = H'_A \right), \quad (3.19d)$$

$$\left(a_{m,2} + 2a_{m,3}s_{m,n_m} = H'_B \right). \quad (3.19e)$$

4.3.12 Bounds

For all $i \in \mathcal{N}, t \in \mathcal{T}$,

$$f_{i,i-1,t}^r \geq 0, \quad (4.10)$$

$$f_{i,i+1,t}^r \geq 0, \quad (4.11)$$

$$f_{i-1,i,t}^l \geq 0, \quad (4.12)$$

$$f_{i+1,i,t}^l \geq 0, \quad (4.13)$$

$$f_{i,i-1,t}^u \geq 0, \quad (4.14)$$

$$f_{i,i+1,t}^u \geq 0, \quad (4.15)$$

for all $j \in \mathcal{B}, t \in \mathcal{T}$,

$$f_{j,\vartheta(j)-1,t}^b \geq 0, \quad (4.16)$$

$$f_{j,\vartheta(j)+1,t}^b \geq 0, \quad (4.17)$$

for all $k \in \mathcal{W}, t \in \mathcal{T}$

$$f_{\varphi(k)-1,k,t}^w \geq 0, \quad (4.18)$$

$$f_{\varphi(k)+1,k,t}^w \geq 0, \quad (4.19)$$

for all $k \in \mathcal{I}, t \in \mathcal{T}$,

$$y_{kt} \in \{0, 1\}, \quad (3.20b)$$

for all $i \in \mathcal{S}$,

$$-h_i^- \leq u_i \leq h_i^+, \quad (3.20c)$$

$$0 \leq V_i^+ \leq M_i^+, \quad (3.20d)$$

$$0 \leq V_i^- \leq M_i^-, \quad (3.20e)$$

and for all $g \in \mathcal{G}, k \in \{1, 2, 3\}$,

$$a_{g,k} \in \mathbb{R}. \quad (3.20f)$$

4.4 Model evaluation

Proposition 4.4. *The MILP of Section 4.3 has $\mathcal{O}(nn_z)$ continuous variables.*

Proof. The MILP has n variables for excavation volume, n variables for embankment volume, $(2n+2)(n_z+1)$ transit flow variables, $4n(n_z+1)$ variables for load and unload flow for sections, $4n_b(n_z+1)$ borrow flow variables in the worst case, $4n_w(n_z+1)$ waste flow variables in the worst case, $3m$ variables for the spline coefficients, n variables for section height. So, the total number of continuous variables is

$$\begin{aligned}
 & n + n + (2n+2)(n_z+1) + 4n(n_z+1) + 4n_b(n_z+1) \\
 & \quad + 4n_w(n_z+1) + 3m + n \\
 = & 3n + 3m + (6n+2+4n_b+4n_w)(n_z+1) \\
 \leq & 6n + (6n+n+4n+4n)(n_z+n_z) \\
 = & 6n + 30nn_z
 \end{aligned}$$

since $n \geq n_b$, $n \geq n_w$, and $n \geq m$ and assuming $n \geq 2$ and $n_z \geq 1$. □

Proposition 4.5. *The MILP of Section 4.3 has $\mathcal{O}(n_z^2 + n)$ binary variables.*

Proof. The total number of binary variables is $n_z(n_z+1) + n \leq 2n_z^2 + n$ assuming $n_z \geq 1$. □

Proposition 4.6. *The MILP of Section 4.3 has $\mathcal{O}(nn_z^3)$ constraints.*

Proof. The number of constraints created by different equations of this model are shown in Table 4.1. The total number of constraints is $\mathcal{O}(nn_z^3)$. □

4.4. Model evaluation

Equation	Number of Constraints (worst case)
(4.2a),(4.2b)	$2n(n_z + 1)$
(4.3a),(4.3b)	$2n$
(4.4a),(4.4b)	$n_b + n_w$
From (4.5m) to (4.5x)	$8n_z(n_z + 1) + 2(n_b + n_w)(n_z + 1)$
(4.6a), (4.6b)	$2(n - 1)(n_z + 1)\frac{n_z(n_z - 1)}{2}$
(4.6c), (4.6d)	$2n_z(n - 1)(n_z + 1)$
(4.6e), (4.6f)	$2n_z(n - 1)(n_z + 1)$
From (4.6g) to (4.6l)	$2(n_b - 2)(n_z + 1)\frac{n_z(n_z - 1)}{2} + 4n_z(n_b - 2)(n_z + 1)$
From (4.6m) to (4.6r)	$2(n_w - 2)(n_z + 1)\frac{n_z(n_z - 1)}{2} + 4n_z(n_w - 2)(n_z + 1)$
(4.9a), (4.9b)	$2n(n_z + 1)$
(3.12s)	n_z
(3.12t)	n_z^2
(3.13q), (3.13r)	$\sum_{i \in \mathcal{S}} (k_i^+ + k_i^-)$
From (3.14a) to (3.14c)	$3n$
From (3.14a) to (3.14c)	$4n$
(3.16)	n
(3.17)	$2m$
(3.18a), (3.18a)	$2(m - 1)$
From (3.19a) to (3.19e)	$4 + \mathcal{H} $

Table 4.1: Number of constraints created by different equations of the quasi-network-flow model

Chapter 5

Numerical Results

However beautiful the strategy, you should occasionally look at the results.

Winston Churchill (1874 - 1965)

This thesis explains two novel models of vertical alignment, namely a basic MILP model and a quasi-network-flow model. The basic MILP model is the composition of Moreb's vertical alignment model [Mor09] with Koch and Lucet's improvements [KL10] and the earthwork block removal model of Hare, Koch, and Lucet [HKL11]. In addition, the basic MILP model introduces a novel way of formulating the side-slopes of the roads. So, for this model, we seek the answers to the following questions.

- How does the basic MILP model of vertical alignment performs in terms of robustness and accuracy compared to Moreb's model for vertical alignment?
- Can we numerically analyze the behavior of the design parameters k_i^+ , k_i^- , and n_g , which we believe to be critical to the model, and suggest a good values for these parameters?

The rationalization for finding a new model is that the basic MILP model starts performing badly as the number of sections increases, since the number of continuous variables is quadratically related to the number of sections. We therefore introduce the quasi-network-flow model. So, for this model, we want to ask the following questions.

- How fast can it solve the problems compared to the basic MILP model?
- Can it maintain the same objective value as the basic MILP model?

In this chapter, we explore these questions numerically.

5.1. Experimental equipments

Table 5.1: The basic problem set showing the roads, their fixed length for all test problems, and their fixed section length for most of the test problems

Road	Length (km)	Section Length (m)	Number of Sections
A	1	20	50
B	5	100	50
C	2	20	100
D	3	20	150
E	15	100	150
F	20	100	200
G	9	20	450

5.1 Experimental equipments

The experiments in this chapter were performed in a Dell OptiPlex 980 workstation with an Intel(R) Core(TM) i7-860 2.80GHz (6 cores) processor, a 16GB of Random Access Memory (RAM), and a 64-bit Windows 7 Enterprise operating system. An academic edition of the IBM ILOG CPLEX Optimizer 12.2 (<http://www.cplex.com>) was used to solve the problems. CPLEX is a high performance MILP solver that can take advantage of the parallel architecture of the processor. The models were programmed in the basic C++ programming language, which was chosen because of its faster execution, platform independence, and easier integration with the CPLEX Optimizer. Microsoft Visual Studio Professional 2008 was used as the programming platform.

5.2 Basic problem set and parameters

For our experiments, we used 7 distinct road samples, which are provided by our industrial partner. The data for these roads were generated by their software, which is used by government and non-government road engineering industries of many countries. We denote these roads with letters from A to G. We generated our problem set by changing different parameters of these roads. However, these roads have fixed section length and number of sections for all the problems. Table 5.1 shows the basic information of these roads. Moreover, all the experiments have the following parameters,

- Relative MIP gap tolerance: 1%,
- LP feasibility tolerance: 10^{-06} ,
- Maximum run time: 2 hrs for each core of the processor totaling 12 hrs,

- Upper bound for slope change: 0.5,
- Lower bound for slope change: -0.5 ,
- Maximum fill height: $10m$,
- Maximum cut height: $10m$.

Other solver and model parameters remain at the default settings.

5.3 Test configuration 1

In this test configuration, we compared the basic MILP model with Moreb's vertical alignment model in terms of robustness and accuracy. Moreb's vertical alignment model uses rectangular approximation for calculating the volume of a section as written in Equation (2.9f). Since the height of a section is a design variable, the appropriate approximation of the volume is highly dependent on the choice of area of a section for Moreb's model. But, as the height changes, so does the appropriate value of area. In other words, Moreb's vertical alignment model can be very accurate if we can predict the height for each section before solving the model. However, if the prediction of heights is far from the actual calculated heights, then Moreb's model can produce very bad results. This behavior makes the model very unstable. To deal with this issue, two strategies are used currently.

The most common strategy is to assume that the output road profile will be very close to the input ground profile. Of course, assuming this will bias the optimizer to choose a solution close to the ground profile. While this strategy may perform well for some examples, for other examples it may perform poorly. We call this approach the Near Ground Moreb (NGM) approach.

The second approach is to predict the height to be the half way to the maximum height for both cuts and fills. The quality of this approach also depends on the initial guess, but since the initial guess is the average of the extreme guesses, this approach should not perform really bad, at the same time, we should not expect very good solution from it. We call this approach Average Moreb (AM) approach.

Unlike Moreb's model, the basic MILP model for vertical alignment does not depend on any initial guess. From Figures 3.3 and 3.4, we see that as the number of entries in the lookup table, k_i^+ (for a cut section i) and k_i^- (for a fill section i) increases, the approximate volume becomes the actual volume. In addition to comparing with the two variations of Moreb's model, we also show the change in accuracy with respect to the change in the values for k_i^+ and k_i^- .

5.3. Test configuration 1

Table 5.2: Problem (R - k , where k is the number of cut and fill entries in the lookup table, AM or NGM for road R) set summary for Test configuration 1

Road	Problems
A	A-400, A-200, A-100, A-50, A-40, A-20, A-10, A-5, A-2, A-AM, A-NGM
B	B-400, B-200, B-100, B-50, B-40, B-20, B-10, B-5, B-2, B-AM, B-NGM
C	C-400, C-200, C-100, C-50, C-40, C-20, C-10, C-5, C-2, C-AM, C-NGM
D	D-400, D-200, D-100, D-50, D-40, D-20, D-10, D-5, D-2, D-AM, D-NGM
E	E-400, E-200, E-100, E-50, E-40, E-20, E-10, E-5, E-2, E-AM, E-NGM
F	F-400, F-200, F-100, F-50, F-40, F-20, F-10, F-5, F-2, F-AM, F-NGM
G	G-400, G-200, G-100, G-50, G-40, G-20, G-10, G-5, G-2, G-AM, G-NGM

5.3.1 Reference solution

The reference solution for the experiments should be models with sufficient large values for k_i^+ and k_i^- for all sections i . For the experiments of this test configuration, we use the basic MILP model with $k_i^+ = k_i^- = 400$ for all section i , as the reference solution.

5.3.2 Problem set and parameters

For each of the roads of Table 5.1, we generated 10 problems for 10 different values of k_i^+ and k_i^- , as well as 2 problems for the AM and NGM approaches. It should be noted that a basic MILP model with $k_i^+ = k_i^- = 1$ is the same as a model generated by the Average Moreb (AM) approach. Therefore, for this test configuration, we generated 11 problems for each of the 7 roads totaling 77 problems with the naming convention R - k , where R is a road from Table 5.1, for the basic MILP model k represents the value for k_i^+ and k_i^- for all sections i , and for Moreb's model k represents AM or NGM. Table 5.2 summarizes the problem set. For this test configuration, the other parameters are,

- number of sections per spline segment: $n_g = 2$ for all $g \in \mathcal{G}$,
- number of blocks: $n_z = 0$,
- number of access roads: $n_r = 1$.

We add an access road to the first section for each of the problems, with an infinite borrow pit and an infinite waste pit to ensure feasibility.

