SCHEDULING AND ROUTING OF SERVICE TRUCKS AND PLANNING OF RESOURCE REPLENISHMENT LOCATIONS FOR WINTER ROADWAY MAINTENANCE

BY

LEILA HAJIBABAI

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Civil Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2014

Urbana, Illinois

Doctoral Committee:

Professor Yanfeng Ouyang, Chair
Professor Rahim Benekohal
Professor Feniosky Peña-Mora
Professor Daniel Work

ABSTRACT

Routing of snow plow trucks in urban and regional areas encompasses a variety of complex decisions, especially for jurisdictions with heavy snowfall. The main activities involve dispatching a fleet of plow trucks from a central depot and/or satellite facilities to clean and spread salt/chemicals on the network links (a.k.a. snow routes). We propose a mixed integer linear program (MILP) model to minimize the total operation time of all snow plow trucks needed to complete a given set of snow routes with multiple plowing priorities, and to reduce the longest individual truck operation time in order to balance the distribution of such travel times for multiple classes of priority. The objective of the formulation includes the weighted sum of the total deadhead travel time and longest individual snow plow truck cycle time. A set of customized construction and local search methods are developed to effectively solve the problem. Empirical case study with real-world data shows that the proposed solution approach is able to optimize snow routes (with or without considering priorities of plowing tasks) in a short amount of time and the model result outperforms the current solution in practice. We also develop a state-of-art snow plow routing software with optimization modules and user-friendly GIS interfaces for snow route analysis and design. This decision-support software optimizes a set of snow plow routes based on a set of user input parameters, and it can help stake-holders, engineers, and planners evaluate snow plow options (such as salt usage, vehicle capacities, fleet size, plowing time during the day) and

provide recommendations on vehicle assignments to snow routes. It also includes sufficient flexibility such that experts can further fine-tune the results before field implementation.

Furthermore, it is sometimes challenging to plan winter maintenance operations in advance because snow storms are stochastic with respect to, e.g. start time, duration, impact area, and severity. Based on progression of the snow storm, additional maintenance demand is arising either periodically or randomly over time and space. Besides, maintenance trucks may not be readily available at all times due to stochastic service disruptions. For instance, the operations of trucks are sometimes subject to failure, possibly due to traffic congestion or mechanical breakdowns. A stochastic dynamic fleet management model is developed to assign available trucks to cover uncertain snow plowing demand. Some tasks, especially those on critical roadway links (such as emergency routes), often have priority and impose a strict service time window constraints (i.e., one or more time periods during which these tasks could be completed) to the fleet schedule. Violation to these time windows may result in severe penalties. In addition, in case service disruption occurs, the backlogged tasks will be counted as new tasks that must be addressed in the next time period. A truck in one specific location and time can be dynamically "repositioned" to a different location, but at a cost. This could occur to in-service trucks which would leave currently assigned tasks in order to serve potentially high priority regions (especially in case of service disruptions), replenish salt or chemicals, etc.). This could also occur to idling vehicles in anticipation of future tasks in certain regions. For simplicity, we assume that all truck repositioning requires exactly one time period.

The objective is to simultaneously minimize the cost for truck deadheading and repositioning, as well as to maximize the benefits (i.e., level of service) of plowing. The problem is formulated into a dynamic programming model and solved using an approximate dynamic programming (ADP) algorithm. Piece-wise linear functional approximations are used to estimate the value function of system states (i.e., snow plow trucks location over time). We apply our model and solution approach to a snow plow operation scenario for Lake County, Illinois. Numerical results show that the proposed algorithm can solve the problem effectively and outperforms a rolling-horizon heuristic solution.

On the other hand, efficiency of winter maintenance service is often affected by network design, and more specifically number and location of resource replenishment facilities that snow plow trucks visit during maintenance operations. Such trucks are often very large in size and their movements affect traffic operations and may contribute to additional congestion during their service. Hence, it is beneficial to simultaneously consider a strategic plan for facility location design as well as transportation network expansion (especially in the neighborhood of the salt replenishment locations) to facilitate traffic operations. Furthermore, routing cost of service trucks under these strategic network decisions shall also be considered. Therefore, an integrated mathematical model for salt dome facility location design is developed, which determines the near-optimum number and location of the salt domes, the near-optimal traffic assignment (both general roadway users and snow plow trucks), snow plow trucks routing cost based on near-optimal network design, and possible roadway capacity expansion. The objective is to minimize the total cost for salt dome facility

construction, transportation infrastructure expansion, transportation delay (for both snow plow truck movements and public travel), as well as deadhead travel. A genetic algorithm (GA) framework (with embedded traffic assignment and continuous approximation (CA) algorithms) is developed. The numerical results show that the integrated solution technique can solve the problem effectively. It shall be noted that although this research focuses on the strategic network design for salt dome facilities and snow plow roadway transportation, the model and solution techniques are suitable for a number of application contexts that simultaneously involve network traffic equilibrium, truck routing, infrastructure expansion, and facility location choices (which determine the origin/destination of multi-commodity flow).

TO MY MOM AND DAD ...

Acknowledgements

I deeply thank my advisor, Professor Yanfeng Ouyang, for his invaluable insights and support on my research achievements. He led me into the field of transportation logistics, and provided me with great vision and excellent research skills throughout my Ph.D. studies. Without his support and guidance, my doctoral research would not have been the same.

I express my sincere gratitude to Professor Feniosky Peña-Mora, who was my advisor during the first year of my Ph.D. studies. His knowledge, expertise, and patience provided me with a great atmosphere for research when I joined his group. I am very grateful to have him on my doctoral committee and thankful for his continuous help and support.

I would like to sincerely thank Professor Rahim Benekohal for his great helps and supports. It was his knowledge and passion in teaching traffic operations courses that led me dig deeply and learn more in this area. I am very grateful to have him in my Ph.D. committee. I am also very thankful to Professor Daniel Work for being in my doctoral committee and providing me with invaluable comments and suggestions. All my committee members, with their expertise in specific areas of civil engineering, helped me expand my perspective regarding the future directions of my research activities.

I greatly appreciate the opportunity that I have been given to conduct research in the fields of transportation engineering and construction management at University of Illinois at Urbana-Champaign (UIUC). I am also very thankful to all UIUC professors from whom I have taken courses and learned knowledge and professional skills. I sincerely thank my colleagues who have helped me or collaborated with me on a number of research and course projects. I would like to specially thank Seyed Mohammad Nourbakhsh and Fan Peng who collaborated with me on developing the model formulation and solution techniques for a research project that is relevant to Chapter 3 of this dissertation.

Most importantly, I would like to thank my family and loved ones. Their support, encouragement, and constant love have sustained me throughout my life and help me achieve my goals.

This research was financially supported in part by the Lake County Division of Transportation, Lake County, Illinois. I would like to thank Martin Buehler, Al Giertych, Paula Trigg, Michael Scott, and Edwin Montano (LCDOT) for data support and helpful feedback. In addition, part of this research is supported by the U.S. National Science Foundation through Grants EFRIRESIN #0835982 and CMMI #0748067.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Each year, ensuring safe mobility of passengers and travelers in adverse winter weather is a difficult task. It requires timely, expedient, and cost effective planning and operation of snow control activities. The responsible agencies must make various complex decisions that range from strategic planning (e.g., snow route planning, snow plow fleet assignment) to operational management (e.g., vehicle routing, salt/chemicals spreading and replenishment). Snow plowing in urban areas is particularly difficult and expensive due to the greater population densities and the lack of space to tolerate snow accumulation. Such planning problems become even more complex when sets of extra operational constraints are required.

Currently, planning of annual snow plow routes is not yet a fully-automated process for most public agencies. Some frequently-used procedures are based on a manual tabular summation of route miles (including both deadhead and salting/plowing miles) of each plow vehicle route (which starts and ends at the central depot). Such approaches do not address vehicle operation times in a systematic optimization framework. As a result, the impact of traffic congestion at different time-of-day is typically not considered, the total deadhead time may not be minimized, and the workloads often differ significantly across snow vehicle routes.

Although a number of new technologies (e.g., AVL) have been implemented in recent years to improve the efficiency of snow removal, the design and management of snow routes still remain a challenge. For example, there are often multiple objectives and complex design requirements related to the snow route design problem. Snow routes need to be designed to work well for the scenario in which each truck travels to the snow routes, makes one plow cycle, and returns to the main depot. The trucks may visit satellite salt locations at any time to replenish salt/chemicals. In addition, existing traffic volumes on the snow routes may cause congestion and longer travel times. Routes with heavier traffic (which varies with time-of-day) should be relatively shorter so the truck cycle time can be balanced. There are also a number of complicating implementation issues that must be considered, such as (a) salting distance for a route must match the truck capacity and (b) the trucks have only limited u-turn locations in the network.

It is common in practice to prioritize roadway segments (e.g., based on traffic volume) so that the plowing and salting operation maximizes certain objectives (e.g., safety). Similarly, higher level of priority can be assigned to roadway links that provide access to critical facilities (e.g., police stations, fire stations, hospitals). On the other hand, the order of performing the plowing tasks for different lanes of a street often begins from the center lane in real practice.

Thus, the center lane of a roadway sometimes has a relatively higher priority than the other lanes.

Furthermore, winter maintenance operations such as snow and ice control services are themselves subject to the adverse impacts of snow storms, and hence they require careful planning. Advance planning of such activities, however, is often very challenging, since (i) storm events are stochastic in nature as their start time, severity, impact areas, and duration generate uncertain maintenance demand, and (ii) maintenance trucks may not be readily available at all times as a result of possible mechanical failures or traffic congestion. These issues affect the overall service reliability, especially in high priority regions where timely snow removal is critical. Therefore, fleet management that accounts for uncertainties in a flexible and demand-responsive fashion is appealing.

On the other hand, snow shall be plowed in a manner so as to minimize traffic obstructions during the service trucks' operations in transportation networks. However, snow plow trucks are relatively heavy in weight and often have difficulty turning at intersections and thus, affect the traffic operations and may contribute to additional congestion during their service. Thus, expanding the roadways (e.g., widening roadway shoulders) can improve roadway capacity and increase truck driver comfort, especially near resource replenishment facilities that trucks visit during maintenance operations. On the other hand, number and location of satellite resource replenishment facilities can also affect the efficiency of maintenance trucks' operations. Hence, it is beneficial to simultaneously consider a strategic plan for facility location design as well as transportation network expansion (especially in the neighborhood of the salt replenishment locations) to facilitate traffic operations and maintain the service level. Furthermore, routing cost of service trucks under these strategic network decisions shall also be considered.

1.2 Objectives and Contributions

We develop mathematical models that aim to not only minimize the total snow plow truck travel times but also balance the distribution of such travel times for multiple classes of priority. The objective of the formulation includes the weighted sum of the total deadhead travel time and longest individual snow plow truck cycle time. A set of customized construction and local search methods are developed to effectively solve the problem. Empirical case study with real-world data shows that the proposed solution approach is able to optimize snow routes (with or without considering priorities of plowing tasks) in a short amount of time. We also develop a state-of-art snow routing software with optimization modules and user-friendly GIS interfaces for snow route analysis and design. This decision-support software optimizes a set of snow plow routes based on a set of user input parameters, and it can help stake-holders, engineers, and planners evaluate snow plow options (such as salt usage, vehicle capacities, fleet size, plowing time during the day) and provide recommendations on vehicle assignments to snow routes. It also includes sufficient flexibility such that experts can further fine-tune the results before field implementation.

Furthermore, we study the fleet assignment problem for snow plow trucks under service disruptions and demand uncertainty as a stochastic dynamic fleet management. Our objective is to minimize the cost of truck repositioning and deadheading (i.e., passing through snow links without salting/plowing) and maximize the benefits from completing the tasks that have arisen. We first develop a mathematical program model for the dynamic management of snow plow trucks considering both uncertain demand and service disruption. Then we implement an approximate dynamic programming (ADP) algorithm Godfrey and Powell (2002a), with piece-wise linear functional approximations, to optimize the size of the snow truck fleet needed at each time period and the tasks to be assigned to each truck. The model and solution algorithm are applied to an empirical case study. Numerical results show that the proposed algorithm can solve the problem effectively and outperforms a benchmark solution.

On the other hand, an integrated mathematical model is developed for salt dome facility location design, which determines the near-optimum number and location of the salt domes, the near-optimal traffic assignment (both general roadway users and snow plow trucks), snow plow trucks routing cost based on near-optimal network design, and possible roadway capacity expansion. The objective is to minimize the total cost for salt dome facility construction, transportation infrastructure expansion, transportation delay (for both snow plow truck movements and public travel), as well as deadhead travel. A genetic algorithm framework (with embedded traffic assignment algorithm and continuous approximation (CA) model for truck routing cost estimation) is developed. The numerical results show that the integrated solution technique can solve the problem effectively.

Chapter 2

Literature Review

This chapter reviews the existing models, solution strategies, and state-of-art applications for network routing of snow plow trucks and dynamic fleet management in winter maintenance operations. Besides, the existing models for strategic planning of network infrastructure in the context of facility location problems are also reviewed.

Section 2.1 below reviews the existing literature related to winter roadway maintenance operations. Section 2.2 reviews the research efforts in the area of dynamic fleet management. Finally, Section 2.3 reviews facility location design literature.

2.1 Network Routing of Service Trucks: Models and Applications

Despite its importance, snow plow routing optimization has not been intensively studied in the literature. This section summarizes some related efforts regarding model development and commercial software design.

On the modeling side, Sochor and Yu (2004) developed heuristic algorithms to route snow plows so as to minimize the total system cost, including the penalty cost for using extra vehicles from the depots. In their solutions, certain depots were repeatedly over- or under- utilized, suggesting that the quantity and/or distribution of available vehicles may be sub-optimal. Perrier et al. (2006) surveyed mathematical models and algorithms for winter road maintenance, including spreading of chemicals and abrasives, snow plowing, loading snow into trucks, and hauling snow to disposal sites. Meta-heuristic approaches, including simulated annealing, tabu search, and elite route pool, were developed to determine a set of vehicle routes that collectively serve all road segments under a set of operational constraints. The model objective includes minimizing deadhead travel time, fixed costs of vehicles and depots, and the number of alternations between deadheading and servicing. The level of service for each class of highways is determined according to their priority. Later, Perrier et al. (2007a,b) optimized the snow plow depot location, fleet sizing and replacement decisions to satisfy service requirements for multiple road classes. The vehicle fleet replacement decision addresses the tradeoff between costs for keeping older vehicles and expenses for purchasing newer vehicles. In addition, Perrier et al. (2008) introduced a heuristic approach to find the optimum routes that can plow an urban area in a minimum time. The model was based on a multi-commodity network flow structure to ensure the connectivity of the route covered by each truck. A more recent study by Salazar-Aguilar et al. (2012) formulated a mixed integer nonlinear program to minimize the longest route service time while ensuring that all lanes in each directional road segment are plowed simultaneously by synchronized vehicles. The set of snow routes are determined by an adaptive large neighborhood search heuristic.

On the practical side, various public agencies have made efforts to design and implement efficient decision support systems for winter maintenance. Haghani and Qiao (2001) and Wang et al. (1995) developed decision support systems that can design efficient routes for salting trucks. A mathematical optimization model as a capacitated rural postman problem was developed and a combination of different heuristics were employed to solve the problem. The Midwest Transportation Consortium (Salim et al., 2002) used GIS and AI techniques to develop an intelligent snow removal asset management system (SRAMS). Later, Wilson et al. (2003) developed a simulation model for snow plow operations so as to support the decisions on snow route length, assignment of snow plows to routes, placement of reloading points, and the collection of labor and material cost in "what-if"scenarios. Sugumaran et al. (2005) implemented a web-based Winter Maintenance Decision Support System (WMDSS) to help stake-holders evaluate various procedures for optimally managing snow removal assets. Later, Hanna et al. (2009) identified various measures to evaluate the performance of snow and ice removal activities, which include friction, time to bare pavement, etc.

In addition, commercial software applications are also available for planning snow routes. "RouteSmart" software (RouteSmart, 2011) includes a snow removal and street sweeping application built within the "ESRI ArcGIS"platform. The system provides the street sweeping routes, constrained days and times, and service to each street. "Automated Snow Plow

Routing Application" is another software application for winter maintenance, which was developed by Caliper Corporation for Hennepin County Department of Public Works, Minnesota in 1999. This application tries to create routes that reduce snow plow deadhead and balance vehicle cycle times. "FleetRoute"(C2Logix, 2011) uses available local city GIS datasets to plan snow disposal and removal, and it considers different roadway priority levels, turn maneuvers at intersections, side of street serviced, multi-lanes, etc. The advantages of such a system include reduced number of trucks and fuel maintenance costs, balanced routes, and reduced vehicle travel time.

Nevertheless, the studies and efforts in the literature still have limitations. First, most developed models do not optimize salt replenishment decisions (e.g., location and timing) for each of the trucks. Second, based on partially revealed information (e.g., the solution approaches and data collection techniques), it appears that few software applications incorporate delays caused by traffic controls (at intersections) and time-of-day congestion. As such, a more comprehensive model is needed to address simultaneously the total travel time, the cycle time balance, salt replenishment location and timing, and time-of-day congestion and traffic control delays (see Chapter 3).

2.2 Stochastic Dynamic Fleet Management

Despite its importance, dynamic management of snow plow trucks has not been thoroughly studied in the literature. This section summarizes some related methodological and empirical efforts in related contexts.

Problems that deal with uncertainty over time are often formulated into dynamic programming models with discrete state and action spaces (Puterman, 2009). Monte Carlo methods (Bertsekas and Tsitsiklis, 1996; Sutton and Barto, 1998) are often used to simulate possible outcomes associated with the uncertainties. For example, Regan et al. (1996) present a demand-responsive fleet management system with dynamic dispatching, load acceptance, and pricing strategies in a real-time decision-making process. A simulation subroutine is developed to evaluate the performance of alternative load acceptance and fleet assignment strategies using real-time information. All these methods often incur excessive computational burden, mainly due to the well-known "curses of dimensionality".

Another school of research on similar problems focuses on the theory of stochastic linear programming (Dantzig and Infanger, 1991; Infanger, 1994; Kall and Wallace, 1994; Prékopa, 1995; Birge and Louveaux, 2011), which can be divided into two-stage vs. multi-stage problems. Dantzig (1955) and Rockafellar and Wets (1991) have formulated the two-stage stochastic linear programs as large-scale optimization problems. However, due to computation burden, this can limit the size of the problem we can solve. There are often possibilities to take an approximation technique that is convergent for a two-stage problem and apply it to a multi-stage problem. However, multi-stage problems are sometimes much more challenging. For example, Birge (1985) presents a nested Benders algorithm for multi-stage stochastic problems; however, the algorithm becomes less effective as the problem size grows. On the other hand, a few Monte Carlo-based methods have been developed in the context of stochastic programs to work with large sample spaces, such as (i) a first sampling-based method for two-stage stochastic programs (Higle and Sen, 1991), (ii) a sampling-based version of nested Benders decomposition (Pereira and Pinto, 1991), and (iii) a convergent sampling-based version of nested Benders for multi-stage problems (Chen and Powell, 1999). Again, due to computational burden, these methods have size limitations as all of them depend on Benders cuts to approximate the impact of decisions on future periods.

Several approximation techniques have been developed to solve stochastic resource allocation problems with either continuous (Jordan and Turnquist, 1983; Powell, 1986) or integer decision variables (Powell, 1987; Frantzeskakis and Powell, 1990; Cheung and Powell, 1996; Powell and Carvalho, 1998). Earlier efforts by Powell (1987), Frantzeskakis and Powell (1990), and Cheung and Powell (1996) have not considered time windows for serving tasks. An interesting approximation technique that works well on deterministic problems, is linear approximation with a multiplier adjustment (LAMA)(Carvalho and Powell, 2000; Schenk and Klabjan, 2008, 2010). This method, however, does not include stochasticity. Later, concave adaptive value estimation (CAVE) algorithm (Godfrey and Powell, 2001) based on a concave piece-wise linear approximation of the value function is introduced, and it is found to be more responsive than linear approximation methods. CAVE algorithm can be applied to two-stage as well as multi-stage problems with single period travel times (Godfrey and Powell, 2002a). Problems with multi-period travel times solved using CAVE technique have been studied in Gregory and Powell (2002b).

Another line of research in this area is devoted to dynamic vehicle routing (Pillac et al., 2013; Jaillet and Wagner, 2008; Jaillet and Lu, 2011; Yang et al., 2004). Pillac et al. (2013)

provides a review of research efforts and real-world applications that address two significant dimensions: evolution and quality of information. Evolution of information presents the availability of information to the planner (e.g., arrival of new customer demands). Quality of information, on the other hand, reflects possible inaccuracy in the available data (e.g., demand uncertainty). Furthermore, vehicle routes can either be designed statically or dynamically. For example, routing problems with stochastic demands may be designed by simply make minor changes to a set of static (but robust) routes (Bertsimas and Simchi-Levi, 1996; Gendreau et al., 1996). In addition, they can also be implemented by communicating to the vehicles in ad-hoc and assigning them to remaining demands in an on-line fashion (Secomandi, 2001; Novoa and Storer, 2009; Secomandi and Margot, 2009).

There are a number of variations to the on-line vehicle routing problem. Yang et al. (2004) introduce a real-time multi-vehicle truckload delivery problem. They propose a mixed integer programming formulation for the off-line version of the problem and a rolling-horizon framework for the real-time version. Jaillet and Lu (2011) propose theoretical models for on-line traveling salesman problems to incorporate acceptance or rejection decisions on newly arrived demand requests. Jaillet and Wagner (2008) consider on-line routing optimization problems with the objective of minimizing the total travel time under precedence and capacity constraints. They use polynomial-time on-line algorithms and show that these algorithms are almost surely asymptotically optimal under general stochastic structures of the problem.

Despite all the efforts in these closely related areas, the problem of dynamic snow plow fleet management under service disruptions and demand uncertainty has not been fully addressed. As such, Chapter 4 proposes a mathematical model to address this problem.

2.3 Facility Location Design under Traffic Congestion

The facility location problem has been extensively studied; e.g., see Drezner (1995) and Daskin (1995) for systematic reviews of classic discrete location models. In the context of location problem for supply chain design, Eathington and Swenson (2007) used Geospatial Information Systems (GIS) applications to facilitate decisions on refinery site, size and technology under various scenarios regarding energy demand level, industry growth, and impacts on the job market. Most research efforts consider distance as the measure of transportation cost (e.g., Graves and Willems, 2005); however, in recent years, researchers started to incorporate the impact of congestion (e.g., travel time) into facility location design. For instance, Konur and Geunes (2011) studied a competitive facility location problem subject to distribution network congestion. Jayaram (2005) evaluated the impacts of roadway congestion on manufacturers, distributors, and the existing supply chains. López and Monzón (2010) developed strategic shipment plan models that integrate sustainability issues into facility location planning and spatial and regional economic analysis. Bai et al. (2011) analyzed the interactions among biofuel refinery location and shipment routing under network congestion decisions as a result of increased biomass supply and biofuel demand shipments. Such effects have been incorporated into the planning of biorefinery location and transportation routing, with the focus on the biofuel industry expansion. A fixed-charge facility location model has been jointly formulated with a traffic assignment model, and the problem is solved effectively by Lagrangian relaxation (Fisher, 1981) with an embedded convex combination method (Frank and Wolfe, 1956; Sheffi, 1985). Another related research has focused on the integrated planning of supply chain networks and multimodal transportation infrastructure expansion to mitigate congestion (Hajibabai and Ouyang, 2013). Bai et al. (2012) considered a decentralized biofuel supply chain system under biomass market equilibrium in which farmers and biofuel manufacturers are non-cooperative decision makers. Recently, Wang et al. (2013) addressed biofuel consumption mandates with renewable identification numbers.

Movement of snow plow trucks induces higher transportation demand that originates or ends at the depots/salt domes. Such demand could cause additional congestion delay in the transportation network particularly on roadway links (bottlenecks) where public traffic demand is near or has reached roadway capacity or there is no enough space for the service trucks to maneuver (especially on local roads close to the salt domes). This not only increases travel time of the general roadway users but also has a significant impact on the operational efficiency of the roadway winter maintenance itself, which, in turn, shall affect the salt dome location decisions. Unfortunately, these endogenous relationships have largely been ignored in the context of winter maintenance operations. While there have been strategic transportation planning models that integrate sustainability issues into facility location design, network analysis, and spatial and regional economic analysis (e.g., López and Monzón (2010)), the most relevant study in this direction is probably Hajibabai and Ouyang (2013), which shows that integrating facility location, multi-modal shipment routing, and infrastructure expansion decisions could help mitigate congestion impacts. Thus, adding lanes to existing roadways in the neighborhood of salt dome facilities (or building local access roads to newly constructed facilities) should be considered as an integral part of the strategic planning for winter maintenance operations.

In general, capital investment and land use restrictions tend to prohibit extensive roadway expansion as an alternative to mitigate congestion (MTP, 2006). Hence, roadway investment decisions under dynamic responsive traffic demand, have largely been focusing on long-term infrastructure maintenance planning and rehabilitation (e.g., Ouyang (2007); Ng et al. (2009); Gao et al. (2012); Hajibabai et al. (2014a)) or highway investment alternatives under a certain budget (e.g., Li et al. (2010)). However, in the context of roadway winter maintenance, building additional capacity in the roadway network may be much more feasible as it facilitates the operation of the trucks and mitigates the resulting traffic delays. The capital investment associated with transportation infrastructure expansion may be considered as part of the winter maintenance investment plan, and potential public-private partnership may be established to facilitate the investment (Unnikrishnan et al., 2009). On the other hand, routing cost of service trucks are affected by network design; however, including the routing component in the integrated strategic network design model makes the problem even more complex. Thus, we apply continuous approximation approach to approximate the optimal routing cost (see (Daganzo, 2005) for more details on continuous approximation and their application in logistics). Shen and Qi (2007) has considered a supply chain design problem with the objective to minimize the total cost for distribution centers' location, inventory at such centers, and distributors' routing in the supply chain. In Chapter 5 we propose an integrated mathematical model to address the interrelationships among network design (i.e., facility location as well as roadway capacity expansion), transportation delay, and approximate routing of service trucks.

Chapter 3

Network Routing of Snow Plows with Resource Replenishment and Plowing Priorities: Formulation, Algorithm, and Application

This chapter is through a joint work with Seyed Mohammad Nourbakhsh, Yanfeng Ouyang, and Fan Peng on the snow plow routing optimization project for the Lake County Division of Transportation.[1]

3.1 Introduction

Routing of snow plow trucks in urban and regional areas encompasses a variety of complex decisions, especially for jurisdictions with heavy snowfall. The main activities involve dispatching a fleet of plow trucks from a central depot and/or satellite facilities to clean and spread salt/chemicals on the network links (a.k.a. snow routes). In this chapter, a mixed integer linear program (MILP) model is proposed to minimize the total operation time of all

[1]Hajibabai, L., Nourbakhsh, S.M., Ouyang, Y., and Peng, F. Network Routing of Snow Plows with Resource Replenishment and Plowing Priorities: Formulation, Algorithm, and Application, Transportation Research Record: Journal of the Transportation Research Board. In Press.

snow plow trucks needed to complete a given set of snow routes with multiple plowing priorities, and to reduce the longest individual truck operation time. Customized construction and local search solution algorithms are developed and used to design snow routes for an empirical application. The computational results show that the proposed solution approach is able to solve the problem effectively and the model result outperforms the current solution in practice. The proposed models and algorithms are also incorporated into the development of a state-of-art snow plow routing software that helps planners optimize snow routes and evaluate resource allocation options.

The exposition of this chapter is as follows. Section 3.2 introduces the problem statement and the mathematical model formulation. Section 3.3 proposes a customized solution approach that can efficiently solve the problem. Section 3.3.2 presents the empirical case study. Conclusion and future research directions are presented in Section 3.4.