5.3.3 Experiment procedure

We used the following three-steps procedure for each of the problems in Table 5.2 to calculate the accuracy.

- In step 1, we solve the given model and save the output vertical alignment and the optimal cost. We denote the optimal cost of this step with $C_{R,k}$.
- In step 2, we generate a model with the same parameters as the given model except $k_i^+ = k_i^- = 400$, then we input the vertical alignment from step 1 to this model, and solve only for the earthwork operations. The optimal cost of this step is the optimal cost with the corrected volume for the vertical alignment of step 1. We denote this cost with $C_{R,k}^{400}$.
- In step 3, we use the formulas $\epsilon_{R,k} = \frac{C_{R,k} - C_{R,k}^{400}}{C_{R,k}^{400}} \times 100\%$ to get the percent relative error and $|\epsilon_{R,k}| = \frac{|C_{R,k} - C_{R,k}^{400}|}{C_{R,k}^{400}} \times 100\%$ to get the percent absolute relative error.

We normalized the solution time with respect to the solution times with the model with $k_i^+ = k_i^- = 400$ to accommodate all the roads in the same graph. If $t_{R,k}$ is the required time for Road R with $k_i^+ = k_i^- = k$, then the normalized time is defined as

$$\bar{t}_{R,k} = \frac{t_{R,k}}{t_{R,400}}. \quad (5.1)$$

5.3.4 Results

The raw results of the problem configuration are shown in Tables B.1, B.2, and B.3. A careful observation of the results reveal that depending on the values of k_i^+ and k_i^- the model can introduce significant errors. But, we wanted to know how the accuracy of the problems changes as the values of k_i^+ and k_i^- change, at the same time, we wanted to define a range for k_i^+ and k_i^- in which the solutions will be reasonably accurate. For this purpose, we define 3 ranges of accuracies of the solutions, i.e.,

- $|\epsilon_{R,k}| \leq 2\%$: accurate solution, since the relative MIP gap tolerance is 1%,
- $2\% < |\epsilon_{R,k}| \leq 5\%$: acceptable solution, since up to 5% error is acceptable to our users, and
- $|\epsilon_{R,k}| > 5\%$: unacceptable solution.

5.3. Test configuration 1

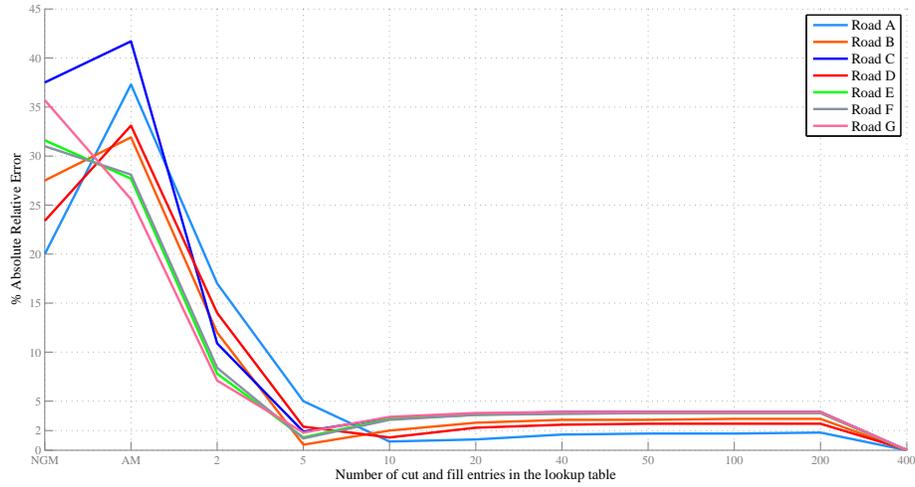


Figure 5.1: Percent absolute relative error for NGM and AM approaches of Moreb's model and the basic MILP model with different numbers of cut and fill entries in the lookup table

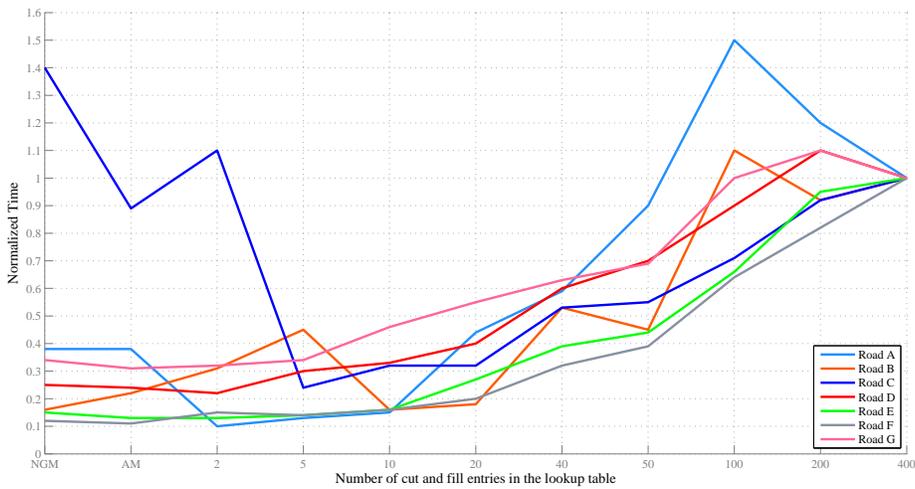


Figure 5.2: Normalized time for NGM and AM approaches of Moreb's model and the basic MILP model with different numbers of cut and fill entries in the lookup table

In Figure 5.1, we plot the percent absolute relative error for both NGM and AM approaches of Moreb’s model and the basic MILP model with different numbers of cut and fill entries in the lookup table. From the plot, we see that for all the roads both approaches for Moreb’s model generate errors that are unacceptable to our users. We also see that the error drops very quickly as we increase the number of entries in the lookup table and for all the roads the error stays in the acceptable range if we use more than 5 entries in the lookup table. We conclude that given a reasonable number of entries in the lookup table, the basic MILP model is robust and accurate.

Similar to Figure 5.1, we plot the timing data in Figure 5.2. We see that both Moreb’s approaches require almost the same amount of time, which is expected since both of this approaches have the same number of variables and constraints. For each additional entries of the lookup table, $2n$ additional constraints are required. So, the timing should increase as we increase the number of entries in the lookup table.

5.4 Test configuration 2

In this test configuration, we analyzed the behavior of another design parameter of the basic MILP model called the number of sections per spline segment (n_g for all $g \in \mathcal{G}$), which we believed to be critical for the model.

5.4.1 Reference solution

Theoretically, the best value for n_g , for all $g \in \mathcal{G}$, is 1, since it gives the best objective value. The reason is $n_g = 1$, for all $g \in \mathcal{G}$, gives the model the most flexibility to move the road profile up and down. We set it as the reference solution.

5.4.2 Problem set and parameters

For each of the roads of Table 5.1, we generated problems for different values of n_g . When generating these problems, we made sure that n was divisible by n_g so that all the segments have the same number of sections. For this test configuration, we generated 51 problems with the naming convention $R-n_g$, where R is a road from Table 5.1, and n_g is the number of sections per segment, that are shown in Table 5.3. It should be noted that $n_g > 10$ is rarely used, but we created some problems with higher n_g values for demonstration purpose. The other parameters for this test configuration are,

- number of entries in the lookup table: $k_i^+ = k_i^- = 200$ for all i ,

5.4. Test configuration 2

Table 5.3: Problem (R - n_g , where n_g is the number of sections per segment for road R) set summary for Test configuration 2

Road	Problems
A	A-1, A-2, A-5, A-10, A-25
B	B-1, B-2, B-5, B-10, B-25
C	C-1, C-2, C-4, C-5, C-10, C-20, C-25
D	D-1, D-2, D-3, D-4, D-5, D-6, D-10, D-15, D-25
E	E-1, E-2, E-3, E-5, E-6, E-10, E-15, E-25
F	F-1, F-2, F-4, F-5, F-8, F-10, F-20, F-25
G	G-1, G-2, G-3, G-5, G-6, G-9, G-10, G-15, G-18, G-25

- number of blocks: $n_z = 0$,
- number of access roads: $n_r = 1$.

5.4.3 Experiment procedure

From Test configuration 1, we see that models with $k_i^+ = k_i^- = 200$ will produce accurate enough solutions for practical use. So, for this test configuration we assume that all the solutions are accurate. Assuming that we want to know how the cost and timing changes with respect to the parameter n_g . We normalized the solutions with respect to the reference solution to accommodate all the roads in the same graph. If C_{R,n_g} is the optimal cost for Road R with n_g sections per segment, then the normalized cost is defined as,

$$\bar{C}_{R,n_g} = \frac{C_{R,n_g}}{C_{R,1}}. \quad (5.2)$$

If t_{R,n_g} is the required time for Road R with n_g sections per segment, then the normalized time is defined as,

$$\bar{t}_{R,n_g} = \frac{t_{R,n_g}}{t_{R,1}}. \quad (5.3)$$

5.4.4 Results

The raw results of this test configuration, as well as, the calculated normalized cost and timing are shown in Table B.4. In Figure 5.3, we plot the change of normalized cost with respect to the number of sections per segment for each of the roads. As expected, we see that the cost continues to increase as the value of n_g

5.4. Test configuration 2

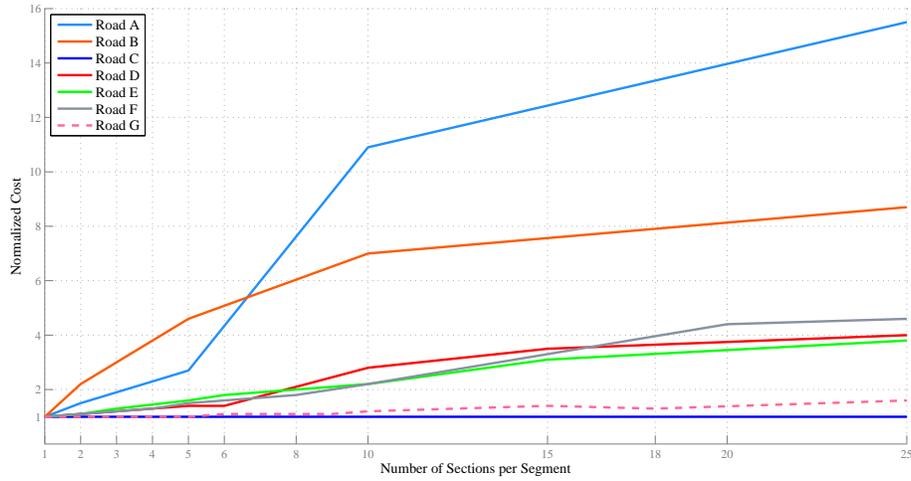


Figure 5.3: The change of normalized cost with respect to the number of sections per segment for each of the roads

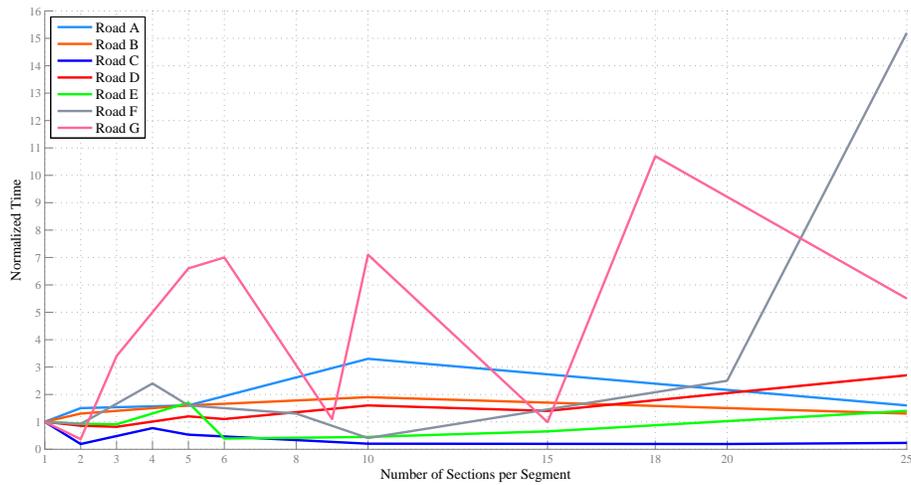


Figure 5.4: The change of normalized time with respect to the number of sections per segment for each of the roads

increases, whereas the time behaves randomly. We conclude that the number of sections per spline segment should not exceed 2.