3.2 Model Development

3.2.1 Data Cleaning and Network Preparation

In a real-world roadway network, each segment may have one or more lanes in each direction, and hence may need to be plowed (i.e., passed) multiple times. We define a plowing task as a single pass on a segment, and therefore every roadway segment in the real-world roadway network may generate a set of tasks. In the first step of data cleaning, we construct a new directed network, where the intersection/node sets remain the same (as the original roadway network) but every roadway segment is replaced by a set of directed arcs, each representing a pass. This process is illustrated in Figure 3.1(a). The travel time on each arc is based on the expected travel time on the corresponding roadway segment, which is available as a function of the time-of-day.

The network database can be further treated to avoid unnecessary computational efforts. Two data cleaning procedures could be applied: (1) end node removal, and (2) mid-block treatments, as shown in Figure 3.1 (b)-(c). End nodes are identified and removed if and only if the following criteria are all satisfied: there is no salt dome at the node, no task link is incident to that node, there is no more than one neighboring node, and that the only neighboring node does allow u-turn. A mid-block node is detected and removed if and only if the following criteria are all satisfied: there is more than one neighboring node, no salt dome is at the node, and the roadway condition on the incident links are exactly identical (e.g., there are the same number of passes).

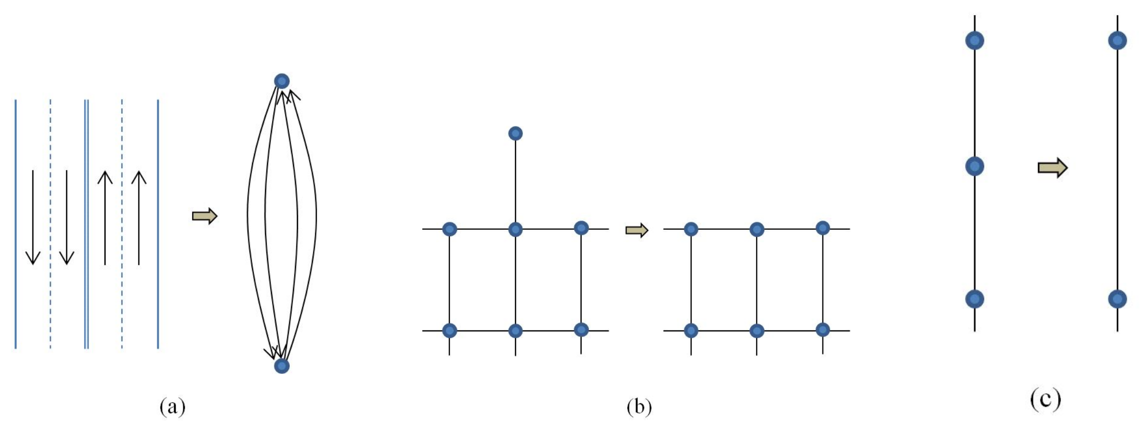


Figure 3.1: Network treatments; (a) number of passes, (b) end-node removal, and (c) mid-block node removal.

Snow plow trucks are relatively heavy in weight and often have difficulty turning at intersections. The delay due to different types of turns (e.g., left turn, right turn, and u-turn) must be considered. There are various studies which proposed network representation methods in order to define specific movement delays (see Pallottino and Scutella, 1998, for more details). For example, Anez et al. (1996) introduced a *dual network*, where the original arcs are considered as nodes and an arc is defined for each pair of subsequent arcs in the original graph, whose cost now includes any turning penalty along the travel. An alternative *path-based* method introduces a path as a sequence of arcs instead of a sequence of nodes, where the path cost can be defined in terms of the arc costs as well as the pair of arcs penalties (Kirby and Potts, 1969). In our study, to define specific movement delays and specify u-turn allowance at each intersection, we follow the *expanded network* method in (Sheffi, 1985) to define each n-leg intersection by a $2n$-node representation.

As in Figure 3.2, the added set of directional arrows help specify a specific delay for each possible turning movement. Similarly, in case of a 3-leg (or 2-leg) intersections we use a 6-node (or 4-node) representation in the new network.

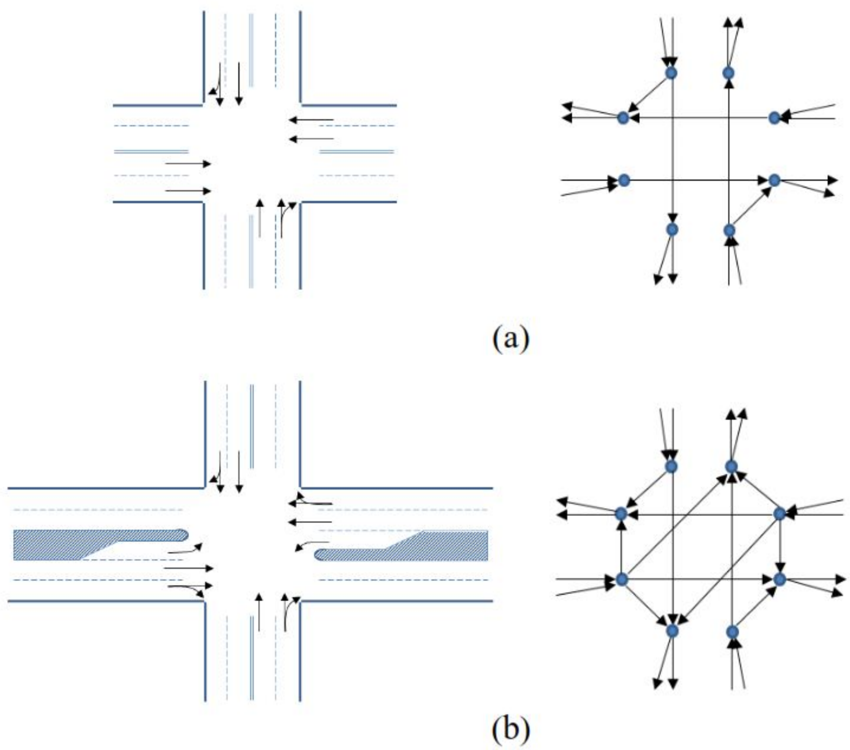


Figure 3.2: Four-leg intersection representation; (a) u-turn is not allowed in any direction and (b) u-turn is allowed for E-W bound traffic.

In case an intersection does not contain 90 degree turns, a delay function is defined to provide the delay penalty, $D(\delta)$ as a function of the turning angle, δ. The turning angle δ is measured between the out-going direction and the in-coming direction for each turn. In our work, we propose a linear interpolation function based on known delay penalties, d_r, d_t, d_l, and d_u, that are associated with the 90-degree right, 180-degree through, 270-degree left, and 0/360-degree u-turn movements, as follows.

$$D(\delta) = \begin{cases} \frac{\delta}{90}d_r + \frac{90-\delta}{90}d_u & 0 \leq \delta < 90; \\ \frac{\delta-90}{90}d_t + \frac{180-\delta}{90}d_r & 90 \leq \delta < 180; \\ \frac{\delta-180}{90}d_l + \frac{270-\delta}{90}d_t & 180 \leq \delta < 270; \\ \frac{\delta-270}{90}d_u + \frac{360-\delta}{90}d_l & 270 \leq \delta < 360 \end{cases} \tag{3.1}$$

As the final step of data preparation, we compute the shortest path travel time/distance from the end of each plowing task to the beginning of every other task. The shortest travel times are stored as data input to the optimization model.

3.2.2 Optimization Model

Let I be the set of plowing tasks, and S be the set of salt replenishment tasks, i.e. visiting one of the satellite salt locations (salt domes). Let K be the set of all snow plow trucks and d_k be the depot for each truck $k \in K$. Similar to Perrier et al. (2008), we let P be the set of priority classes for plowing tasks. In order to factor congestion in travel time computation for each link from i to j, the average delay at intersections (due to signal timing) is included in the link travel time. Based on the shortest path computations in Section 3.2.1, we use $t_{i,j}$ to denote the deadhead time needed from the task (or the depot) $i \in I \cup S \cup \{d_k\}$ to the task (or depot) $j \in I \cup S \cup \{d_k\}$ across the network. Let $t_{task,i}$ denote the duration of each task $i \in I \cup S$. At the salt domes, this task time includes the average vehicle waiting time as well as the salt loading time. We let l_i represent the salt consumption during a task $i \in I$, L_k

be the maximum salt carrying capacity of a truck $k \in K$, and U be the time length of the planning horizon. We let parameter $\delta_{i,p}$ define the priority of each task $p \in P$, as follows:

$$\delta_{i,p} = \begin{cases} 1 & \text{if task } i \in I \text{ has priority } p \in P; \\ 0 & \text{otherwise.} \end{cases}$$

The movements of the trucks are determined by the binary decision variables $\mathbf{x} = \{x_{i,j,k} : i, j \in I \cup S \cup \{d_k\}, k \in K\}$ as follows:

$$x_{i,j,k} = \begin{cases} 1 & \text{if truck } k \in K \text{ travels directly from } i \in I \cup S \cup \{d_k\} \text{ to } j \in I \cup S \cup \{d_k\}; \\ 0 & \text{otherwise.} \end{cases}$$

We let $\mathbf{u} = \{u_i : i \in I \cup S\}$ denote the set of real-valued start times of plowing or salt replenishment tasks. Without losing generality, let the scheduling horizon start at time 0. To catch the amount of salt on a truck $k \in K$, we use $\mathbf{v}_{i,k}$ to denote the remaining salt on a truck $k \in K$ before beginning a task $i \in I \cup S$. Finally, we let z_p be the longest individual truck cycle time for each priority $p \in P$. A smaller z_p indicates a better level of service (LOS); i.e., the balance of workload across vehicles. Besides z_p, we also would like to minimize the total travel time of all vehicles in order to reduce fuel consumption (and labor cost, etc). Let $c_{LOS,p}$ and c_{Fuel} be, respectively, the weights of z_p and the total travel time in the objective function. Therefore, the snow plow routing problem under resource constraints and plowing priorities can be formulated as (3.2a)-(3.2l).

The objective function (3.2a) simultaneously minimizes the total deadhead travel time and the longest individual truck cycle time for all priority classes with their associated weights. The total task time is a constant and hence excluded here. Constraints (3.2b) guarantee that any truck $k \in K$ exits from the depot; similarly constraints (3.2c) guarantee that any truck $k \in K$ goes back to the depot. Constraints (3.2d) indicate that any truck $k \in K$ that enters a link or satellite salt location, will exit that link or satellite salt location. Constraints (3.2e) ensure that every task should be performed exactly once. $\sum_{j \in I \cup S \cup \{d_k\}} x_{i,j,k} = 1$ if task $i \in I$ is assigned to truck $k \in K$, and 0 otherwise. Since sub-tour elimination is very important in solving vehicle routing problem, constraints (3.2f) are considered to establish the relationship between plowing task or salt replenishment start times and truck routes, and to eliminate sub-tours. Constraints (3.2g) represent the salt consumption for each task link. According to these constraints when a truck $k \in K$ moves from task $i \in I$ to a task or satellite salt location $j \in I \cup S$, the amount of salt will decrease by l_i. Constraints (3.2h) indicate the salt replenishment for vehicle $k \in K$. Constraints (3.2i) define the longest individual truck cycle time z_p in the objective function. For simplicity, we assume the initial amount of this variable is the maximum salt carrying capacity of truck $k \in K$; however, in general, different amounts for each truck $k \in K$ can be assumed. Constraints (3.2j) define the binary variables $\{x_{i,j,k}\}$. Constraints (3.2k) set the initial times of the tasks to 0. Finally, constraints (3.2l) indicate the non-negativity of variables $\{v_{i,k}\}$.

$$\text{minimize} \quad \sum_{p \in P} c_{LOS,\,p}\, z_p + c_{Fuel} \sum_{k \in K} \sum_{i \in I \cup S \cup \{d_k\}} \sum_{j \in I \cup S \cup \{d_k\}} t_{i,j}\, x_{i,j,k}, \tag{3.2a}$$

subject to

$$\sum_{i \in I} x_{d_k,i,k} = 1,\ \forall k \in K, \tag{3.2b}$$

$$\sum_{i \in I} x_{i,d_k,k} = 1, \forall k \in K, \tag{3.2c}$$

$$\sum_{j \in I \cup S \cup \{d_k\}, i \neq j} x_{i,j,k} - \sum_{j \in I \cup S \cup \{d_k\}, i \neq j} x_{j,i,k} = 0,\ \forall i \in I \cup S,\ k \in K, \tag{3.2d}$$

$$\sum_{k \in K} \sum_{j \in I \cup S \cup \{d_k\}} x_{i,j,k} = 1, \forall i \in I, \tag{3.2e}$$

$$u_i + t_{task,i} + t_{i,j} + U\,(\,x_{i,j,k} - 1) \leq u_j, \forall i \in I \cup S, j \in I \cup S \cup \{d_k\}, i \neq j, k \in K, \tag{3.2f}$$

$$v_{i,k} - l_i - L_k\,(x_{i,j,k} - 1) \geq v_{j,k}, \forall i \in I, j \in I \cup S \cup \{d_k\}, i \neq j, k \in K, \tag{3.2g}$$

$$v_{j,k} \leq L_k, \forall j \in I, k \in K, \tag{3.2h}$$

$$z_p \geq \delta_{i,p}\,(t_{task,i} + u_i), \forall i \in I, p \in P, \tag{3.2i}$$

$$x_{i,j,k} \in \{0, 1\}, \forall i, j \in I \cup S \cup \{d_k\}, k \in K, \tag{3.2j}$$

$$u_i \geq 0, \forall i \in I\ , \text{ and} \tag{3.2k}$$

$$v_{i,k} \geq 0, \forall i \in I, k \in K. \tag{3.2l}$$

3.3 Solution Approach

Our formulation of the snow plow routing problem is in the form of a vehicle routing problem (VRP), which is usually solved using constructive heuristics and local search algorithms (Pillac et al., 2013). Constructive heuristics build a solution from scratch, while local search algorithms improve an existing solution. Constructive heuristics proposed for VRP includes "I1"by Solomon (1987), where every route is initialized with a "seed" activity and the remaining unscheduled activities are added to the route until its total duration reaches the scheduling horizon. A parallel version of I1 by Potvin and Rousseau (1993) initializes all routes at once and then adds the remaining unscheduled activities one by one. Another I1-based algorithm by Ioannou et al. (2001) inserts activities in a way that the impact on all customers is minimized. A comprehensive review can be found in Bräysy and Gendreau (2005). Several algorithms have been developed to solve such routing problems with various side constraints such as vehicle capacity constraints, pickup and delivery constraints, route length constraints, and time window constraints (Solomon, 1987; Laporte, 1992). Considering the fact that each snow route involves a set of segments of the roadway, snow plow routing problem can also be formulated into an arc routing problem (ARP) (Eiselt et al., 1995; Peng and Ouyang, 2012, 2014; Peng et al., 2011, 2014), where each plow task is represented by a directed arc. When it is required to traverse only a subset of arcs, the problem becomes a rural postman problem (RPP). There are a few practical contexts where it is necessary to service all arcs of the network, hence, most real-world arc routing applications are often modeled as RPPs (Eiselt et al., 1995). The objective, here, is to find the cheapest

Hamiltonian cycle containing each of these edges (and possibly others). This is a special case of the minimum general routing problem which specifies precisely which vertices the cycle must contain. If this subset does not form a weakly connected graph but forms a number of disconnected components, the problem is NP-Complete, and is also a generalization of the asymmetric traveling salesman problem (TSP) (Christofides et al., 1986). For large-scale routing and scheduling problems with many side constraints, constructive heuristics may be more customized and complex. In most construction heuristics, after the activities are assigned to each vehicle, a TSP needs to be solved to determine the routing of each vehicle.