5.5 Test configuration 3

In this test configuration, we compare the performance of the quasi-network-flow model with respect to the basic MILP model. Since the quasi-network-flow model has $\mathcal{O}(nn_z)$ continuous variables compared to $\mathcal{O}(n^2n_z)$ continuous variables of the basic MILP model, we are expecting a significant improvement in timing. At the same time, we are expecting that both models will report the same optimal cost for the same problem.

5.5.1 Reference solution

Since we are comparing the performance of the quasi-network-flow model with respect to the basic MILP model, our reference solution is the basic MILP model for both timing and optimal cost.

5.5.2 Problem set and parameters

We add (5, 2), (5, 4), (10, 2), and (10, 4) number of (blocks, access roads) to each of the problems in Table 5.2 generating 280 problems with naming convention $R-k, n_z, n_r$, where R is a road, $k = k_i^+ = k_i^-$, n_z is the number of blocks, and n_r is the number of access roads. Table 5.4 shows the problem set. For the problems, the number of sections per spline segment is $n_g = 2$ for all $g \in \mathcal{G}$.

5.5.3 Experiment procedure

We solve the same problem with both models. For a problem $R-k, n_z, n_r$, we denote the optimal cost of the basic MILP model with C_{R,k,n_z,n_r}^b and that of the quasi-network-flow model with C_{R,k,n_z,n_r}^n . So, the percent difference in optimal cost is

$$\delta_{R,k,n_z,n_r} = \frac{C_{R,k,n_z,n_r}^n - C_{R,k,n_z,n_r}^b}{C_{R,k,n_z,n_r}^b} \times 100\%. \quad (5.4)$$

Since the relative MIP gap tolerance is 1%, a δ_{R,k,n_z,n_r} value between -2% and 2% implies that both models give the statistically equivalent optimal cost. For a problem $R-k, n_z, n_r$, we denote the timing of the basic MILP model with t_{R,k,n_z,n_r}^b and that of the quasi-network-flow model with t_{R,k,n_z,n_r}^n . Since theoretically $t_{R,k,n_z,n_r}^n \leq$

5.5. Test configuration 3

Table 5.4: Problem ($R-k, n_z, n_r$, where $k = k_i^+ = k_i^-$, n_z is the number of blocks, and n_r is the number of access roads for road R) set summary for Test configuration 3

Road	Problems
A	A-400,5,2; A-200,5,2; A-100,5,2; A-50,5,2; A-40,5,2; A-20,5,2; A-10,5,2; A-5,5,2; A-2,5,2; A-1,5,2; A-400,10,2; A-200,10,2; A-100,10,2; A-50,10,2; A-40,10,2; A-20,10,2; A-10,10,2; A-5,10,2; A-2,10,2; A-1,10,2; A-400,5,4; A-200,5,4; A-100,5,4; A-50,5,4; A-40,5,4; A-20,5,4; A-10,5,4; A-5,5,4; A-2,5,4; A-1,5,4; A-400,10,4; A-200,10,4; A-100,10,4; A-50,10,4; A-40,10,4; A-20,10,4; A-10,10,4; A-5,10,4; A-2,10,4; A-1,10,4;
B	B-400,5,2; B-200,5,2; B-100,5,2; B-50,5,2; B-40,5,2; B-20,5,2; B-10,5,2; B-5,5,2; B-2,5,2; B-1,5,2; B-400,10,2; B-200,10,2; B-100,10,2; B-50,10,2; B-40,10,2; B-20,10,2; B-10,10,2; B-5,10,2; B-2,10,2; B-1,10,2; B-400,5,4; B-200,5,4; B-100,5,4; B-50,5,4; B-40,5,4; B-20,5,4; B-10,5,4; B-5,5,4; B-2,5,4; B-1,5,4; B-400,10,4; B-200,10,4; B-100,10,4; B-50,10,4; B-40,10,4; B-20,10,4; B-10,10,4; B-5,10,4; B-2,10,4; B-1,10,4;
C	C-400,5,2; C-200,5,2; C-100,5,2; C-50,5,2; C-40,5,2; C-20,5,2; C-10,5,2; C-5,5,2; C-2,5,2; C-1,5,2; C-400,10,2; C-200,10,2; C-100,10,2; C-50,10,2; C-40,10,2; C-20,10,2; C-10,10,2; C-5,10,2; C-2,10,2; C-1,10,2; C-400,5,4; C-200,5,4; C-100,5,4; C-50,5,4; C-40,5,4; C-20,5,4; C-10,5,4; C-5,5,4; C-2,5,4; C-1,5,4; C-400,10,4; C-200,10,4; C-100,10,4; C-50,10,4; C-40,10,4; C-20,10,4; C-10,10,4; C-5,10,4; C-2,10,4; C-1,10,4;
D	D-400,5,2; D-200,5,2; D-100,5,2; D-50,5,2; D-40,5,2; D-20,5,2; D-10,5,2; D-5,5,2; D-2,5,2; D-1,5,2; D-400,10,2; D-200,10,2; D-100,10,2; D-50,10,2; D-40,10,2; D-20,10,2; D-10,10,2; D-5,10,2; D-2,10,2; D-1,10,2; D-400,5,4; D-200,5,4; D-100,5,4; D-50,5,4; D-40,5,4; D-20,5,4; D-10,5,4; D-5,5,4; D-2,5,4; D-1,5,4; D-400,10,4; D-200,10,4; D-100,10,4; D-50,10,4; D-40,10,4; D-20,10,4; D-10,10,4; D-5,10,4; D-2,10,4; D-1,10,4;
E	E-400,5,2; E-200,5,2; E-100,5,2; E-50,5,2; E-40,5,2; E-20,5,2; E-10,5,2; E-5,5,2; E-2,5,2; E-1,5,2; E-400,10,2; E-200,10,2; E-100,10,2; E-50,10,2; E-40,10,2; E-20,10,2; E-10,10,2; E-5,10,2; E-2,10,2; E-1,10,2; E-400,5,4; E-200,5,4; E-100,5,4; E-50,5,4; E-40,5,4; E-20,5,4; E-10,5,4; E-5,5,4; E-2,5,4; E-1,5,4; E-400,10,4; E-200,10,4; E-100,10,4; E-50,10,4; E-40,10,4; E-20,10,4; E-10,10,4; E-5,10,4; E-2,10,4; E-1,10,4;
F	F-400,5,2; F-200,5,2; F-100,5,2; F-50,5,2; F-40,5,2; F-20,5,2; F-10,5,2; F-5,5,2; F-2,5,2; F-1,5,2; F-400,10,2; F-200,10,2; F-100,10,2; F-50,10,2; F-40,10,2; F-20,10,2; F-10,10,2; F-5,10,2; F-2,10,2; F-1,10,2; F-400,5,4; F-200,5,4; F-100,5,4; F-50,5,4; F-40,5,4; F-20,5,4; F-10,5,4; F-5,5,4; F-2,5,4; F-1,5,4; F-400,10,4; F-200,10,4; F-100,10,4; F-50,10,4; F-40,10,4; F-20,10,4; F-10,10,4; F-5,10,4; F-2,10,4; F-1,10,4;
G	G-400,5,2; G-200,5,2; G-100,5,2; G-50,5,2; G-40,5,2; G-20,5,2; G-10,5,2; G-5,5,2; G-2,5,2; G-1,5,2; G-400,10,2; G-200,10,2; G-100,10,2; G-50,10,2; G-40,10,2; G-20,10,2; G-10,10,2; G-5,10,2; G-2,10,2; G-1,10,2; G-400,5,4; G-200,5,4; G-100,5,4; G-50,5,4; G-40,5,4; G-20,5,4; G-10,5,4; G-5,5,4; G-2,5,4; G-1,5,4; G-400,10,4; G-200,10,4; G-100,10,4; G-50,10,4; G-40,10,4; G-20,10,4; G-10,10,4; G-5,10,4; G-2,10,4; G-1,10,4;

5.5. Test configuration 3

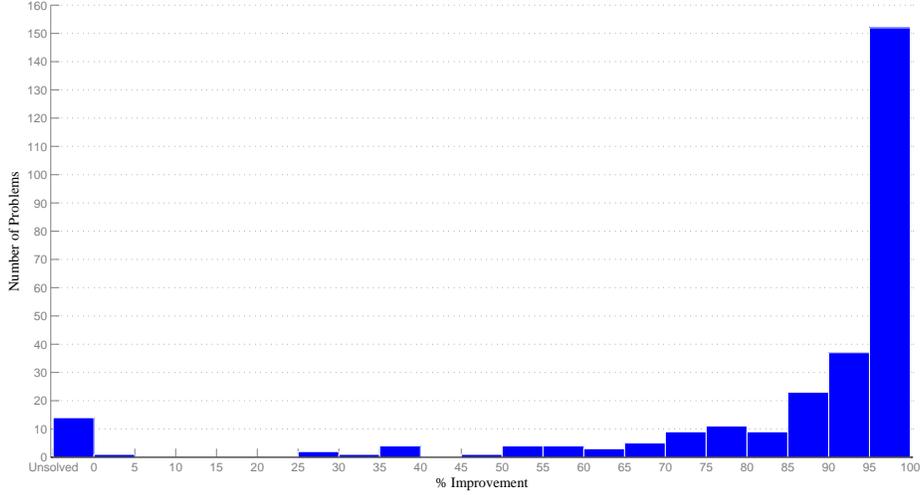


Figure 5.5: A histogram showing the number of problems solved within different improvement range

t_{R,k,n_z,n_r}^b , we define the percent improvement in timing as

$$\eta_{R,k,n_z,n_r} = \frac{t_{R,k,n_z,n_r}^b - t_{R,k,n_z,n_r}^n}{t_{R,k,n_z,n_r}^b} \times 100\%. \quad (5.5)$$

We expect $0 \leq \eta_{R,k,n_z,n_r} \leq 100$ with a larger value implies larger improvement. A negative value for η_{R,k,n_z,n_r} implies a negative improvement.

5.5.4 Results

The raw results of this test configuration are shown in Tables B.5, B.6, B.7, B.8, B.9, B.10, B.11, and B.12. In the tables, a problem name ending with an * denotes that the basic MILP model either terminated in the middle of the program execution because of insufficient memory i.e., no timing was recorded, or the solver could not solve the model within the given time limit. However, the tables records the best feasible solution value before it ran out of time or memory. In Figure 5.5, we show a histogram of the number of problems solved within different range of improvement in timing, as well as, the number of unsolved problems. From the data, we see that

- the basic MILP model cannot solve 14 problems in the given time limit

and computer memory, whereas the quasi-network-flow model solves all the problems,

- the quasi-network-flow model not only solves all the problems faster than the basic MILP model but also the improvement is significant for most of the problems, and
- the problems that both model can solve are within the 2% of each other i.e., both models give statistically equivalent cost for all the problems.

5.6 Summary of the results

We can summarize the results of this chapter as follows.

For all i , the parameters k_i^+ and k_i^- are critical for the accuracy of the model, and the recommended value is 5 or more. The timing can change from a factor of 0.1 to 1.5, but if the parameter value is less than 20, the solution time remains relatively unchanged.

For all $g \in \mathcal{G}$, the parameters n_g are critical for finding the minimum cost, and the recommended value is 1. These parameters have random effects on solution time.

The quasi-network-flow model is a noticeable improvement over the basic MILP model in regards to solution time.

From our observation of the resource monitor of the computer, we suggest the users to run the basic MILP model on a work station with a larger computer memory and the quasi-network-flow model on a work station with higher computing power.

Chapter 6

Conclusion and Future Work

The important thing is not to stop questioning, curiosity has its own reason for existing.