Local search algorithms proposed for VRP often explore two types of neighborhood structures: node interchange and edge interchange. Node interchange includes "insertion"or "relocation"(Savelsbergh, 1992), "exchange"or "swap"(Savelsbergh, 1992), CROSS-exchange (Taillard et al., 1997), and ejection chain (Glover, 1992). Edge interchange includes -opt* (Potvin and Rousseau, 1995), -opt (Russell, 1977), and Or-opt (Or, 1976). Besides these two neighborhood structures, many other structures are also developed, such as GENI-exchange (Gendreau et al., 1992) and cyclic transfers (Thompson and Psaraftis, 1993). A comprehensive review can be found in Bräysy and Gendreau (2005) and Funke et al. (2005).

The implementation of a heuristic usually requires customized algorithm design based on the characteristics of the specific problem, and its computational performance highly depends on the problem structure and even the input data. Therefore, although the algorithms developed by previous studies can serve as useful references, it is very important to adapt and further improve them for the snow plow routing optimization.

3.3.1 Heuristic Methods

Figure 3.3 illustrates the structure of the proposed heuristic solution methods. First, an initial solution is generated by a construction heuristic, and then this solution is improved by a local search algorithm until the stopping criteria are met.

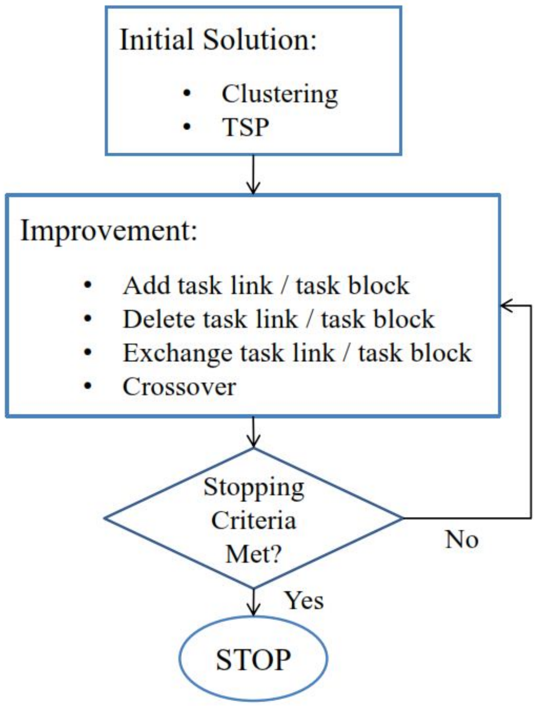


Figure 3.3: Heuristic method diagram.

Initial Solution

In order to generate an initial solution to the truck routing problem, we use a customized K-mean clustering method, which aims to partition $|I|$ task links in the network into $|K|$ clusters, $\{R_k : k \in K\}$. Each task link belongs to the "nearest" cluster in terms of the network travel time from this task link to the cluster. In so doing, we want to first find a task link μ_k in each cluster $k \in K$ (i.e., center of this cluster) which has the minimum travel time from/to all the tasks in cluster R_k. Then, we minimize $\sum_{k \in K} \sum_{i \in R_k} t(w_i, \mu_k)$, where

w_i is a vector representing the network location of task link i, and $t(w_i, \mu_k)$ is the shortest network travel time between w_i and μ_k. Solving this minimization problem results in a set of clusters, each representing a subset of task links I_k to be plowed by truck $k \in K$. This clustering problem is NP-hard, and we employ the iterative Lloyd's algorithm (Kanungo et al., 2002) to find a near-optimum solution.

Clustering yields an assignment of task links to each truck; however, the sequence of plowing these tasks is still unknown. Therefore, we apply the traveling salesman problem (TSP) algorithm to minimize the travel time of each truck while visiting each of the assigned task links exactly once. Knowing that TSP is NP-hard, we first apply the nearest neighbor algorithm to find the initial sequence of the task links, which begins and ends at the truck depot. We further introduce local perturbations to the sequence of the tasks to improve the solution.

During the random perturbation, in order to avoid separating neighboring task links (i.e., those with a short deadhead travel time in between), we also define a "task block" which includes a set of task links: if the deadhead time between any two tasks is less than a predetermined threshold, these tasks belong to the same block. The higher the value of the threshold, the fewer number of task links included in task blocks. We keep each task block unchanged in the TSP improvement algorithm as if it is an individual task.

Improvement

Given the initial solution from the construction heuristic, we apply a customized tabu search algorithm to improve the current solution. We iteratively apply four different types of local perturbations on the task links or task blocks: add, delete, exchange, and crossover. After each local perturbation, we solve the TSP for each cluster and check the objective value. If it is improved, we accept the current perturbation, otherwise we reject it with a high probability and the task links/task blocks in the new solution will be added to a number of tabu lists (i.e., T_a, T_d, or T_e denoting the add, delete, and exchange tabu lists, respectively) so that they will not be considered again in the next few iterations. Figure 3.4 below illustrates the different types of perturbations.

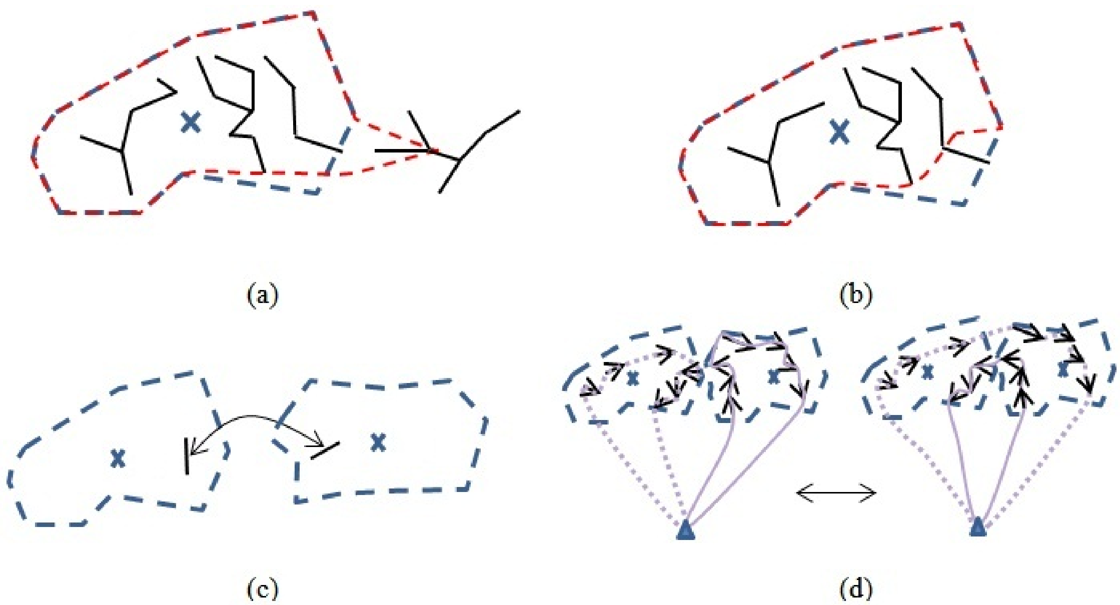


Figure 3.4: (a) Add a task link to a cluster, (b) Delete a task link from a cluster, (c) Exchange task links between two clusters, and (d) Crossover two paths in two clusters.

We keep improving the solution until either of the two stopping criteria are met: maximum running time is reached or there is no significant improvements after several consecutive iterations. Since snow plow trucks apply salt/chemicals on the task links while plowing, we need to determine the optimum location to visit the salt domes for salt replenishments.

To address this issue, a dynamic programming algorithm is implemented for each truck within each tabu-search iteration. Given the sequence of the plowing task (as given by the local search), we need to determine the time and location to replenish salt. The decision stages are either the starting point at the depot or end of each task link, and the state is the remaining salt level in the truck. There are two allowed actions between two consecutive stages: either travel from one stage to the other without salt replenishment, or visit the best salt dome before traveling to the next stage. The best salt dome is determined by minimizing the "detour" time including the travel time from the current task link to the salt dome, the salt replenishment time, and the travel time from the salt dome to the next task link. Going from one stage to next stage, if salt dome is not visited, the salt level decreases by the amount of consumption; otherwise, the salt/chemical level goes up to the truck's capacity before consumption. The cost-to-go function at any stage is defined as the minimum travel time from the depot (starting with full salt load) to that stage without running out of salt along the way. The optimal salt level trajectory found at the last stage (i.e., end of the last task link) provides the exact minimum travel time and the corresponding optimal salt dome visits.

3.3.2 Real-World Case Study

In this section, the model and solution algorithms are applied to a real-world case study based on data from Lake County, Illinois. The highway network database of the Lake County Division of Transportation (LCDOT) contains 1,354 nodes, 3,350 roadway links, 13 salt replenishment locations (including the main depot), and 816 task links. The data cleaning treatments on this network (i.e., end node and mid-block node removals) led to an approximately 30% reduction in the total number of nodes and links, as shown in Figure 3.5.

The intersection representation, however, increases the network size almost by six folds. Table 3.1 summarizes the network size after each step of the network treatments.

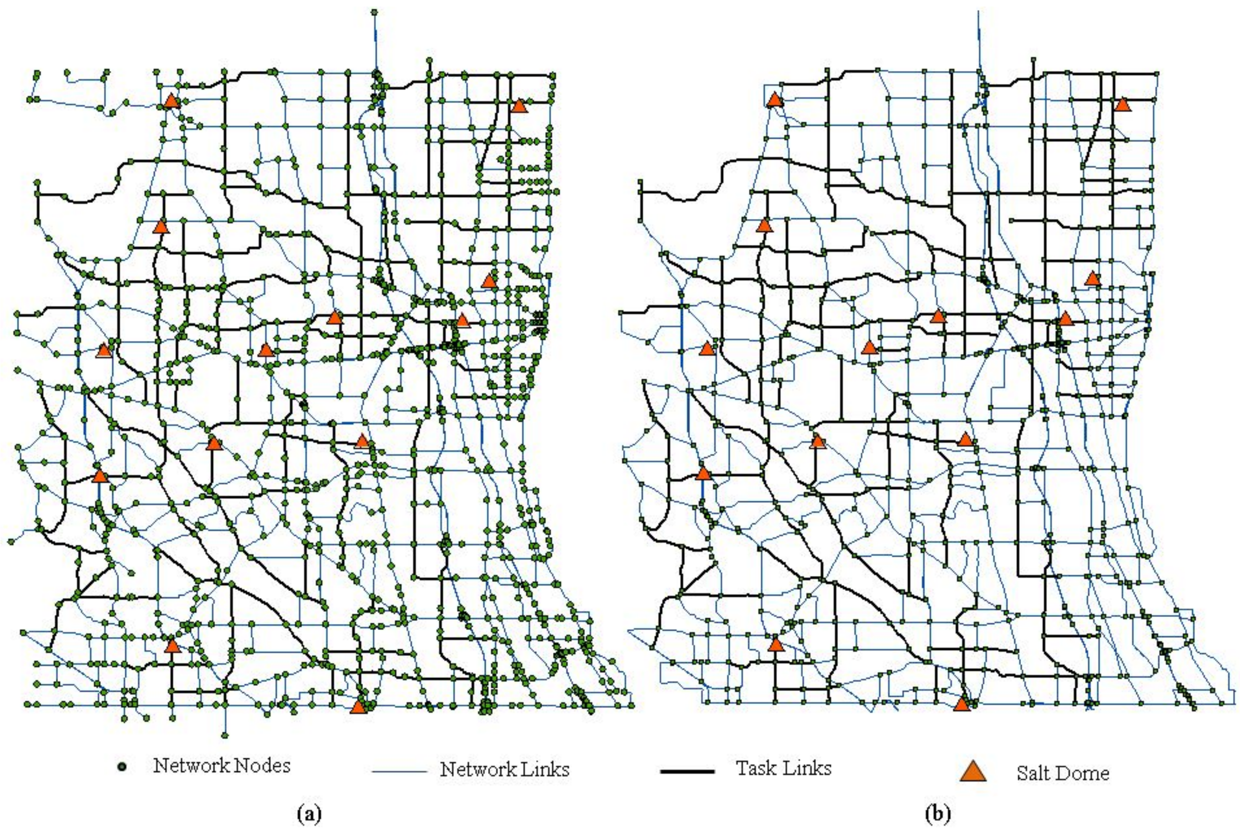


Figure 3.5: Lake County network; (a) original network and (b) cleaned network after end node and mid-block node removal.