Albert Einstein (1879 - 1955)

6.1 Conclusion

Since both Moreb's vertical alignment model [Mor09] and Hare, Koch, and Lucet's earthwork model with blocks [HKL11] are capable of satisfying most of the requirements of our clients, we have combined both ideas in a single model. The resulting model is the first model in this research area that can compute the vertical alignment considering blocks and access roads.

We have further looked into possible improvements for this model. We have found that the model was generating substantial errors in the volume approximation from not considering the side-slopes of the roads. So, we have developed a novel way of implementing side-slopes using lookup table without significantly increasing the time complexity of the model.

Finally, we have come up with a novel way of reducing the number of earth movement variables using the ideas from network flow algorithms. Similar idea has been implemented for the earthwork optimization problem in [HHL12] and extended for vertical alignment in this thesis. We call the formulation a quasi-network-flow model of vertical alignment.

6.2 Future work

6.2.1 The dynamic block extension

The described models assume that no roadside stockpiling will occur. One possible extension is to change the models to allow roadside stockpiling. When too much roadside stockpiling occurs in a section, a temporary block may be created

that did not previously exist. This type of block is called a dynamic block. One of the important future direction is to handle dynamic blocks in the model.

6.2.2 The multi-material extension

In a multi-material model of a vertical alignment, the different types of materials found in earth, which vary in cost and quality, are taken into account. Considering multiple materials provides a more realistic solution, because the cost of the material sometimes depends on the types of materials in the earth, and often some parts of the road require a specific material type for engineering reasons.

In a multi-material model, every cut section has given volumes for all material types, which comes from the input ground profile, and every fill section has a user defined required volumes for all material types. We explored few types of multi-material models, which are described briefly in this section.

Exact multi-material model

This simple form of multi-material model sub-divide the earth movement variables for each material. The discussed vertical alignment models of this thesis are easily extendable to this form. Moreb discussed this approach in [Mor09]. This approach gives impractical solutions, since it requires a considerable use of the borrow and waste pits for feasibility. This approach is not acceptable in the road design field.

Quality based multi-material model

This approach categorizes the materials into different levels of quality, and a higher quality material can be used in place of a lower quality material. This approach is more practical. There are two variations of this approach.

- In the first variation, there is no minimum required material quality for a section. A section can accept any material at any amount, as long as their quality is better than or same as the required quality.
- In the second variation, a minimum amount of each quality for each section is specified. After the minimum amount is provided, the section can accept materials of the same or better quality.

Rule based multi-material model

This is the most generic form of a multi-material model. Instead of defining the quality of the materials, this method defines some rules that must be followed

for cutting and filling. The quality based multi-material model is an example of the rule based multi-material model, where the rule that a higher quality material can be used in place of a lower quality material with or without a minimum amount is enforced. Different rules requires different modeling techniques for this approach.

For a detailed explanation of the multi-material extension we refer to [BRLH11], where we also show an earthwork implementation of the quality based multi-material model with a required minimum amount. However, extending the earthwork multi-material model to a vertical alignment model is not trivial and is left for future research.

6.2.3 Other areas of improvement

Future research can be done on finding the optimal section length, reducing the number of binary variables, considering the cost of land, and incorporating retaining walls.

6.2.4 Horizontal alignment

The next step of the research is to model the horizontal alignment and simultaneously optimize the horizontal and vertical alignment of a road. Unlike vertical alignment, the horizontal alignment formulation can have a non-convex or discontinuous objective function. The modeling also involves considering political, socioeconomic, and environmental issues. Discrete modeling techniques like network optimization or dynamic programming seems promising at this stage of the research.

Bibliography

- [AAS04] AASHTO, editor. *A policy on geometric design of highways and streets*. AASHTO, 5 edition, 2004. pages: 3 and 4.
- [AC73] G C Athanassoulis and V Calogero. Optimal location of a new highway from a to b, a computer technique for route planning. In *PTRC Seminar Proceedings on Cost Models & Optimization in Highway*, 1973. page: 7.
- [Aka03] A. E. Akay. *Minimizing Total Cost of Construction, Maintenance, and Transportation Costs with Computer-Aided Forest Road Design*. PhD thesis, Oregon State University, Corvallis, Oregon, 2003. pages: 1 and 6.
- [AM03] M. S. Aljohani and A. A. Moreb. Roadway profile modeled by polynomials to minimize earthwork cost. In *WSEAS Transactions on Mathematics*, pages 210–213, 2003. page: 11.
- [Aru05] Kazuhiro Aruga. Tabu search optimization of horizontal and vertical alignments of forest roads. *Journal of Forest Research*, 10:275–284, 2005. 10.1007/s10310-004-0136-5. page: 6.
- [ATSM06] Kazuhiro Aruga, Toshiaki Tasaka, John Sessions, and S. Miyata. Tabu search optimization of forest road alignments combined with shortest paths and cubic splines. *Croatian Journal of Forest Engineering*, 27, 2006. page: 6.
- [Bak72] A. B. Baker. *The design and phasing of horizontal and vertical alignments: Program JANUS*. Transport and Road Research Laboratory, 1972. page: 8.
- [BD62] Richard Ernest Bellman and Stuart E. Dreyfus. *Applied Dynamic programming*. Princeton University Press, 1962. page: 9.
- [Bel57] Richard Ernest Bellman. *Dynamic Programming*. Princeton University Press, 1957. pages: 7 and 9.

Bibliography

- [Bis09] Johannes Bisschop. *AIMMS Optimization Modeling*. Paragon Decision Technology, 2009. page: 38.
- [BRLH11] Kasra Bigdeli, Faisal Rahman, Yves Lucet, and Warren Hare. A multi-material multi-block model for earthwork optimization. Technical report, NSERC ENGAGE and undisclosed, 2011. page: 95.
- [CGF89] E P Chew, C J Goh, and T F Fwa. Simultaneous optimization of horizontal and vertical alignments for highways. *Transportation Research Part B: Methodological*, 23(5):315–329, 1989. page: 6.
- [CL06] Juey-Fu Cheng and Yusin Lee. Model for three-dimensional highway alignment. *Journal of Transportation Engineering*, 132(12):913–920, 2006. page: 6.
- [CLR01] Thomas H. Cormen, Charles E. Leiserson, and Ronald Linn Rivest. *Introduction to Algorithms, Section 24.1: The Bellman Ford algorithm*. MIT Press and McGraw-Hill, 2nd edition, 2001. page: 7.
- [Dij59] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959. 10.1007/BF01386390. page: 7.
- [DJ09] Coray Davis and Manoj K. Jha. Modeling the effects of socioeconomic factors in highway construction and expansion. *Journal of Transportation Engineering*, 135(12):990–998, 2009. page: 2.
- [Eas87] Said M Easa. Earthwork allocations with nonconstant unit costs. *J Constr Eng M ASCE*, 113(1):34–50, March 1987. page: 10.
- [Eas88a] Said M. Easa. Earthwork allocations with linear unit costs. *Journal of Construction Engineering and Management*, 114(4):641–655, 1988. page: 10.
- [Eas88b] Said M. Easa. Selection of roadway grades that minimize earthwork cost using linear programming. *Transportation Research Part A: General*, 22(2):121 – 136, 1988. page: 10.
- [Fwa89] T. F. Fwa. Highway vertical alignment analysis by dynamic programming. *Transportation Research Record*, 1239:1–9, 1989. pages: 2, 3, 4, and 9.

Bibliography

- [GAA09] A. Burak Goktepe, Selim Altun, and Perviz Ahmedzade. Optimization of vertical alignment of highways utilizing discrete dynamic programming and weighted ground line. *Turkish J. Eng. Env. Sci.*, 33:105–116, 2009. page: 10.
- [GCF88] C.J. Goh, E.P. Chew, and T.F. Fwa. Discrete and continuous models for computation of optimal vertical highway alignment. *Transportation Research Part B: Methodological*, 22(6):399 – 409, 1988. pages: 2 and 9.
- [GH09] Nicholas J. GARBER and Lester A. HOEL. *Traffic and highway engineering. 4th ed.* Cengage Learning, Ohio, 2009. page: 11.
- [GLA05] A. Burak Goktepe, A.Hilmi Lav, and Selim Altun. Dynamic optimization algorithm for vertical alignment of highways. *Mathematical and Computational Applications*, 10(3):341–350, 2005. pages: 2, 4, and 9.
- [GLA09] A. Burak Goktepe, A.Hilmi Lav, and Selim Altun. Method for optimal vertical alignment of highways. In *Proceedings of the Institution of Civil Engineers (ICE) - Transport*, volume 162, pages 177 –188. Institution of Civil Engineers, 2009. page: 11.
- [GT88] C. J. Goh and K. L. Teo. Control parametrization: a unified approach to optimal control problems with general constraints. *Automatica*, 24:3–18, January 1988. page: 9.
- [Hay70] R W Hayman. Optimization of vertical alignment for highways through mathematical programming. *Highway Research Record*, pages 1–9, 1970. page: 8.
- [HBS68] B E Howard, Z Bramnick, and J F Shaw. Optimum curvature principle in highway routing. *Journal of the Highway Division*, 94:61–82, 1968. page: 7.
- [HHLM12] Donovan Hare, Warren Hare, Yves Lucet, and Sunny Mehrotra. An arc-flow model for earthwork operations. Technical report, (undisclosed), 2012. page: 93.
- [HKL11] Warren L. Hare, Valentin R. Koch, and Yves Lucet. Models and algorithms to improve earthwork operations in road design using mixed integer linear programming. *European Journal of Operational Research*, 215(2):470–480, 2011. pages: 11, 13, 24, 55, 60, 80, and 93.

Bibliography

- [Hog73] J. D. Hogan. Experience with optloc—optimum location of highways by computer. In *PTRC seminar proceedings on Cost Models and Optimization in Highways (Session L 10)*, London, 1973. page: 5.
- [Hua73] J. Puy Huarte. Opygar: Optimization and automatic design of highway profiles. In *PTRC Seminar Proceedings on Cost Models and Optimization in Highways, Session L13*, London, UK, 1973. page: 9.
- [Jha03] Manoj K. Jha. Criteria-based decision support system for selecting highway alignments. *Journal of Transportation Engineering*, 129(1):33–41, 2003. pages: 1 and 6.
- [JS03] Jyh-Cherng Jong and Paul Schonfeld. An evolutionary model for simultaneously optimizing three-dimensional highway alignments. *Transportation Research Part B: Methodological*, 37(2):107 – 128, 2003. page: 6.
- [JSB⁺10] Y. Ji, F. Seipp, A. Borrmann, S. Ruzika, and E. Rank. Mathematical modeling of earthwork optimization problems. In *Proc. of the International Conference on Computing in Civil and Building Engineering 2010*, pages 403–410, 2010. page: 11.
- [JSJ06] Manoj Kumar Jha, Paul Schonfeld, and J-C Jong. *Intelligent Road Design*, volume 19. WIT Press, 2006. pages: 2 and 7.
- [KJS05] Eungcheol Kim, Manoj Kumar Jha, and Bongsoo Son. Improving the computational efficiency of highway alignment optimization models through a stepwise genetic algorithms approach. *Transportation Research Part B: Methodological*, 39:339–360, 2005. page: 6.
- [KJSK07] Eungcheol Kim, Manoj K. Jha, Paul Schonfeld, and Hong Sok Kim. Highway alignment optimization incorporating bridges and tunnels. *Journal of Transportation Engineering*, 133(2):71–81, 2007. page: 6.
- [KL10] Valentin Koch and Yves Lucet. A note on: Spline technique for modeling roadway profile to minimize earthwork cost. *J. Ind. Manag. Optim.*, 6(2):393 – 400, May 2010. pages: 11, 13, 18, 21, and 80.
- [Koc10] Valentin Koch. Optimizing earthwork block removal in road construction. Master’s thesis, University of British Columbia Okanagan, Kelowna BC, Canada, 2010. pages: 17, 24, and 55.
- [KvT05] Jon Kleinberg and Éva Tardos. *Algorithm Design*. Addison-Wesley, 2005. pages: 103 and 104.

Bibliography

- [Lay93] M. G. Lay. *Ways of the world : a history of the world's roads and of the vehicles that used them*. Primavera Press, Sydney, 1993. page: 1.
- [LRFF62] Jr. Lester Randolph Ford and Delbert Ray Fulkerson. Flows in networks. *Princeton University Press*, 1962. page: 7.
- [MA04] Ahmad A. Moreb and Mohammad S. Aljohani. Quadratic representation for roadway profile that minimizes earthwork cost. *J Sys Sc Sys Eng*, 13(2):245–252, April 2004. pages: 11 and 13.
- [Min10a] Minister of Public Works and Government Services. Transportation in canada 2010 - addendum tables and figures. *Transport Canada*, Cat. No. T1-21/2010E-PDF:7–13, 2010. page: 1.
- [Min10b] Minister of Public Works and Government Services. Transportation in canada 2010 - an overview. *Transport Canada*, Cat. No. T1-21/2010E-PDF:7–13, 2010. page: 1.
- [MJ09] A. Maji and M. K. Jha. Multi-objective highway alignment optimization using a genetic algorithm. *J. Adv. Transp.*, 43(4):2042–3195, 2009. page: 6.
- [Mor96] Ahmad A. Moreb. Linear programming model for finding optimal roadway grades that minimize earthwork cost. *European Journal of Operational Research*, 93(1):148 – 154, 1996. page: 10.
- [Mor09] Ahmad A Moreb. Spline technique for modeling roadway profile to minimize earthwork cost. *Journal of Industrial and Management Optimization (JIMO)*, 5(2):275–283, 2009. pages: 11, 13, 18, 20, 80, 93, and 94.
- [MS81] R Mayer and R Stark. Earthmoving logistics. *Journal of the Construction Division*, 107(2):297–312, Jun 1981. page: 10.
- [Mur73] J. D. Murchland. Methods of vertical profile optimisation for an improvement to an existing road. In *PTRC Seminar Proceedings on Cost Models and Optimisation in Highways (session L12)*, London, 1973. page: 9.
- [Nan81] S. Nandgaonkar. Earthwork transportation allocations. *Journal of Construction Division*, 107:373–392, 1981. page: 10.
- [NEW76] A. J. Nicholson, D. G. Elms, and A. Williman. A variational approach to optimal route location. *Highway Engineers*, 23:22–25, 1976. page: 5.

Bibliography

- [OEC73] OECD. Optimisation of road alignment by the use of computers. Technical report, Organization of Economic Co-operation and Development, Paris, 1973. pages: 7 and 8.
- [OH82] Clarkson H. Oglesby and Russell G. Hicks. *Highway Engineering*. John Wiley, New York, 4th edition, 1982. page: 10.
- [Ont84] Ontario Ministry of Transportation and Communications. Geometric design standards for ontario highways. *Information Management Office, Downsview, Ontario, Canada*, 1984. page: 8.
- [Par77] Neville A. Parker. Rural highway route corridor selection. *Transportation Planning and Technology*, 3(4):247–256, 1977. page: 7.
- [Pea73] A D Pearman. Investigation of the objective function in problems of optimising highway vertical alignment. In *PTRC Seminar Proc., Cost Models and Optimisation in Highways, Session L18, London, 1973*. page: 8.
- [RL11] Faisal Md Rahman and Yves Lucet. Dynamic programming for vertical alignment optimization. Technical report, (undisclosed), 2011. page: 10.
- [Rob73] R Robinson. Automatic design of the road vertical alignment. In *PTRC Seminar Proceedings on Cost Models and Optimization in Highways, Session L9, London, UK, 1973*. page: 8.
- [RWW97] C. ReVelle, E.E. Whitlatch, and J.R. Wright. *Civil and environmental systems engineering*. Prentice Hall, 1997. page: 10.
- [Sch00] John George Schoon. *Geometric Design Projects for Highways: an Introduction*. American Society of Civil Engineers, 2000. page: 2.
- [SH81] Jocelyn F. B. Shaw and Bernard E. Howard. Comparison of two integration methods in transportation routing. *Transportation Research Record*, pages 8–13, 1981. page: 7.
- [SH82] Jocelyn F. B. Shaw and Bernard E. Howard. Expressway route optimization by ocp. *Transportation Engineering Journal*, 108:227–243, 1982. page: 7.
- [SN71] Robert M Stark and Robert L Nicholls. *Mathematical Foundations for Design: Civil Engineering Systems*. Technology and Engineering. McGraw-Hill, 1971. page: 10.

Bibliography

- [TM71] A K Turner and R D Miles. The gcars system: A computer assisted method of regional route location. *Highway Research Record*, pages 1–15, 1971. page: 7.
- [Tri87a] Dan Trietsch. Comprehensive design of highway networks. *TRANSPORTATION SCIENCE*, 21(1):26–35, 1987. page: 7.
- [Tri87b] Dan Trietsch. A family of methods for preliminary highway alignment. *Transportation Science*, 21(1):17–25, February 1987. page: 7.
- [TS88] N. R. Thomson and J. F. Sykes. Route selection through a dynamic ice field using the maximum principle. *Transportation Research Part B: Methodological*, 22(5):339–356, October 1988. page: 7.
- [Tur78] A K Turner. A decade of experience in computer aided route selection. *Photogrammetric Engineering and Remote Sensing*, pages 1561–1576, 1978. page: 7.
- [Wan95] Frederic Y. M. Wan. *Introduction to the calculus of variations and its applications*. Chapman & Hall, 1995. page: 6.

Appendix A

Asymptotic Bounds

Definition A.1. Let f and g be real-valued functions. We say that $f(n)$ is $\mathcal{O}(g(n))$ or g is an asymptotically upper bound for f , if there exists constants c and n_0 such that $|f(n)| \leq c|g(n)|$ for all $n > n_0$.

Example A.2. For any algorithm, the running time $T(n) = n^3 + 10n + 1$ is $\mathcal{O}(n^3)$ because $n^3 + 10n + 1 \leq n^3 + 10n^3 + n^3 \leq 12n^3$ for all $n > 1$.

Definition A.3. Let f and g be real-valued functions. We say that $f(n)$ is $\Omega(g(n))$ or g is an asymptotically lower bound for f , if there exists constants c and n_0 such that $|f(n)| \geq c|g(n)|$ for all $n > n_0$.

Example A.4. For any algorithm, the running time $T(n) = n^3 + 10n + 1$ is $\Omega(n^3)$ because $n^3 + 10n + 1 \geq n^3$ for all $n > -\frac{1}{10}$.

Definition A.5. Let f and g be real-valued functions. We say that $f(n)$ is $\Theta(g(n))$ or g is an asymptotically tight bound for f , if and only if $f(n)$ is $\Omega(g(n))$ and $\mathcal{O}(g(n))$.

Example A.6. For any algorithm, the running time $T(n) = n^3 + 10n + 1$ is $\Theta(n^3)$ by Examples A.2 and A.4.

Theorem A.7. $a_n x^n + \dots + a_1 x + a_0$ is $\mathcal{O}(x^n)$ for any real numbers a_n, \dots, a_1, a_0 with $a_n \neq 0$ and any non-negative number n .

Proof. The proof can be found elsewhere [KvT05]. □

Theorem A.8. If $f_1(x)$ is $\mathcal{O}(g_1(x))$, and $f_2(x)$ is $\mathcal{O}(g_2(x))$, then $(f_1 + f_2)(x)$ is $\mathcal{O}(\max(g_1(x), g_2(x)))$.

Proof. The proof can be found elsewhere [KvT05]. □

Theorem A.9. If $f_1(x)$ is $\mathcal{O}(g_1(x))$, and $f_2(x)$ is $\mathcal{O}(g_2(x))$, then $(f_1 * f_2)(x)$ is $\mathcal{O}(g_1(x) * g_2(x))$.

Proof. The proof can be found elsewhere [KvT05]. □

Appendix A. Asymptotic Bounds

Theorem A.10. $a_n x^n + \dots + a_1 x + a_0$ is $\Theta(x^n)$ for any real numbers a_n, \dots, a_1, a_0 with $a_n \neq 0$ and any non-negative number n .

Proof. The proof can be found elsewhere [KvT05]. □

Definition A.11. Let f and g be real-valued functions. We say that $f(n)$ is $o(g(n))$ if $f(n)$ is $\mathcal{O}(g(n))$ but not $\Theta(g(n))$, and $f(n)$ is $\omega(g(n))$ if $f(n)$ is $\Omega(g(n))$ but not $\Theta(g(n))$.

Example A.12. x is $o(x^2)$ and x^2 is $\omega(x)$.

Appendix B

Tables

Table B.1: Required time in seconds ($t_{R,k}$ s), normalized time ($\bar{t}_{R,k}$), optimal cost ($C_{R,k}$), optimal cost with corrected volume ($C_{R,k}^{400}$), and percent absolute relative error ($|\epsilon_{R,k}|$) for the problems ($R-k$) of Test configuration 1 (Roads-A,B)

$R-k$	$t_{R,k}$	$\bar{t}_{R,k}$	$C_{R,k}$	$C_{R,k}^{400}$	$ \epsilon_{R,k} $
A-400	0.61	1.0	12882.6	12882.6	0
A-200	0.75	1.2	12707.4	12936.3	1.8
A-100	0.89	1.5	12726.3	12943.2	1.7
A-50	0.55	0.90	12736.5	12951.8	1.7
A-40	0.36	0.59	12747.9	12952.2	1.6
A-20	0.27	0.44	12878.3	13015.1	1.1
A-10	0.09	0.15	13287.7	13170.8	0.89
A-5	0.08	0.13	14126.1	13459.7	5
A-2	0.06	0.10	16544.3	14144	17
A-AM	0.23	0.38	20408.7	14867.6	37.3
A-NGM	0.23	0.38	10396.8	12997.2	20
B-400	0.51	1.0	246325	246293	0.01
B-200	0.47	0.92	238615	246462	3.2
B-100	0.58	1.1	238612	246422	3.2
B-50	0.23	0.45	238667	246428	3.1
B-40	0.27	0.53	238788	246499	3.1
B-20	0.09	0.18	239592	246587	2.8
B-10	0.08	0.16	242599	247627	2
B-5	0.23	0.45	254468	253048	0.56
B-2	0.16	0.31	304477	271852	12
B-AM	0.11	0.22	386232	292768	31.9
B-NGM	0.08	0.16	185555	255932	27.5

Appendix B. Tables

Table B.2: Required time in seconds ($t_{R,k}$), normalized time ($\bar{t}_{R,k}$), optimal cost ($C_{R,k}$), optimal cost with corrected volume ($C_{R,k}^{400}$), and percent absolute relative error ($|\epsilon_{R,k}|$) for the problems ($R-k$) of Test configuration 1 (Roads-C,D,E)

$R-k$	$t_{R,k}$	$\bar{t}_{R,k}$	$C_{R,k}$	$C_{R,k}^{400}$	$ \epsilon_{R,k} $
C-400	3.8	1.0	415788	415765	0.01
C-200	3.5	0.92	399624	415766	3.9
C-100	2.7	0.71	399637	415766	3.9
C-50	2.1	0.55	399648	415766	3.9
C-40	2	0.53	399707	415766	3.9
C-20	1.2	0.32	400196	415767	3.7
C-10	1.2	0.32	402063	415767	3.3
C-5	0.9	0.24	408086	415811	1.9
C-2	4.3	1.1	463629	417949	10.9
C-AM	3.4	0.89	602781	425284	41.7
C-NGM	5.3	1.4	267501	428078	37.5
D-400	3	1.0	94893	94836	0.06
D-200	3.2	1.1	92278.5	94869.1	2.7
D-100	2.7	0.90	92332.2	94881.8	2.7
D-50	2.1	0.70	92357.7	94889.2	2.7
D-40	1.8	0.60	92410.8	94888.4	2.6
D-20	1.2	0.40	92843.4	95056	2.3
D-10	1	0.33	94529.3	95796.4	1.3
D-5	0.89	0.30	100316	97999.2	2.4
D-2	0.67	0.22	118819	104210	14
D-AM	0.73	0.24	148328	111462	33.1
D-NGM	0.76	0.25	73689.6	96142.2	23.4
E-400	4.1	1.0	1601620	1601610	0
E-200	3.9	0.95	1541690	1602890	3.8
E-100	2.7	0.66	1541640	1602990	3.8
E-50	1.8	0.44	1541940	1603070	3.8
E-40	1.6	0.39	1542320	1603160	3.8
E-20	1.1	0.27	1545260	1603520	3.6
E-10	0.67	0.16	1554970	1605780	3.2
E-5	0.59	0.14	1596920	1618010	1.3
E-2	0.53	0.13	1824600	1691980	7.8
E-AM	0.53	0.13	2304130	1804020	27.7
E-NGM	0.61	0.15	1169080	1708980	31.6