Table 3.1: Data treatment summary.

number	original data	after end node removal	after mid-block removal	after intersection representation
nodes	1354	1227	902	5884
links	3350	3189	2539	11240
task links	816	816	706	706

Scenario 1: No Priority

The Lake County currently does not specify priority for the snow routes in its jurisdiction. A fleet of $|K| = 25$ snow plow trucks (out of 2 different truck types) is used, including 6 trucks with salt carrying capacity $L_k = 7.5\,tons$ and $MPG_k = 6\,miles/gallon$, and 19 trucks with salt carrying capacity $L_k = 12\,tons$ and $MPG_k = 5\,miles/gallon$. In this section we assume the same class of priority for all plowing task links and let $P = \{1\}$. With the objective of balancing truck travel times, we assume $c_{Fuel} = 1$ and $c_{LOS,1} = 10$. In addition, a salt replenishment rate of $0.2\,tons/mile$ is used. All other input data, e.g., the vehicle speed, length, travel time of all roadway links, the plow time and the number of passes on all task links, the turning delays and u-turn permits at all intersections, salt replenishment time, and locations of salt domes are obtained from existing LCDOT databases. Mid-day is considered as the time-of-day for both deadhead travel time and plow time.

The snow plow routing optimization problem is solved using the methodology in Section 3.3. The algorithms are coded in Visual C++ and run on a desktop computer with $2.00\,GHz$ CPU and $8.00\,GB$ memory. The results (based on 60 min of computation time) show that the solution algorithms can effectively optimize the snow plow routes and their performance measures. Table 3.2 presents the route statistics, i.e., the total plowed distance, total plowed time, total deadhead distance, total deadhead time, total distance, and total time, based on a "semi-manual" solution and the optimal solution. To evaluate the results of the optimization approach, we compare our model solution with that from the semi-manual method by LCDOT. LCDOT do not have any historical records on the exact routing of its trucks, therefore the actual travel cost is unknown and it is expected to be high as the trucks do not move on the optimal routes. However, LCDOT does have records on the assignment of the task links to trucks, which is determined manually based on a tabular summation of route segments including deadhead travels and plow distances (both in *miles*) in a cycle. Then, the sequence of plowing tasks and salt domes visits is solved independently for each truck using our TSP algorithm. Hence, the semi-manual approach does include optimization. Note that the proposed optimization approach has improved the semi-manual solution deadhead distance and time by 4.1% and 3.5% in one cycle from the depot and back. These percentages are quite significant considering the fact that many snow plows must travel significant portions of distance from and to the depot; and, the snow plowing cost is huge, hence even 1 % improvement can save a lot for the agency. By the way, as discussed above, these comparisons are not with the solutions to the pure manual method in practice.

Table 3.2: Comparison to the Manual Solution.

solution method	total plowed dist. (*miles*)	total plowed time (*min*)	total deadhead dist. (*miles*)	total deadhead time (*min*)	total dist. (*miles*)	total time (*min*)
semi-manual	832.290	1780.871	516.271	979.848	1348.560	2760.718
optimal	832.290	1780.871	495.969	946.292	1328.259	2727.163
% diff.	-	-	4.1 %	3.5 %	1.5 %	1.2 %

Furthermore, to further test the solution algorithms, sensitivity analyses are conducted to study the impacts of the weights in the objective function on the performance of the overall design. The parameter values are assumed as follows. Morning peak hour is considered as the time-of-day for both travel time and plow time. A set of different values for the weights in the objective function are tested, i.e., c_{Fuel} and $c_{LOS,1} \in \{1, 10, 100\}$. The salt application rate is $0.125\, tons/mile$. A fleet of $|K| = 25$ snow plow trucks (out of 4 different truck types) is used, each with the salt carrying capacity $L_k = 7, 7.5, 8, 11\, tons$ and $MPG_k =$

$9, 10, 11, 12\, miles/gallon$. All other input data are obtained from the LCDOT databases.

Table 3.3 shows the longest individual cycle time and the total deadhead time after 60 *min* of computation time, for each scenario. Scenarios (a)-(b) demonstrate the effect of input parameters, $c_{LOS,1}$ and c_{Fuel}, on the optimal route performance.

Table 3.3: LCDOT Network Results.

case number	c_{Fuel}	$c_{LOS,1}$	z_1 (min)	total deadhead time (min)	iteration number
a	100	1	234.64	3388.33	17,677
b	1	1	195.63	3389.55	17,398
c	1	10	179.65	3452.61	23,726
d	1	100	169.42	3454.43	16,055

Figure 3.6 depicts the sensitivity of the solutions to the selection of the weights of the objective function. Note that, as expected, the higher the ratio of the $c_{LOS,1}/c_{Fuel}$, the more balanced the truck travel times.

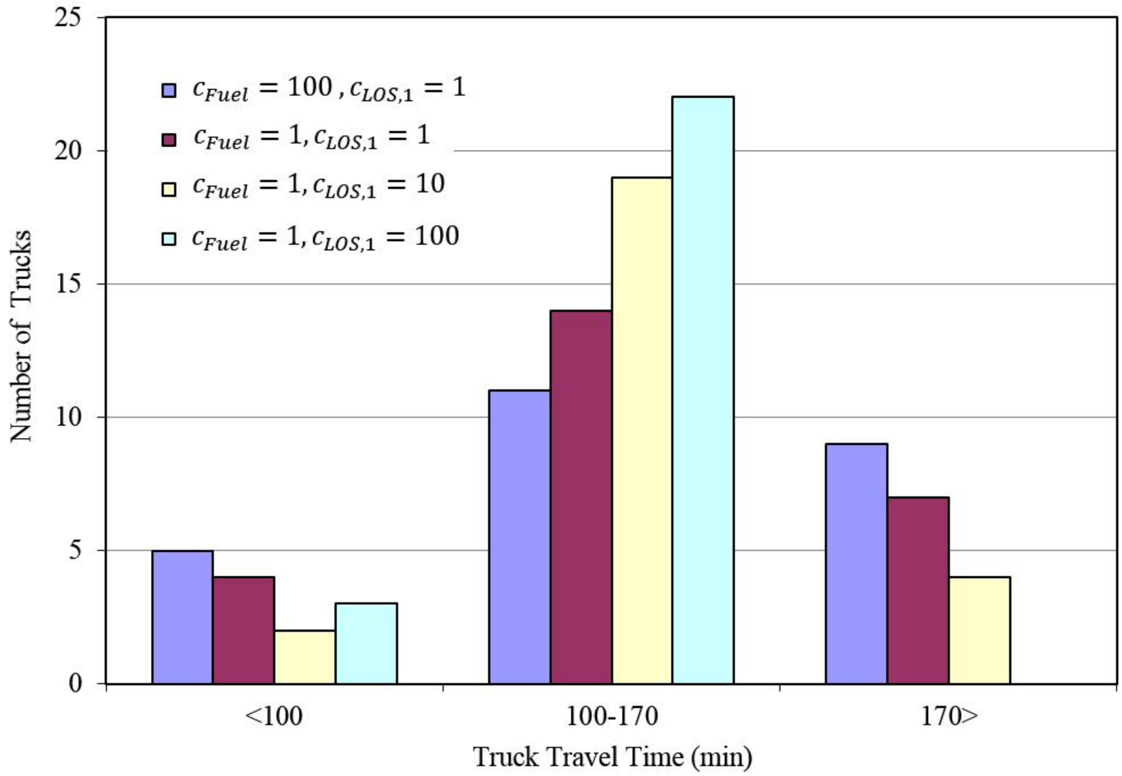


Figure 3.6: Sensitivity of the truck cycle time distribution to the objective weights.

On the other hand, another study has been conducted to see the remaining salt in the trucks before visiting salt domes for resource replenishment. We solve the problem for a higher salt application rate to use resources up to trucks' capacity (alternatively, the problem can be solved for a long-storm scenario (i.e., multiple cycles) as well). In this test, we assume a salt application rate of $0.8\,tons/mile$ and $c_{Fuel} = c_{LOS,1} = 1$. Again, fleet of $|K| = 25$ snow plow trucks (out of 4 different truck types) is used, each with the salt carrying capacity $L_k = 7, 7.5, 8, 11\,tons$ and $MPG_k = 9, 10, 11, 12\,miles/gallon$. All other input data are obtained from the LCDOT databases.

Figure 3.7 shows salt levels in each truck before replenishments. Letters a to y represent the trucks' indices before each salt dome facility visit. Each truck can visit the facilities more than once. Note that despite holding a high salt level (e.g., more than half of trucks' capacity), a truck may still replenish salt. This often occurs when the next task link that a truck plows is long and the truck would run out of salt before finishing the task.

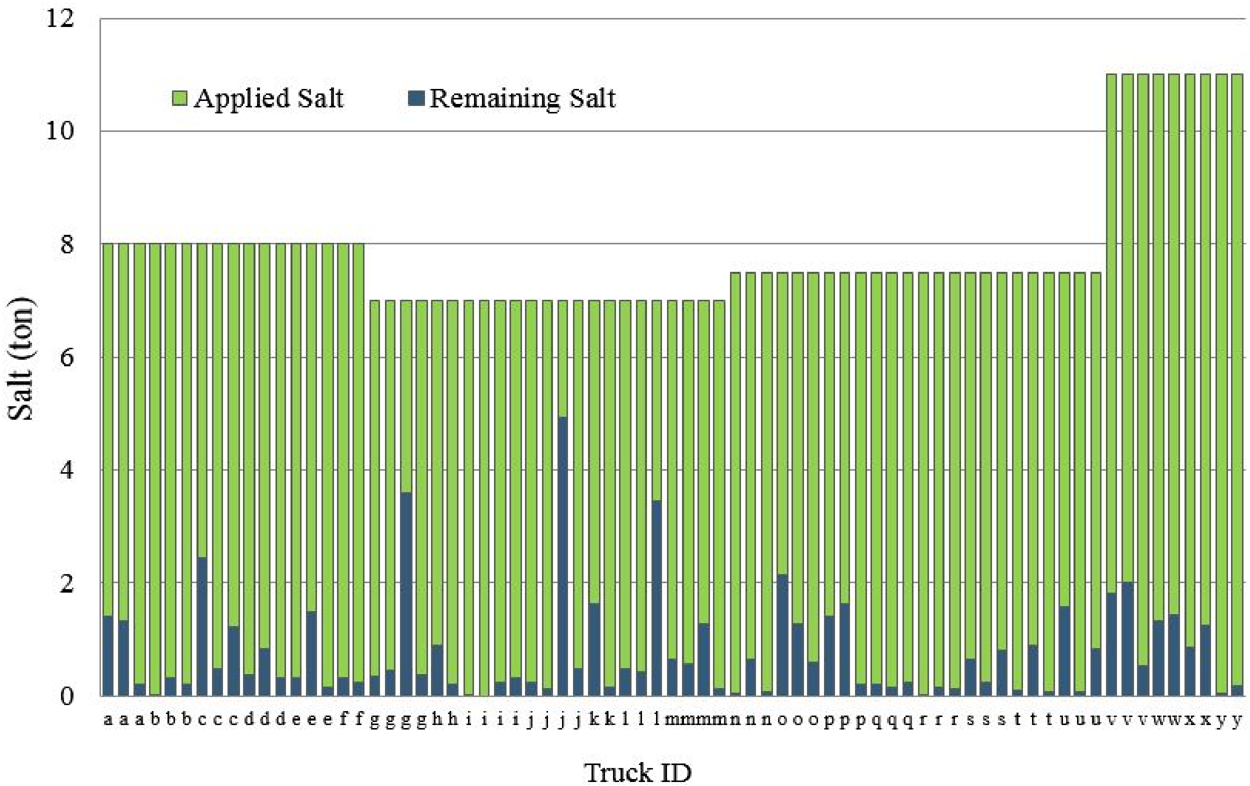


Figure 3.7: Salt level in trucks before replenishment.

Scenario 2: Multiple Priority Classes

This section presents the numerical results of applying model (3.2a) - (3.2l) to a hypothetical case where some of the roadway segments are given priority. We still use data from Lake County, Illinois. Since there is no priority among the plowing links under the jurisdiction of LCDOT, we prioritize links based on the level of daily traffic volume. We assume there are two classes of priority, $p \in \{1, 2\}$. High priority is given to about 12 % of the total task links which have high traffic volume in a morning peak hour from historical data. It turns out that these high priority links are mostly near interstate highway I-94 and the main streets in the network. Following our solution algorithm in Section 3.3, we include in the objective the maximum time needed to complete high-priority task links, with weights $c_{LOS,1} = 20$ and $c_{LOS,2} = 10$.

Using the same input data and parameter values as in Section 2, we find from the numerical results that the total deadhead distance and deadhead time increase to 588.538 *miles* and 1107.536 *min*, respectively. Such reduction in efficiency is expected, however, as we now give priority to some of the links (rather than trying to plow all the links within the minimum distance/time).

3.4 Conclusion

Winter maintenance on roadways involves truck routing, plowing, salting, and salt replenishment. Total deadhead time and the operation time of each truck is affected by the route choices and time-of-day traffic congestion. It is important and yet challenging to find the optimum solution to the snow plow routing problem.

In this chapter, a mixed integer linear program is proposed to simultaneously minimize the total travel time of all snow plow trucks and the longest individual operation time under multiple plowing priority classes. A set of heuristic methods are developed and applied to case studies to verify the model formulation, algorithms, and to draw managerial insights. The computational results show that the proposed solution approach is capable of solving the problem effectively.