Appendix B. Tables

Table B.3: Required time in seconds ($t_{R,k}$), optimal cost ($C_{R,k}$), optimal cost with corrected volume ($C_{R,k}^{400}$), and percent absolute relative error ($|\epsilon_{R,k}|$) for the problems ($R-k$) of Test configuration 1 (Roads-F,G)

$R-k$	$t_{R,k}$	$\bar{t}_{R,k}$	$C_{R,k}$	$C_{R,k}^{400}$	$ \epsilon_{R,k} $
F-400	7.4	1.0	2078030	2078010	0
F-200	6.1	0.82	2001190	2079570	3.8
F-100	4.7	0.64	2001270	2079670	3.8
F-50	2.9	0.39	2001640	2079820	3.8
F-40	2.4	0.32	2002120	2079910	3.7
F-20	1.5	0.20	2005920	2080570	3.6
F-10	1.2	0.16	2018810	2083480	3.1
F-5	1	0.14	2075680	2100290	1.2
F-2	1.1	0.15	2388580	2203500	8.4
F-AM	0.84	0.11	3008870	2349300	28.1
F-NGM	0.92	0.12	1515610	2196990	31
G-400	38.8	1.0	1049830	1049760	0.01
G-200	44.2	1.1	1009470	1050500	3.9
G-100	39.9	1.0	1009490	1050690	3.9
G-50	26.8	0.69	1009660	1050880	3.9
G-40	24.5	0.63	1009860	1050920	3.9
G-20	21.4	0.55	1011290	1051330	3.8
G-10	17.8	0.46	1017120	1052750	3.4
G-5	13	0.34	1039140	1057940	1.8
G-2	12.5	0.32	1164140	1087030	7.1
G-AM	12.2	0.31	1427720	1136960	25.6
G-NGM	13.2	0.34	728216	1132610	35.7

Appendix B. Tables

Table B.4: Optimal cost (C_{R,n_g}), normalized cost (\bar{C}_{R,n_g}), required time (t_{R,n_g}), and normalized time (\bar{t}_{R,n_g}) for the problems ($R-n_g$) of Test configuration 2

$R-n_g$	C_{R,n_g}	\bar{C}_{R,n_g}	t_{R,n_g}	\bar{t}_{R,n_g}	$R-n_g$	C_{R,n_g}	\bar{C}_{R,n_g}	t_{R,n_g}	\bar{t}_{R,n_g}
A-1	8614.45	1.0	1.5	1.0	E-1	1363880	1.0	23.3	1.0
A-2	12707.4	1.5	2.3	1.5	E-2	1541690	1.1	21.7	0.93
A-5	23316.4	2.7	2.5	1.6	E-3	1752290	1.3	21.2	0.91
A-10	93495.4	10.9	5.0	3.3	E-5	2144850	1.6	40.6	1.7
A-25	133842	15.5	2.4	1.6	E-6	2400900	1.8	9.2	0.39
B-1	109555	1.0	1.7	1.0	E-10	3035850	2.2	10.5	0.45
B-2	238615	2.2	2.2	1.3	E-15	4207050	3.1	15.2	0.65
B-5	502577	4.6	2.7	1.6	E-25	5162780	3.8	33.2	1.4
B-10	767983	7.0	3.2	1.9	F-1	1816420	1.0	36.4	1.0
B-25	957011	8.7	2.3	1.3	F-2	2001190	1.1	33.5	0.92
C-1	399592	1.0	45.3	1.0	F-4	2365320	1.3	86.5	2.4
C-2	399624	1.0	8.8	0.19	F-5	2639150	1.5	57.2	1.6
C-4	399592	1.0	34.8	0.77	F-8	3289100	1.8	46.6	1.3
C-5	400201	1.0	24.0	0.53	F-10	3937060	2.2	15.1	0.41
C-10	401230	1.0	9.3	0.20	F-20	7933540	4.4	90.5	2.5
C-20	407700	1.0	8.5	0.19	F-25	8356650	4.6	554.6	15.2
C-25	404997	1.0	10.5	0.23	G-1	1000420	1.0	115.5	1.0
D-1	85844.2	1.0	25.7	1.0	G-2	1009470	1.0	42.8	0.37
D-2	92278.5	1.1	22.1	0.86	G-3	1019870	1.0	395.7	3.4
D-3	99181.4	1.2	21.0	0.82	G-5	1049250	1.0	758.0	6.6
D-5	116551	1.4	31.9	1.2	G-6	1058990	1.1	804.9	7.0
D-6	120897	1.4	27.1	1.1	G-9	1123460	1.1	131.0	1.1
D-10	237660	2.8	40.3	1.6	G-10	1207270	1.2	815.9	7.1
D-15	303055	3.5	36.4	1.4	G-15	1357230	1.4	113.9	0.99
D-25	343509	4.0	69.6	2.7	G-18	1340080	1.3	1236.7	10.7
					G-25	1580330	1.6	637.8	5.5

Appendix B. Tables

Table B.5: Cost (C_{R,k,n_z,n_r}^b) and timing (t_{R,k,n_z,n_r}^b) of the basic MILP model, cost (C_{R,k,n_z,n_r}^n) and timing (t_{R,k,n_z,n_r}^n) of the quasi-network-flow model, percent difference in cost (δ_{R,k,n_z,n_r}), and percent improvement (η_{R,k,n_z,n_r}) for the problems ($R-k, n_z, n_r$) of Test configuration 3 (Part-1)

$R-k, n_z, n_r$	C_{R,k,n_z,n_r}^b	t_{R,k,n_z,n_r}^b	C_{R,k,n_z,n_r}^n	t_{R,k,n_z,n_r}^n	δ_{R,k,n_z,n_r}	η_{R,k,n_z,n_r}
A-400,5,2	12997.3	5.2	12997.3	1.7	0.00	67.5
A-200,5,2	12795.7	7.1	12818.7	3.6	0.18	48.5
A-100,5,2	12813.3	9.1	12813.3	4.2	0.00	54.3
A-50,5,2	12823.0	9.2	12833.4	1.2	0.08	87.4
A-40,5,2	12834.8	4.9	12847.1	2.1	0.10	58.1
A-20,5,2	12962.3	5.7	12987.8	1.0	0.20	82.2
A-10,5,2	13373.9	4.3	13373.9	0.89	0.00	79.3
A-5,5,2	14239.2	4.4	14221.3	0.97	-0.13	78.2
A-2,5,2	16669.7	2.6	16658.9	0.47	-0.06	82.0
A-1,5,2	20534.6	6.3	20523.6	0.67	-0.05	89.4
A-400,10,2	13034.7	841.0	13034.7	49.0	0.00	94.2
A-200,10,2	12852.7	378.0	12852.7	37.0	0.00	90.2
A-100,10,2	12870.8	264.0	12870.8	33.0	0.00	87.5
A-50,10,2	12880.0	210.0	12880.0	26.0	0.00	87.6
A-40,10,2	12893.4	270.0	12893.4	23.0	0.00	91.5
A-20,10,2	13021.4	470.0	13021.4	22.0	0.00	95.3
A-10,10,2	13431.1	293.0	13438.0	23.0	0.05	92.2
A-5,10,2	14290.1	429.0	14290.1	22.0	0.00	94.9
A-2,10,2	16731.2	316.0	16752.7	16.0	0.13	94.9
A-1,10,2	20651.1	364.0	20651.1	24.0	0.00	93.4
A-400,5,4	12997.3	5.6	12997.3	1.6	0.00	71.6
A-200,5,4	12795.7	7.4	12818.7	4.7	0.18	36.9
A-100,5,4	12813.3	9.2	12813.3	2.7	0.00	70.2
A-50,5,4	12844.0	3.3	12844.0	2.2	0.00	32.1
A-40,5,4	12856.9	5.3	12834.8	1.4	-0.17	73.4
A-20,5,4	12987.8	5.7	12987.8	0.78	0.00	86.2
A-10,5,4	13399.3	5.0	13388.4	0.56	-0.08	88.7
A-5,5,4	14221.3	2.6	14239.2	0.44	0.13	83.3
A-2,5,4	16669.7	2.7	16669.7	0.58	0.00	78.2
A-1,5,4	20571.9	4.9	20534.6	0.50	-0.18	89.7
A-400,10,4	13011.8	61.7	13006.0	14.6	-0.04	76.4
A-200,10,4	12826.4	149.2	12826.4	24.7	0.00	83.5
A-100,10,4	12844.7	55.1	12844.7	14.5	0.00	73.7
A-50,10,4	12854.9	199.6	12854.9	10.2	0.00	94.9
A-40,10,4	12867.4	73.2	12867.4	7.7	0.00	89.5

Appendix B. Tables

Table B.6: Cost (C_{R,k,n_z,n_r}^b) and timing (t_{R,k,n_z,n_r}^b) of the basic MILP model, cost (C_{R,k,n_z,n_r}^n) and timing (t_{R,k,n_z,n_r}^n) of the quasi-network-flow model, percent difference in cost (δ_{R,k,n_z,n_r}), and percent improvement (η_{R,k,n_z,n_r}) for the problems ($R-k, n_z, n_r$) of Test configuration 3 (Part-2)

$R-k, n_z, n_r$	C_{R,k,n_z,n_r}^b	t_{R,k,n_z,n_r}^b	C_{R,k,n_z,n_r}^n	t_{R,k,n_z,n_r}^n	δ_{R,k,n_z,n_r}	η_{R,k,n_z,n_r}
A-20,10,4	12994.3	124.8	12994.3	5.1	0.00	95.9
A-10,10,4	13417.0	102.9	13414.3	9.1	-0.02	91.1
A-5,10,4	14255.6	45.0	14255.6	5.8	0.00	87.2
A-2,10,4	16708.7	25.8	16708.7	5.7	0.00	77.7
A-1,10,4	20623.7	873.0	20623.7	22.2	0.00	97.5
B-400,5,2	248092.0	11.9	248092.0	0.83	0.00	93.1
B-200,5,2	240152.0	13.5	240152.0	1.2	0.00	91.5
B-100,5,2	240204.0	7.2	240154.0	0.95	-0.02	86.8
B-50,5,2	240212.0	10.7	240212.0	0.56	0.00	94.7
B-40,5,2	240331.0	9.3	240331.0	1.0	0.00	89.0
B-20,5,2	241156.0	9.5	241156.0	0.36	0.00	96.2
B-10,5,2	243803.0	9.2	243803.0	0.39	0.00	95.7
B-5,5,2	255556.0	6.2	255556.0	0.25	0.00	95.9
B-2,5,2	306133.0	6.1	307433.0	0.27	0.42	95.7
B-1,5,2	388360.0	6.1	388360.0	1.2	0.00	80.5
B-400,10,2	248092.0	577.5	248092.0	43.2	0.00	92.5
B-200,10,2	240152.0	419.3	240152.0	40.6	0.00	90.3
B-100,10,2	240208.0	233.8	240154.0	12.9	-0.02	94.5
B-50,10,2	240409.0	368.9	240212.0	13.3	-0.08	96.4
B-40,10,2	240472.0	440.5	240357.0	32.6	-0.05	92.6
B-20,10,2	241310.0	329.4	241156.0	33.1	-0.06	89.9
B-10,10,2	244022.0	156.8	243803.0	26.3	-0.09	83.2
B-5,10,2	255732.0	121.0	255556.0	14.8	-0.07	87.8
B-2,10,2	306341.0	106.3	306133.0	28.3	-0.07	73.3
B-1,10,2	388360.0	236.9	388360.0	24.5	0.00	89.7
B-400,5,4	246967.0	4.1	247953.0	2.5	0.40	39.4
B-200,5,4	239245.0	4.6	240427.0	3.3	0.49	28.4
B-100,5,4	239251.0	3.0	240438.0	2.1	0.50	29.3
B-50,5,4	239305.0	1.42	240492.0	1.39	0.50	2.2
B-40,5,4	239420.0	1.5	240606.0	0.90	0.50	38.3
B-20,5,4	240230.0	1.4	241503.0	0.83	0.53	39.1
B-10,5,4	243272.0	1.1	243732.0	0.48	0.19	56.9
B-5,5,4	255199.0	1.3	255556.0	0.62	0.14	51.8
B-2,5,4	305395.0	1.1	306133.0	0.34	0.24	68.6
B-1,5,4	387439.0	1.1	387439.0	0.44	0.00	59.4

Appendix B. Tables

Table B.7: Cost (C_{R,k,n_z,n_r}^b) and timing (t_{R,k,n_z,n_r}^b) of the basic MILP model, cost (C_{R,k,n_z,n_r}^n) and timing (t_{R,k,n_z,n_r}^n) of the quasi-network-flow model, percent difference in cost (δ_{R,k,n_z,n_r}), and percent improvement (η_{R,k,n_z,n_r}) for the problems ($R-k, n_z, n_r$) of Test configuration 3 (Part-3)

$R-k, n_z, n_r$	C_{R,k,n_z,n_r}^b	t_{R,k,n_z,n_r}^b	C_{R,k,n_z,n_r}^n	t_{R,k,n_z,n_r}^n	δ_{R,k,n_z,n_r}	η_{R,k,n_z,n_r}
B-400,10,4	247783.0	23.2	247783.0	8.8	0.00	62.1
B-200,10,4	240001.0	32.0	239859.0	14.6	-0.06	54.2
B-100,10,4	239863.0	17.9	239863.0	5.7	0.00	68.2
B-50,10,4	239922.0	20.1	240069.0	5.0	0.06	75.0
B-40,10,4	240045.0	12.6	240185.0	4.7	0.06	62.6
B-20,10,4	240837.0	9.9	240837.0	3.8	0.00	61.8
B-10,10,4	243732.0	12.0	243732.0	2.6	0.00	78.2
B-5,10,4	255556.0	11.9	256988.0	5.8	0.56	51.1
B-2,10,4	306133.0	11.1	306133.0	3.0	0.00	72.8
B-1,10,4	388360.0	12.6	389967.0	3.2	0.41	74.9
C-400,5,2	418176.0	157.0	418176.0	4.0	0.00	97.5
C-200,5,2	400173.0	130.0	401927.0	6.0	0.44	95.4
C-100,5,2	401940.0	74.0	402054.0	4.0	0.03	94.6
C-50,5,2	399767.0	69.0	402065.0	3.0	0.57	95.7
C-40,5,2	402009.0	73.0	402009.0	2.0	0.00	97.3
C-20,5,2	402939.0	47.0	402492.0	2.0	-0.11	95.7
C-10,5,2	404352.0	79.0	404380.0	2.0	0.01	97.5
C-5,5,2	408842.0	47.0	410372.0	1.0	0.37	97.9
C-2,5,2	463228.0	83.0	463228.0	1.0	0.00	98.8
C-1,5,2	606528.0	56.0	606455.0	1.0	-0.01	98.2
C-400,10,2	421207.0	24665.0	421205.0	292.0	0.00	98.8
C-200,10,2	404836.0	15929.0	405458.0	390.0	0.15	97.6
C-100,10,2	404851.0	13407.0	404851.0	240.0	0.00	98.2
C-50,10,2	404864.0	24764.0	406450.0	549.0	0.39	97.8
C-40,10,2	404925.0	34714.0	404925.0	279.0	0.00	99.2
C-20,10,2	405451.0	13802.0	405451.0	309.0	0.00	97.8
C-10,10,2	408996.0	35082.0	407376.0	200.0	-0.40	99.4
C-5,10,2	413550.0	19250.0	413550.0	185.0	0.00	99.0
C-2,10,2	468418.0	30212.0	466963.0	162.0	-0.31	99.5
C-1,10,2	611882.0	17797.0	611882.0	105.0	0.00	99.4
C-400,5,4	415879.0	31.7	415879.0	6.6	0.00	79.0
C-200,5,4	399713.0	29.3	399713.0	8.9	0.00	69.7
C-100,5,4	401940.0	36.0	399725.0	6.6	-0.55	81.7
C-50,5,4	402034.0	29.9	399737.0	3.5	-0.57	88.4
C-40,5,4	402093.0	17.2	399795.0	3.2	-0.57	81.3

Appendix B. Tables

Table B.8: Cost (C_{R,k,n_z,n_r}^b) and timing (t_{R,k,n_z,n_r}^b) of the basic MILP model, cost (C_{R,k,n_z,n_r}^n) and timing (t_{R,k,n_z,n_r}^n) of the quasi-network-flow model, percent difference in cost (δ_{R,k,n_z,n_r}), and percent improvement (η_{R,k,n_z,n_r}) for the problems ($R-k, n_z, n_r$) of Test configuration 3 (Part-4)

$R-k, n_z, n_r$	C_{R,k,n_z,n_r}^b	t_{R,k,n_z,n_r}^b	C_{R,k,n_z,n_r}^n	t_{R,k,n_z,n_r}^n	δ_{R,k,n_z,n_r}	η_{R,k,n_z,n_r}
C-20,5,4	402575.0	18.6	400284.0	2.5	-0.57	86.4
C-10,5,4	404435.0	17.1	402150.0	2.3	-0.56	86.6
C-5,5,4	410464.0	16.6	409790.0	1.6	-0.16	90.6
C-2,5,4	463334.0	16.6	460590.0	1.5	-0.59	91.3
C-1,5,4	606477.0	17.8	606477.0	1.2	0.00	93.2
C-400,10,4	421207.0	6952.0	420824.0	182.0	-0.09	97.4
C-200,10,4	404374.0	1307.0	404453.0	163.0	0.02	87.5
C-100,10,4	405311.0	566.0	405365.0	134.0	0.01	76.3
C-50,10,4	404398.0	2550.0	404864.0	100.0	0.12	96.1
C-40,10,4	404458.0	3852.0	404537.0	124.0	0.02	96.8
C-20,10,4	404945.0	3876.0	405933.0	83.0	0.24	97.9
C-10,10,4	406816.0	13933.0	406895.0	43.0	0.02	99.7
C-5,10,4	412972.0	4022.0	413053.0	99.0	0.02	97.5
C-2,10,4	465830.0	5483.0	467195.0	75.0	0.29	98.6
C-1,10,4	609939.0	4680.0	610055.0	116.0	0.02	97.5
D-400,5,2	94893.0	196.0	95028.4	6.0	0.14	96.9
D-200,5,2	92278.5	210.0	92278.5	18.0	0.00	91.4
D-100,5,2	92332.2	161.0	92464.7	5.0	0.14	96.9
D-50,5,2	92357.7	162.0	92493.9	3.0	0.15	98.1
D-40,5,2	92410.8	191.0	92410.8	7.0	0.00	96.3
D-20,5,2	92843.4	138.0	93426.1	2.0	0.63	98.6
D-10,5,2	94529.3	130.0	95125.3	1.0	0.63	99.2
D-5,5,2	100316.0	150.0	100316.0	1.0	0.00	99.3
D-2,5,2	119106.0	121.0	118819.0	1.0	-0.24	99.2
D-1,5,2	149405.0	117.0	148835.0	2.0	-0.38	98.3
D-400,10,2	95492.3	3137.0	95383.4	111.0	-0.11	96.5
D-200,10,2	92662.9	5143.0	92557.1	81.0	-0.11	98.4
D-100,10,2	92672.5	4008.0	92719.1	96.0	0.05	97.6
D-50,10,2	92913.1	1549.0	92635.8	71.0	-0.30	95.4
D-40,10,2	92691.1	3648.0	92532.5	79.0	-0.17	97.8
D-20,10,2	93232.7	5237.0	92964.3	53.0	-0.29	99.0
D-10,10,2	94650.8	5554.0	95037.0	41.0	0.41	99.3
D-5,10,2	100844.0	1651.0	100469.0	38.0	-0.37	97.7
D-2,10,2	119386.0	2809.0	119069.0	45.0	-0.27	98.4
D-1,10,2	148780.0	2299.0	148709.0	42.0	-0.05	98.2

Appendix B. Tables

Table B.9: Cost (C_{R,k,n_z,n_r}^b) and timing (t_{R,k,n_z,n_r}^b) of the basic MILP model, cost (C_{R,k,n_z,n_r}^n) and timing (t_{R,k,n_z,n_r}^n) of the quasi-network-flow model, percent difference in cost (δ_{R,k,n_z,n_r}), and percent improvement (η_{R,k,n_z,n_r}) for the problems ($R-k, n_z, n_r$) of Test configuration 3 (Part-5)

$R-k, n_z, n_r$	C_{R,k,n_z,n_r}^b	t_{R,k,n_z,n_r}^b	C_{R,k,n_z,n_r}^n	t_{R,k,n_z,n_r}^n	δ_{R,k,n_z,n_r}	η_{R,k,n_z,n_r}
D-400,5,4	95343.8	35.0	94964.1	5.0	-0.40	85.7
D-200,5,4	92724.3	34.0	92278.5	14.0	-0.48	58.8
D-100,5,4	92780.6	32.0	92332.2	11.0	-0.48	65.6
D-50,5,4	92807.0	29.0	92357.7	2.0	-0.48	93.1
D-40,5,4	92855.8	28.0	92410.8	6.0	-0.48	78.6
D-20,5,4	93294.0	28.0	92912.7	1.0	-0.41	96.4
D-10,5,4	94965.2	28.0	94529.3	1.0	-0.46	96.4
D-5,5,4	100748.0	26.0	100316.0	1.0	-0.43	96.2
D-2,5,4	118819.0	30.0	119241.0	1.0	0.36	96.7
D-1,5,4	148328.0	39.0	148876.0	1.0	0.37	97.4
D-400,10,4	94974.8	224.0	95169.3	8.0	0.20	96.4
D-200,10,4	92346.8	265.0	92485.3	26.0	0.15	90.2
D-100,10,4	92401.3	287.0	92543.3	37.0	0.15	87.1
D-50,10,4	92689.7	172.0	92504.8	19.0	-0.20	89.0
D-40,10,4	92481.1	137.0	92685.3	13.0	0.22	90.5
D-20,10,4	92993.4	166.0	93129.0	3.0	0.15	98.2
D-10,10,4	94595.1	251.0	94677.8	2.0	0.09	99.2
D-5,10,4	100396.0	235.0	100646.0	2.0	0.25	99.1
D-2,10,4	119333.0	168.0	118933.0	3.0	-0.34	98.2
D-1,10,4	148455.0	236.0	149036.0	2.0	0.39	99.2
E-400,5,2	1610000.0	207.0	1610000.0	13.0	0.00	93.7
E-200,5,2	1550000.0	220.0	1550000.0	27.0	0.00	87.7
E-100,5,2	1550000.0	184.0	1550000.0	9.0	0.00	95.1
E-50,5,2	1550000.0	204.0	1550000.0	13.0	0.00	93.6
E-40,5,2	1550000.0	181.0	1550000.0	13.0	0.00	92.8
E-20,5,2	1550000.0	194.0	1550000.0	4.0	0.00	97.9
E-10,5,2	1560000.0	192.0	1570000.0	2.0	0.41	99.0
E-5,5,2	1610000.0	203.0	1610000.0	1.0	0.00	99.5
E-2,5,2	1830000.0	175.0	1830000.0	4.0	0.00	97.7
E-1,5,2	2310000.0	186.0	2320000.0	2.0	0.27	98.9
E-400,10,2	1620000.0	4388.0	1620000.0	204.0	0.00	95.4
E-200,10,2	1560000.0	3267.0	1560000.0	171.0	-0.05	94.8
E-100,10,2	1560000.0	3935.0	1560000.0	109.0	0.00	97.2
E-50,10,2	1560000.0	3160.0	1560000.0	138.0	0.05	95.6
E-40,10,2	1560000.0	3112.0	1560000.0	48.0	0.00	98.5