Future research can be generalized in a few directions. In this chapter, only one cycle routing is formulated for each vehicle in a short snow storm. A possible extension is to develop a model formulation for a long-storm scenario under two possible options: (*i*) having exactly the same truck path sequence in different cycles, and (*ii*) enforcing a minimum time separation between two consecutive plowing of the same task links. In addition, in this research the possibility of system breakdown (e.g., satellite salt depot disruption or truck failure) is ignored; it would be interesting in the future to solve snow plow routing problem under stochastic reliability constraints. It would also be interesting to apply our model and solution approach to a comprehensive list of diverse networks for additional real-world case studies.

Acknowledgment

This research was financially supported in part by the Lake County Division of Transportation, Lake County, Illinois. We thank Martin Buehler, Al Giertych, Paula Trigg, Michael Scott, and Edwin Montano (LCDOT) for data support and helpful feedback.

Chapter 4

Dynamic Snow Plow Fleet Management under Uncertain Demand and Service Disruption

4.1 Introduction

This chapter addresses a fleet assignment problem for snow plow trucks under service disruptions and demand uncertainty.[1] Figure 4.1 illustrates the problem over a planning horizon of discrete time periods (e.g., each for 30 *min* or an hour). Plowing or salting demand (which we call "tasks") may include a set of roadway links along a number of designated snow routes that can be plowed within exactly one time period, and repeated demand may arise periodically or randomly over time. At the beginning of the planning horizon, we assume that a fleet of identical plow trucks are available at the main depot. They can be dispatched dynamically to complete the service tasks across the snow routes during the planning horizon. The completion of each task requires a truck to cover a particular segment of snow routes (i.e., location) at a particular time period.

[1] Hajibabai, L. and Ouyang, Y. (2014). Dynamic Snow Plow Fleet Management under Uncertain Demand and Service Disruption. Submitted to Transportation Research Record: Journal of the Transportation Research Board.

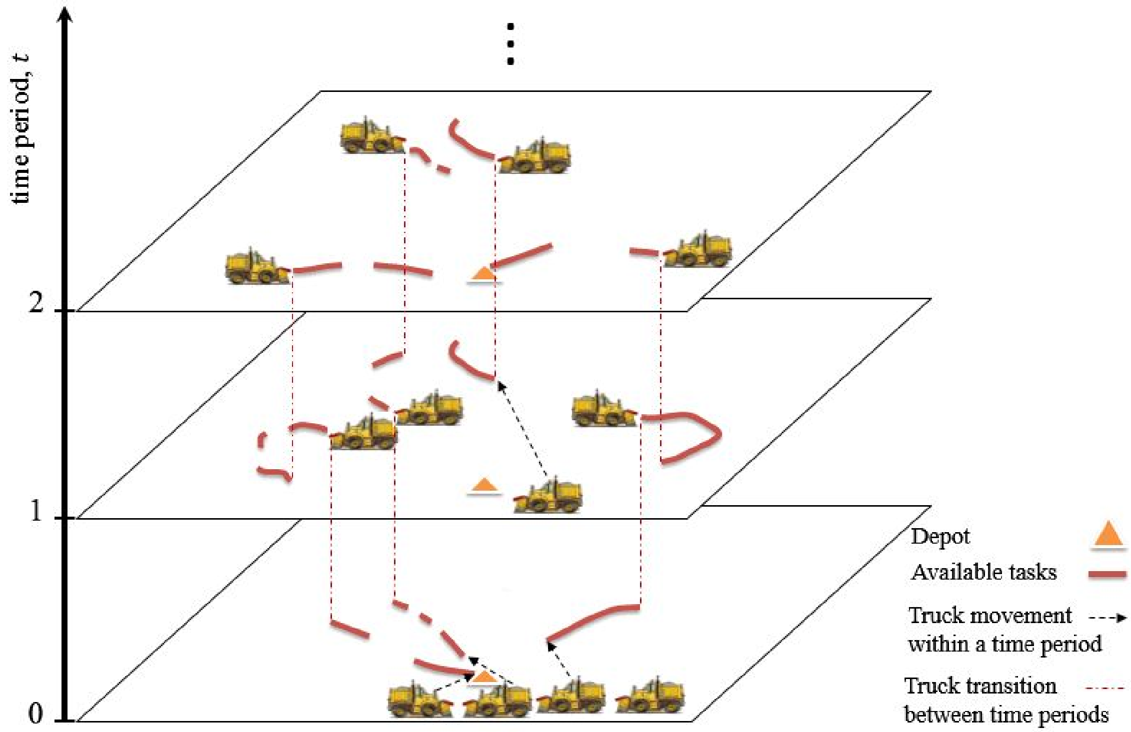


Figure 4.1: Hypothetical illustration of the task availability and trucks movement over time.

At the beginning of each time period, decisions on fleet assignment to the tasks are made according to spatial and temporal distribution of the maintenance demand. Figure 4.2 shows the relationships among the spatial and temporal distribution of the tasks and fleet management in the network.

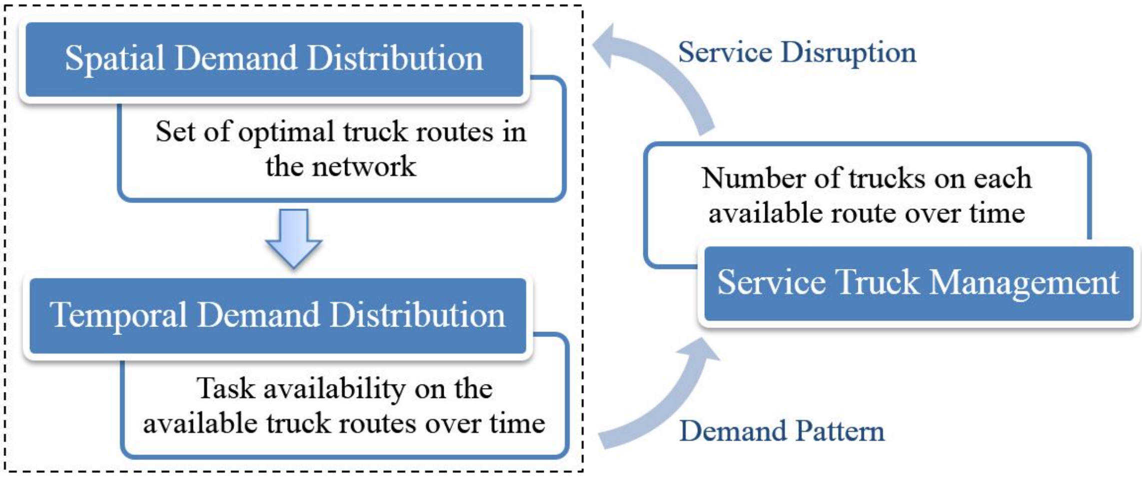


Figure 4.2: Inter-relationships among demand distribution and stochastic dynamic fleet management.

This problem involves several challenges. As aforementioned, additional tasks are arising either periodically or randomly over time and space subject to progression of the snow storm. The operations of trucks are sometimes subject to failure, possibly due to traffic congestion, mechanical breakdowns, or incidents. Some tasks, especially those on critical roadway links (such as emergency routes), often have priority and impose a strict service time window constraints (i.e., one or more time periods during which these tasks could be completed) to the fleet schedule. Violation to these time windows may result in severe penalties. In addition, in case service disruption occurs, the backlogged tasks will be counted as new tasks that must be addressed in the next time period. A truck in one specific location and time can be dynamically "repositioned" to a different location, but at a cost. This could occur to in-service trucks which would leave currently assigned tasks in order to serve potentially high priority regions (especially in case of service disruptions), replenish salt or chemicals, etc.). This could also occur to idling vehicles in anticipation of future tasks in certain regions. For simplicity, we assume that all truck repositioning requires exactly one time period.

The remainder of this chapter is organized as follows. Section 4.2 presents the mathematical formulation of dynamic fleet management model under uncertainty. Section 4.3 introduces the solution technique and Section 4.4 presents the numerical results and discusses managerial insights. A discussion on the summary and future research directions is presented in Section 4.5.

4.2 Model Development

This section introduces a dynamic programming model for the stochastic snow plow fleet management problem. We first define the physical and temporal elements of the problem as follows. Let Ψ be the set of physical snow routes in the transportation network, each of which generates a set of service tasks over time, T be the number of discrete time periods in the planning horizon, and $\Gamma = \{0, 1, \cdots, T-1\}$ be the times at which decisions are made.

For each $t \in \Gamma$, $i \in \Psi$, let $\widehat{\mathcal{K}}_{it}$ denote the number of trucks that first become available on route i at time t, and $\widehat{\mathcal{K}}_t = (\widehat{\mathcal{K}}_{it})_{i\in\Psi}$ describe the spatial distribution of all newly realized trucks at time t. Let $\mathcal{K}_{it}$ be the number of trucks available on route i at time t before any new arrivals (e.g., from other routes) or failures, and $\mathcal{K}_t = (\mathcal{K}_{it})_{i\in\Psi}$ be the total number of trucks that are already available at time t. Then, $\mathcal{K}_t^+$ presents the total number of trucks that are available to be dispatched in time period t, where $\mathcal{K}_t^+ = \mathcal{K}_t + \widehat{\mathcal{K}}_t$ that includes all existing as well as newly realized trucks. Similarly, to keep track of the available tasks, for each $t \in \Gamma$, we define $\widehat{\mathcal{A}}_t$ to denote the set of tasks that first become available in time period

t, and $\mathcal{A}_t$ to be the tasks available at time t before the new arrivals in $\widehat{\mathcal{A}}_t$ are added to the system. Then, $\mathcal{A}_t^+$ presents the set of tasks available to be served at time t, including the new tasks that just arrived in this period, i.e., $\mathcal{A}_t^+ = \mathcal{A}_t \cup \widehat{\mathcal{A}}_t$. Thus, $\mathcal{A}_{it}^+$ denotes the set of tasks that must be serviced by trucks on route i at time t, i.e., $\mathcal{A}_t^+ = \bigcup_{i \in \Psi} \mathcal{A}_{it}^+$. In addition, $\mathcal{A}_{it}^d$ is the set of deadhead links required to reach the available tasks on route i at time period t.

Furthermore, we let Γ_a represent the time window for task $a \in \mathcal{A}_t^+$ to be completed. If any task a is not served within Γ_a, it is assumed lost in that time period but backlogged for the future. If the time window Γ_a is tight (e.g., only a single time period), $\mathcal{A}_t$ will be empty, meaning that there will be no tasks left over from time period t for the next time period. We let $\mathcal{W}_t = (\widehat{\mathcal{K}}_t, \widehat{\mathcal{A}}_t)$ represent the new information arriving in time period t, where $(\mathcal{W}_t)_{t=0}^T$ denotes our stochastic information process with realization $\mathcal{W}_t(\omega) = \omega_t = (\widehat{\mathcal{K}}_t(\omega), \widehat{\mathcal{A}}_t(\omega))$.

The relevant state of our system is captured by the distribution of available trucks and tasks over a set of routes, i.e., $\mathcal{S}_t = \{\mathcal{K}_t, \mathcal{A}_t\}$. In practice, $\mathcal{A}_t$ is relatively small since it represents the set of tasks left over from a previous time period and in special cases, where the time window is tight, we omit it from the state definition.

At each time period t, a truck could do one of three things: serve a set of tasks, reposition, or sit idle. A truck can be repositioned by moving it (i) from one route to another, (ii) within one route (i.e., only deadheading on a route during a time period), or (iii) from a route to a salt dome for resource replenishment (and vice versa). We assume that the time required to move between any pair of routes is a single time period. The decision to sit idle is basically equivalent to reposition from route i to the same route i. Note that this decision is not favorable in the context of snow plow fleet management since we are not willing to waste truck resources while there are tasks to perform. For each $t \in \Gamma$ and $i, j \in \Psi$, decision variable η_{ij}^t denotes the number of trucks repositioned from $i \in \Psi$ to $j \in \Psi$ at time $t \in \Gamma$, and variable μ_i^t represents the presence of a truck on snow route i at time t, which equals 1 if a truck travels on route $i \in \Psi$ at time $t \in \Gamma$, or 0 otherwise. All decisions in period t are captured in vectors $\mu^t \equiv (\mu_i^t)_{i \in \Psi}$ and $\eta^t \equiv (\eta_{ij}^t)_{i,j \in \Psi}$.

A number of cost parameters are defined for the system. We let c_{ij}^p denote the cost of repositioning one truck from route i to route j, c_a^r be the reward for plowing task $a \in \mathcal{A}_{it}^+$, and c_a^d be the cost of deadheading through link $a \in \mathcal{A}_{it}^d$. The decision to sit idle on route i incurs a cost c_{ii}^p. Repositioning and deadheading cost coefficients can be defined as functions of distance, resource load in the truck, etc. Let $f_t(\mu_t, \eta_t)$ be the one-period reward function, which represents the total benefit gained from the decisions made at time t, and it can be written as follows.

$$f_t(\mu_t, \eta_t) = \sum_{i \in \Psi} \left\{ \mu_i^t \left(\sum_{a \in \mathcal{A}_{it}^+} c_a^r \lambda_{i,r}^t - \sum_{a \in \mathcal{A}_{it}^d} c_a^d \lambda_{i,d}^t \right) - \sum_{i \in \Psi} \sum_{j \in \Psi} c_{ij}^p \, \eta_{ij}^t \right\}, \tag{4.1a}$$

subject to

$$\sum_{j \in \Psi} \eta_{ij}^t(\omega) + \sum_{i \in \Psi} \mu_i^t(\omega) = \mathcal{K}_{it} + \widehat{\mathcal{K}}_{it}, \; \forall i \in \Psi, \tag{4.1b}$$

$$\mu_i^t(\omega) \in \{0, 1\}, \forall i \in \Psi, \text{ and} \tag{4.1c}$$

$$\eta_{ij}^t(\omega) \geq 0, \forall i, j \in \Psi \,, \tag{4.1d}$$

where $\lambda_{i,r}^t$ and $\lambda_{i,d}^t$ are the number of tasks and deadheads required to reach the tasks on route i at time t. Constraints (4.1b) enforce the conservation of truck flow since each truck on route i at time t must either serve a set of tasks by traveling on a route (μ_i^t) or be repositioned (η_{ij}^t). Constraints (4.1c) and (4.1d) ensure the binary and non-negativity requisites of decision variables.