Appendix B. Tables

Table B.10: Cost (C_{R,k,n_z,n_r}^b) and timing (t_{R,k,n_z,n_r}^b) of the basic MILP model, cost (C_{R,k,n_z,n_r}^n) and timing (t_{R,k,n_z,n_r}^n) of the quasi-network-flow model, percent difference in cost (δ_{R,k,n_z,n_r}), and percent improvement (η_{R,k,n_z,n_r}) for the problems ($R-k, n_z, n_r$) of Test configuration 3 (Part-6)

$R-k, n_z, n_r$	C_{R,k,n_z,n_r}^b	t_{R,k,n_z,n_r}^b	C_{R,k,n_z,n_r}^n	t_{R,k,n_z,n_r}^n	δ_{R,k,n_z,n_r}	η_{R,k,n_z,n_r}
E-20,10,2	1560000.0	3746.0	1560000.0	99.0	0.04	97.4
E-10,10,2	1570000.0	2343.0	1570000.0	95.0	-0.20	95.9
E-5,10,2	1620000.0	5143.0	1610000.0	45.0	-0.20	99.1
E-2,10,2	1840000.0	3187.0	1840000.0	35.0	0.08	98.9
E-1,10,2	2320000.0	3808.0	2320000.0	93.0	0.00	97.6
E-400,5,4	1600000.0	166.5	1600000.0	26.5	0.00	84.1
E-200,5,4	1540000.0	185.9	1550000.0	6.1	0.43	96.7
E-100,5,4	1540000.0	163.2	1550000.0	14.3	0.43	91.3
E-50,5,4	1540000.0	139.5	1550000.0	8.1	0.42	94.2
E-40,5,4	1540000.0	151.1	1550000.0	6.6	0.43	95.7
E-20,5,4	1550000.0	149.8	1550000.0	3.5	0.43	97.7
E-10,5,4	1560000.0	144.4	1560000.0	3.6	0.42	97.5
E-5,5,4	1600000.0	149.3	1600000.0	1.0	0.00	99.3
E-2,5,4	1830000.0	142.5	1830000.0	0.92	0.00	99.4
E-1,5,4	2310000.0	126.1	2310000.0	1.2	-0.26	99.1
E-400,10,4	1620000.0	1855.0	1620000.0	63.0	0.00	96.6
E-200,10,4	1560000.0	2220.0	1560000.0	122.0	0.00	94.5
E-100,10,4	1560000.0	909.0	1560000.0	83.0	0.00	90.9
E-50,10,4	1560000.0	894.0	1560000.0	64.0	0.18	92.8
E-40,10,4	1560000.0	1614.0	1560000.0	55.0	0.00	96.6
E-20,10,4	1560000.0	1183.0	1560000.0	51.0	-0.04	95.7
E-10,10,4	1570000.0	1696.0	1570000.0	41.0	-0.04	97.6
E-5,10,4	1610000.0	888.0	1620000.0	26.0	0.21	97.1
E-2,10,4	1850000.0	653.0	1840000.0	30.0	-0.31	95.4
E-1,10,4	2330000.0	1303.0	2330000.0	11.0	-0.13	99.2
F-400,5,2	2080000.0	483.0	2100000.0	26.0	0.63	94.6
F-200,5,2	2000000.0	506.0	2010000.0	28.0	0.33	94.5
F-100,5,2	2000000.0	471.0	2010000.0	16.0	0.33	96.6
F-50,5,2	2010000.0	485.0	2010000.0	13.0	0.17	97.3
F-40,5,2	2010000.0	459.0	2010000.0	14.0	0.00	96.9
F-20,5,2	2010000.0	568.0	2020000.0	7.0	0.33	98.8
F-10,5,2	2020000.0	447.0	2030000.0	6.0	0.17	98.7
F-5,5,2	2080000.0	448.0	2090000.0	3.0	0.29	99.3
F-2,5,2	2390000.0	442.0	2400000.0	1.0	0.29	99.8
F-1,5,2	3010000.0	445.0	3020000.0	0.32	0.25	99.9

Appendix B. Tables

Table B.11: Cost (C_{R,k,n_z,n_r}^b) and timing (t_{R,k,n_z,n_r}^b) of the basic MILP model, cost (C_{R,k,n_z,n_r}^n) and timing (t_{R,k,n_z,n_r}^n) of the quasi-network-flow model, percent difference in cost (δ_{R,k,n_z,n_r}), and percent improvement (η_{R,k,n_z,n_r}) for the problems ($R-k, n_z, n_r$) of Test configuration 3 (Part-7)

$R-k, n_z, n_r$	C_{R,k,n_z,n_r}^b	t_{R,k,n_z,n_r}^b	C_{R,k,n_z,n_r}^n	t_{R,k,n_z,n_r}^n	δ_{R,k,n_z,n_r}	η_{R,k,n_z,n_r}
F-400,10,2	2090000.0	12902.0	2090000.0	152.0	-0.17	98.8
F-200,10,2	2010000.0	10836.0	2010000.0	143.0	0.30	98.7
F-100,10,2*	2019872.0	-	2010000.0	146.0	-0.55	-
F-50,10,2	2010000.0	16129.0	2010000.0	106.0	0.00	99.3
F-40,10,2	2010000.0	16037.0	2010000.0	98.0	0.06	99.4
F-20,10,2	2010000.0	8241.0	2020000.0	79.0	0.30	99.0
F-10,10,2*	2018807.1	-	2040000.0	66.0	0.92	-
F-5,10,2	2090000.0	14153.0	2090000.0	72.0	0.17	99.5
F-2,10,2	2410000.0	13049.0	2390000.0	85.0	-0.57	99.3
F-1,10,2*	3190000.0	-	3020000.0	57.0	-5.3	-
F-400,5,4	2090000.0	99.0	2090000.0	23.0	-0.30	76.8
F-200,5,4	2010000.0	102.0	2010000.0	21.0	-0.31	79.4
F-100,5,4	2010000.0	84.0	2010000.0	21.0	-0.31	75.0
F-50,5,4	2010000.0	80.0	2010000.0	8.0	0.00	90.0
F-40,5,4	2010000.0	80.0	2010000.0	6.0	0.00	92.5
F-20,5,4	2020000.0	63.0	2010000.0	5.0	-0.31	92.1
F-10,5,4	2030000.0	64.0	2020000.0	2.0	-0.46	96.9
F-5,5,4	2090000.0	63.0	2080000.0	2.0	-0.29	96.8
F-2,5,4	2400000.0	59.0	2400000.0	1.0	-0.39	98.3
F-1,5,4	3030000.0	61.0	3020000.0	1.0	-0.41	98.4
F-400,10,4	2090000.0	4484.0	2080000.0	61.0	-0.38	98.6
F-200,10,4	2010000.0	2796.0	2010000.0	68.0	0.05	97.6
F-100,10,4	2010000.0	2752.0	2010000.0	48.0	0.05	98.3
F-50,10,4	2010000.0	3477.0	2010000.0	37.0	0.00	98.9
F-40,10,4	2010000.0	3031.0	2010000.0	40.0	0.00	98.7
F-20,10,4	2020000.0	2603.0	2010000.0	32.0	-0.38	98.8
F-10,10,4	2030000.0	2557.0	2030000.0	22.0	0.05	99.1
F-5,10,4	2080000.0	3107.0	2080000.0	17.0	-0.23	99.5
F-2,10,4	2400000.0	2558.0	2390000.0	26.0	-0.17	99.0
F-1,10,4	3020000.0	3627.0	3010000.0	14.0	-0.14	99.6
G-400,5,2	1050000.0	11510.0	1050000.0	83.0	-0.01	99.3
G-200,5,2	1010000.0	12132.0	1010000.0	71.0	-0.01	99.4
G-100,5,2	1010000.0	11710.0	1010000.0	40.0	0.00	99.7
G-50,5,2	1010000.0	11433.0	1010000.0	25.0	0.00	99.8
G-40,5,2	1010000.0	11438.0	1010000.0	32.0	-0.01	99.7

Appendix B. Tables

Table B.12: Cost (C_{R,k,n_z,n_r}^b) and timing (t_{R,k,n_z,n_r}^b) of the basic MILP model, cost (C_{R,k,n_z,n_r}^n) and timing (t_{R,k,n_z,n_r}^n) of the quasi-network-flow model, percent difference in cost (δ_{R,k,n_z,n_r}), and percent improvement (η_{R,k,n_z,n_r}) for the problems ($R-k, n_z, n_r$) of Test configuration 3 (Part-8)

$R-k, n_z, n_r$	C_{R,k,n_z,n_r}^b	t_{R,k,n_z,n_r}^b	C_{R,k,n_z,n_r}^n	t_{R,k,n_z,n_r}^n	δ_{R,k,n_z,n_r}	η_{R,k,n_z,n_r}
G-20,5,2	1010000.0	11128.0	1010000.0	13.0	0.00	99.9
G-10,5,2	1020000.0	11191.0	1020000.0	9.0	-0.01	99.9
G-5,5,2	1040000.0	11199.0	1040000.0	6.0	0.00	99.9
G-2,5,2	1170000.0	11352.0	1170000.0	5.0	-0.05	100.0
G-1,5,2	1430000.0	12170.0	1430000.0	14.0	0.00	99.9
G-400,10,2*	-	49191.0	1050000.0	750.0	-	-
G-200,10,2*	-	49148.0	1020000.0	595.0	-	-
G-100,10,2*	-	49191.0	1020000.0	374.0	-	-
G-50,10,2*	-	49037.0	1020000.0	482.0	-	-
G-40,10,2*	-	49194.0	1010000.0	249.0	-	-
G-20,10,2*	-	49117.0	1020000.0	228.0	-	-
G-10,10,2*	-	49228.0	1020000.0	231.0	-	-
G-5,10,2*	-	49171.0	1050000.0	230.0	-	-
G-2,10,2*	-	-	1170000.0	225.0	-	-
G-1,10,2*	-	49103.0	1440000.0	194.0	-	-
G-400,5,4	1050000.0	1456.0	1050000.0	48.0	-0.06	96.7
G-200,5,4	1010000.0	1470.0	1010000.0	72.0	0.00	95.1
G-100,5,4	1010000.0	1424.0	1010000.0	39.0	-0.06	97.3
G-50,5,4	1010000.0	1333.0	1010000.0	24.0	-0.05	98.2
G-40,5,4	1010000.0	1338.0	1010000.0	15.0	-0.06	98.9
G-20,5,4	1010000.0	1301.0	1010000.0	9.0	-0.14	99.3
G-10,5,4	1020000.0	1285.0	1020000.0	6.0	-0.13	99.5
G-5,5,4	1040000.0	1291.0	1040000.0	4.0	0.00	99.7
G-2,5,4	1170000.0	1331.0	1160000.0	4.0	-0.11	99.7
G-1,5,4	1430000.0	1256.0	1430000.0	4.0	0.00	99.7
G-400,10,4	1050000.0	4974.0	1060000.0	153.0	0.06	96.9
G-200,10,4	1010000.0	5886.0	1010000.0	122.0	0.07	97.9
G-100,10,4	1010000.0	4909.0	1010000.0	97.0	0.37	98.0
G-50,10,4	1020000.0	5101.0	1010000.0	79.0	-0.08	98.5
G-40,10,4	1020000.0	4919.0	1020000.0	65.0	0.00	98.7
G-20,10,4	1020000.0	5029.0	1010000.0	16.0	-0.39	99.7
G-10,10,4	1020000.0	12360.0	1020000.0	10.0	0.39	99.9
G-5,10,4	1040000.0	5824.0	1040000.0	9.0	0.05	99.8
G-2,10,4*	1187914.5	-	1170000.0	8.0	-1.5	-
G-1,10,4	1430000.0	6594.0	1440000.0	20.0	0.02	99.7