In order to further express the dynamics of the system over time, we define $\mathcal{A}_t^e(\mu_t)$ to denote the set of expired tasks that are either served or never served by the end of the time window. Thus, the dynamics of the tasks and trucks are

$$\mathcal{A}_{t+1} = \mathcal{A}_t^+ \setminus \mathcal{A}_t^e, \text{ and} \tag{4.2}$$

$$\mathcal{K}_j^{t+1}(\omega) = \sum_{i \in \Psi} \eta_{ij}^t(\omega) + \sum_{j \in \Psi} \mu_j^t(\omega), \; \forall j \in \Psi, \tag{4.3}$$

where (4.3) counts how many trucks move to route j at time $t+1$. Our objective is to maximize the expected benefits over the planning horizon, given an initial state $\mathcal{S}_0$, as follows.

$$\underset{\mu_0,\eta_0}{\text{maximize}}\, f_0(\mu_0,\eta_0) + \mathbb{E}\left\{\sum_{t\in\Gamma\setminus\{0\}} \underset{(\mu_t,\eta_t)}{\text{maximize}}\, f_t(\mu_t,\eta_t)\right\}. \tag{4.4}$$

Thus, the overall optimization problem is represented by (4.1b)-(4.1d), (4.2)-(4.3), and (4.4).

4.3 Solution Technique

We use an ADP approach with CAVE updates (Godfrey and Powell, 2002a) to solve problem (4.1b)-(4.1d), (4.2)-(4.3), and (3.2a). In so doing, we first rewrite the objective function (3.2a) into equivalent Bellman's equations,

$$V_t(\mathcal{S}_t) = \mathbb{E}\{\underset{(\mu_t,\eta_t)}{\text{maximize}}\, f_t(\mu_t,\eta_t) + V_{t+1}(\mathcal{S}_{t+1})|\mathcal{S}_t\}.$$

The corresponding problem for a single sample realization is

$$V_t(\mathcal{S}_t,\omega) = \underset{(\mu_t(\omega),\eta_t(\omega))}{\text{maximize}}\, f_t(\mu_t(\omega),\eta_t(\omega)) + V_{t+1}(\mathcal{S}_{t+1}(\omega)).$$

Our problem, in its current form, suffers from a huge state space (i.e., too many combinations of total available trucks and tasks over all times $t \in \Gamma$ on all snow routes $i \in \Psi$). Recall that we exclude $\mathcal{A}_{t+1}$ from the state variable definition. We further substitute our value function $V_{t+1}(\mathcal{S}_{t+1})$ with an approximation $\widehat{V}_{t+1}(\mathcal{K}_{t+1})$, which is only a function of the available trucks vector. Our problem for one sample realization is now written as

$$\widetilde{V}_t(\mathcal{K}_t, \omega) = \underset{(\mu_t(\omega), \eta_t(\omega))}{\text{maximize}}\, f_t(\mu_t(\omega), \eta_t(\omega)) + \widehat{V}_{t+1}(\mathcal{K}_{t+1}(\omega)), \tag{4.5}$$

subject to

$$(4.1\text{b}) - (4.1\text{d}), \text{ and } (4.3).$$

In the above, we use a concave piece-wise linear approximation $\sum_{i\in\Psi} \widehat{V}_{it}(\mathcal{K}_{it})$ for estimating $\widehat{V}_t(\mathcal{K}_t)$, which is more flexible and effective than linear approximations. Equations (4.1b)-(4.1d), (4.3), and (4.5) represent a network flow problem and the solution to the approximated sub-problem will be integer valued if we guarantee that the break-points of the piece-wise linear function $\widehat{V}_{it}(\mathcal{K}_{it})$ occur at integer values of $\mathcal{K}_{it}$.

4.3.1 Algorithm Framework

The general algorithm includes (i) a forward simulation and (ii) a value function update procedure. In the forward simulation, we generate a random future outcome ω and solve a sequence of network sub-problems (4.1b)-(4.1d), (4.3), and (4.5) for $t = 0, 1, \cdots, \Gamma - 1$, using the current approximation of the expected value function. We then use the dual sub-gradient information π_{it}^{+} and π_{it}^{-} derived from solving our dynamic fleet management problem (i.e., the network sub-problems) to update the value function approximations $\widehat{V}_{it}(\mathcal{K}_{it})$ using the CAVE algorithm, and we ensure the concavity of the estimation at each iteration n (see Godfrey and Powell (2001, 2002a) for more detail). These stochastic sub-gradients are basically the flow-augmenting and flow-decrementing dual values and provide the marginal benefit (or loss) of one more (or one fewer) truck on each snow route.

Figure 4.3 summarizes the steps of our general algorithm framework for the multi-stage stochastic value function approximation. In the CAVE algorithm, three parameters control the amount of change in the piece-wise approximation $\widehat{V}(\mathcal{K}_{it})$ at each iteration: (i) parameter ε^-, (ii) parameter ε^+, where $[\mathcal{K}_{it} - \varepsilon^-, \mathcal{K}_{it} + \varepsilon^+)$ is the smallest updating range for the piece-wise approximation, and (iii) updating parameter α for the segment slopes. We first initialize these updating parameters and get the dual stochastic sub-gradients π_{it}^- and π_{it}^+ from solving the problem (4.1b)-(4.1d), (4.3), and (4.5) in an iteration of the algorithm for a given ω and an estimation of the value function approximation. We then update the value function approximation following the CAVE algorithm.

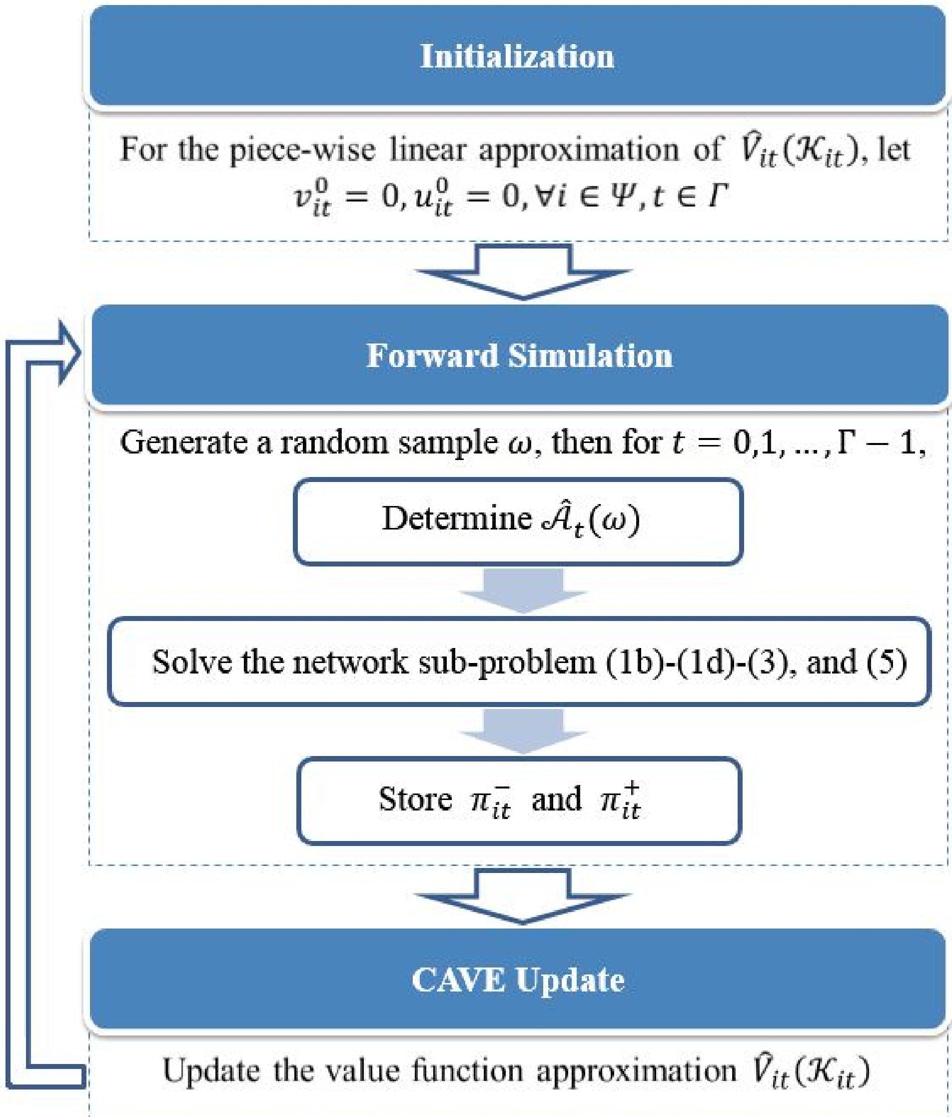


Figure 4.3: The multi-stage stochastic value function approximation algorithm; general framework (Godfrey and Powell, 2002a).

Figure 4.4 illustrates the embedded CAVE procedure. Given the state value on each route i at each time t at an iteration of the algorithm, we define the update interval for the value function approximation on each route i at each time t, and create new break points. In so doing, we compare the sub-gradient values π_{it}^- and π_{it}^+ with the segment slopes of the approximation at $(n-1)^{th}$ iteration on each route i at time t. We determine how many segments have slopes less than or equal to the dual sub-gradients (i.e., determine n_{it}^- and n_{it}^+). Then, we compare the break points at n_{it}^- with $\mathcal{K}_{it} - \varepsilon^-$ and break points at n_{it}^+ with $\mathcal{K}_{it} + \varepsilon^+$, and extend the update interval $[\mathcal{K}_{it} - \varepsilon^-, \mathcal{K}_{it} + \varepsilon^+)$ if necessary. Afterwards, we perform the update procedure over the update interval by comparing the stochastic sub-gradients' values with the slopes of the current approximation on both sides of the state value on route i at time t to maintain concavity (i.e., accepting larger slopes on the left and smaller ones on the right-hand-side of the state value break point). At each iteration n, declining step-size rules are applied to parameters ε^-, ε^+, and α for stability. It can be proved that the concavity of the approximation is preserved under the updating rules (Godfrey and Powell, 2001). To guarantee integer solutions from our approximation, parameters ε^- and ε^+ need to be defined as positive integers so that the break-points always occur at integer values of $\mathcal{K}_{it}$.

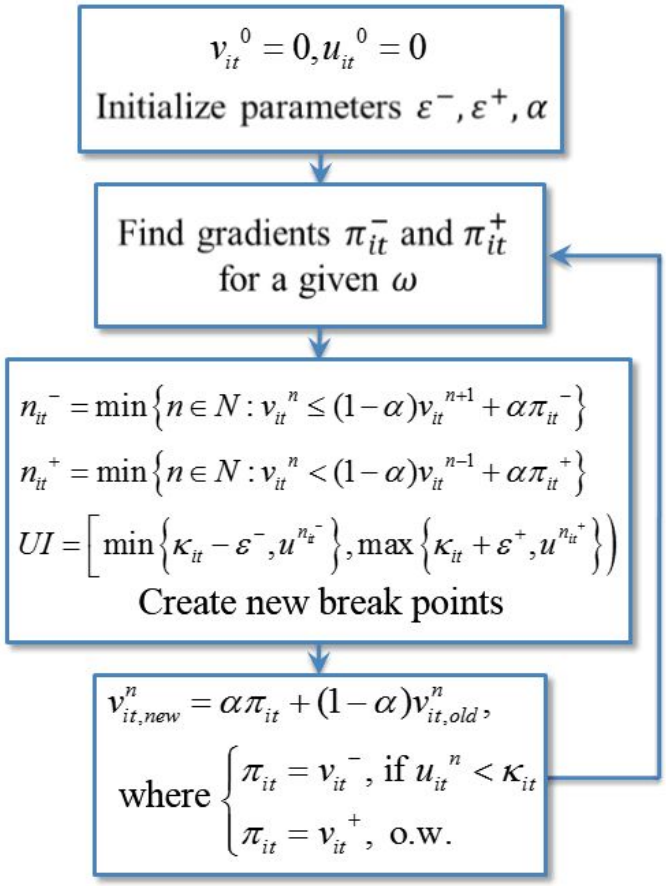


Figure 4.4: The multi-stage stochastic value function approximation algorithm; CAVE procedure (Godfrey and Powell, 2002a).

An illustration of the updating procedure for a particular state and set of gradient estimates appear in Figure 4.5 and 4.6. For simplicity, we assume that $\alpha = 1$ in both examples. In Figure 4.5, $\pi^-(\mathcal{S}) \geq v^2$ and $\pi^-(\mathcal{S}) \geq v^3$, therefore $n^- = 2$. However, $\pi^+(\mathcal{S}) \geq v^3$, so $n^+ = 3$. Thus, we stretch the updating range from $[\mathcal{S} - \varepsilon^-, \mathcal{S} + \varepsilon^+)$ to include $u^{n^+} = u^3$ on the right side to maintain concavity, leading to the update range of $[\mathcal{S} - \varepsilon^-, u^3)$. New break-points are added at $\mathcal{S} - \varepsilon^-$ and $\mathcal{S}$, where both are between u^1 and u^2, with slopes initialized to v^1. In this example, updating decreases all of the slopes in UR. Similarly, in

Figure 4.6, $n^- = 1$ and $n^+ = 2$, hence we stretch the updating range $[\mathcal{S} - \varepsilon^-, \mathcal{S} + \varepsilon^+)$ to include $u^{n^-} = u^1$ on the left side, which leads to UR=$[u^1, \mathcal{S} + \varepsilon^+)$. Then, we insert new break-points at $\mathcal{S}$ and $\mathcal{S} + \varepsilon^+$ with initial slopes v^1 and v^2, respectively. Here, updating increases the slopes over $[u^1, \mathcal{S}) \bigcup [u^2, \mathcal{S} + \varepsilon^+)$ and decreases the slopes over $[\mathcal{S}, u^2)$.

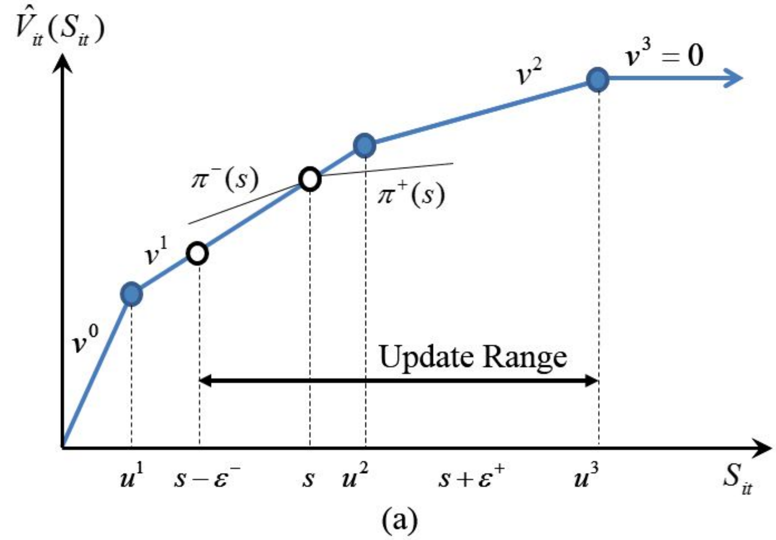


Figure 4.5: Example of CAVE update; example 1 [source: Godfrey and Powell (2001)].

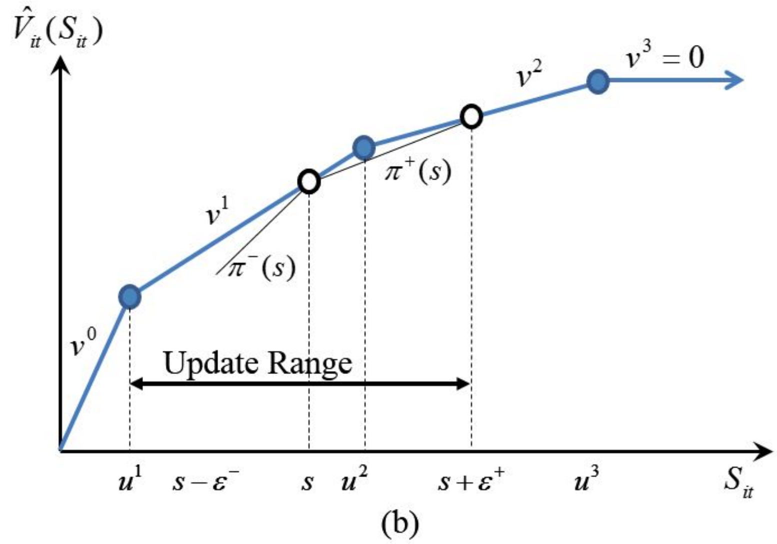


Figure 4.6: Example of CAVE update; example 2 [source: Godfrey and Powell (2001)].

4.3.2 Network Representation of One-Step Decisions

A two-stage time-space network illustration of the decisions at time t is presented in Figure 4.7. In the first stage of time t, each truck either travels to a route to serve available tasks, or is repositioned, or sits idling. Each node on the left side of the network represents a route i with available truck supply $\mathcal{K}_{it}$. From each origin node, three types of arcs (i.e., service arcs on each route, repositioning arcs between routes within an allowable travel distance , and idling arcs to the same route) are constructed. The cost on the task arcs is $\sum_{a\in\mathcal{A}_{it}^{+}} c_a^r\lambda_{i,r}^t - \sum_{a\in\mathcal{A}_{it}^{d}} c_a^d\lambda_{i,d}^t$ and their upper capacity is $|\mathcal{A}_{it}^{+}(\omega)|$. The repositioning arcs have cost $-c_{ij}^p$ and upper capacity $+\infty$.

For each time period $t \in \Gamma$, we approximate $V_t(\mathcal{K}_t)$ by a summation of one-dimensional separable functions $\widehat{V}_t(\mathcal{K}_t) = \sum_{i\in\Psi}\widehat{V}_{it}(\mathcal{K}_{it})$ from CAVE. We actually incorporate the CAVE update $\widehat{V}_{j,t+1}$ into the second stage decisions of the network sub-problem (time $t+1$). For each piece-wise linear segment in $\widehat{V}_{j,t+1}$, we construct an arc from each node $(j, t+1)$ to a sink node (Figure 4.7). In our network model (first stage), we do not constrain the repositioning flow between route i and j at time t, however, the CAVE algorithm (second stage) limits the number of repositions instead. In other words, since slopes $v_{j,t+1}^n$ decrease monotonically over iteration n, all trucks move to the destination node in the order determined by $\widehat{V}_{j,t+1}$. This means that it is optimal to let trucks flow through the link with the highest marginal benefit until its upper capacity is reached, and then through the link with the next highest marginal benefit, etc. This is the key reason for maintaining concavity in the CAVE algorithm.

Finally, the state $\mathcal{S}_t^+$ is updated to $\mathcal{S}_{t+1}^+$ by solving the time-t network sub-problem in the first stage, updating the value function estimation in the second stage, and then adding new tasks and trucks (including truck failures) from $\widehat{\mathcal{A}}_{t+1}$ and $\widehat{\mathcal{K}}_{t+1}$. We keep generating and solving this sequence of sub-problems until $t = T - 1$ is reached. Further iterations are run using new random samples as desired. The proposed network configuration leads to fast integer optimal solutions.

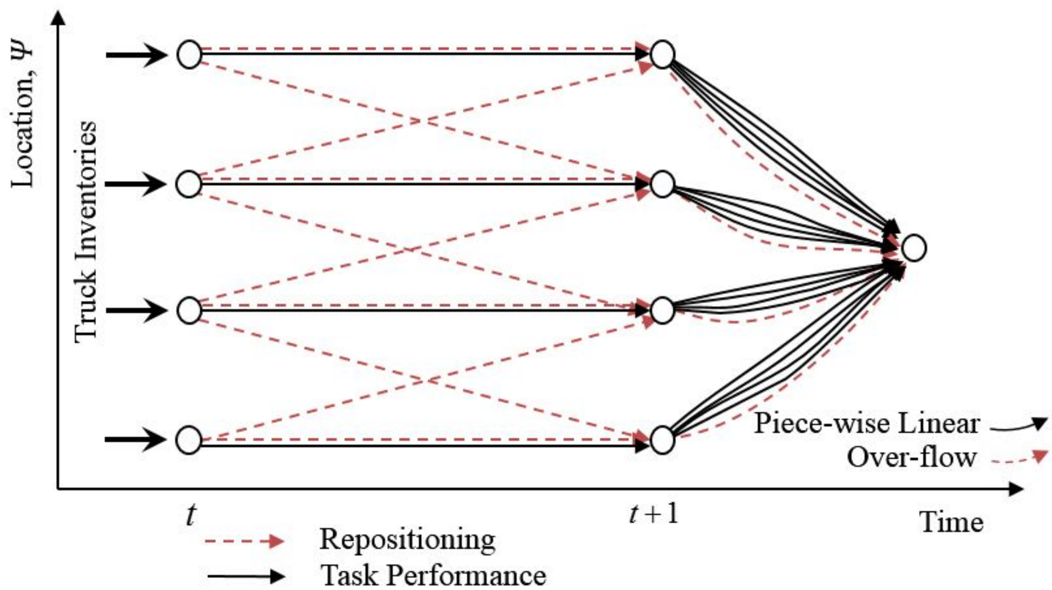


Figure 4.7: Illustration of the sub-problem at time t in two stages: first stage (time t decisions; left side) and second stage (time $t+1$ decisions; right side) (Godfrey and Powell, 2002a).

4.4 Numerical Results

The proposed solution approach in Section 4.3 is applied to a snow plow operation scenario for Lake County, Illinois. Figure 4.8 shows the existing snow links, salt domes, and the trucks depot (i.e., salt dome number 1) in the network, per Hajibabai et al. (2014b). A planning horizon of 160 minutes, which equals to the longest cycle time across all snow routes from/to the depot, is chosen. It is split into 20 *min* time periods (i.e., 8 time periods in total).

Table 4.1 also presents other data parameters we use to construct our experiments. We assume that at each iteration n of the algorithm, tasks arise randomly (i.e., following a Poisson random process) over a set of at most 25 designated snow routes . We assume that any truck may serve any set of tasks on route i at time t. Serving tasks generates benefit and it simultaneously incurs a cost that is associated with truck movement on the deadhead links. The completed tasks will be eliminated from the set of available tasks in the next time period. Trucks may fail (following an independent random process) to serve tasks on any route i and any time t. In case service disruption occurs on a route i and a time t, the backlogged tasks will be counted as new tasks in the next time period $t + 1$. If no task is available on a route i at a time period t (i.e., all deadhead travels over a time period t), traveling on that route i is counted as a repositioning.

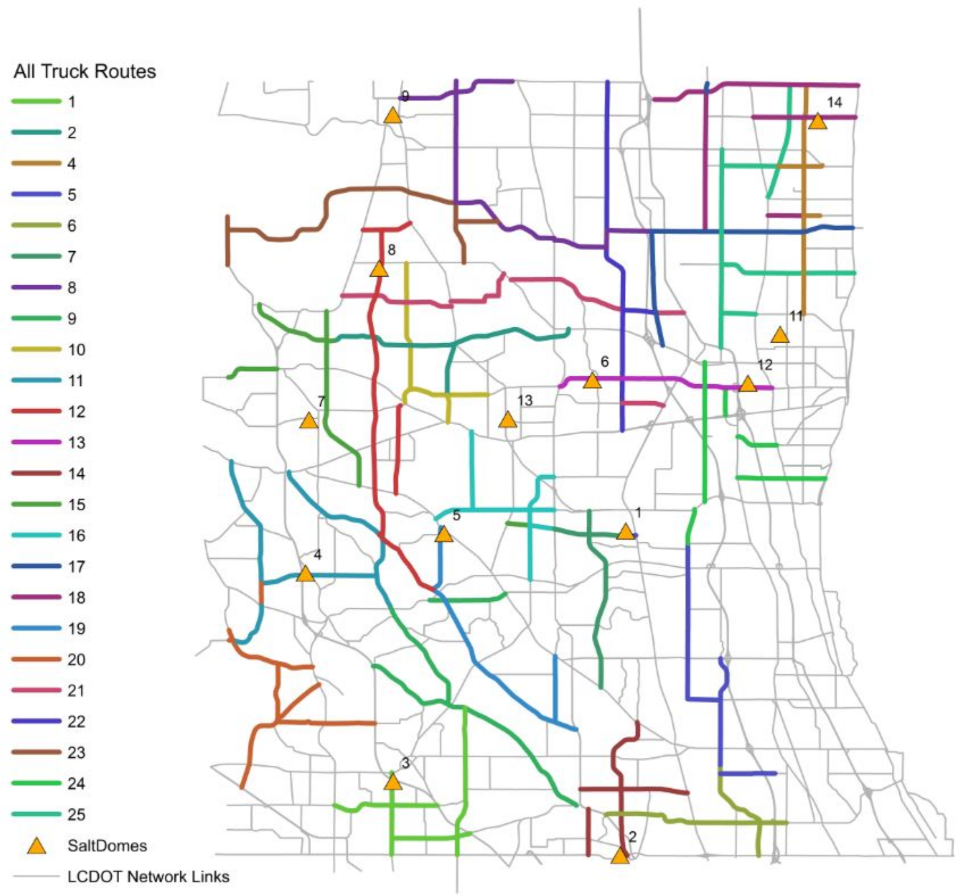


Figure 4.8: Optimal snow plow routes from Lake County, Illinois.

Table 4.1: Parameters for stochastic experiments.

parameter characteristic	attribute value
number of routes	at most 25 (Lake County, IL snow routes)
number of trucks	equal to the number of routes
planning horizon length, Γ	8 periods
time window length	1 period (20 min)
number of tasks over simulation	approximately 1000 (Lake County, IL task links)
net task revenue ($\$/mile$)	10
deadhead cost ($\$/mile$)	1
repositioning cost ($\$/mile$)	1

The cost of repositioning is an important parameter. If the repositioning cost is too high, the problem can almost be decomposed by route since trucks would not be willing to move to another route. When the problem becomes more separable, we expect the CAVE algorithm performs better too. We define the repositioning cost as a linear function of shortest path distance from route i to route j at time t. Similarly, the deadheading cost is defined as a function of deadhead distance on the optimal snow plow routes.

The proposed solution algorithm is coded in C++ and run on a desktop computer with 2.67 GHz CPU and 3.00 GB memory. We call the CPLEX solver (Clarke, 2004) within our program to solve the network sub-problem (4.1b)-(4.1d), (4.3), and (4.5). Table 4.2 reports the trucks' movements decisions (i.e., repositioning or performing tasks) over the planning horizon within a few minutes of computation time. At each iteration of the algorithm our proposed model determines the optimal decisions for either performing tasks or repositioning. Due to space limit, only 5 out of total available routes in the last iteration of the algorithm are presented. It can be observed that the total number of repositions at each time period t are quite small, implying that the trucks mainly stick to their assigned routes to decrease the total cost (based on the current parameter setting). In case of excessive repositioning, a threshold can further be determined (e.g., 0.1% repositions) to avoid unnecessary re-routes before finishing the assigned tasks on route i at time t.

Table 4.2: Summary of the truck movements (repositioning and performing tasks) over time.

route	time period	available trucks	repositioned trucks	available tasks	performed tasks
0	0	1	1	0	0
	1	1	0	12	12
	2	1	0	6	6
	3	1	0	15	15
	4	1	0	4	4
	5	1	1	0	0
	6	0	0	0	0
	7	0	0	0	0
1	0	1	0	4	4
	1	1	0	14	14
	2	1	0	17	17
	3	1	0	16	16
	4	1	0	13	13
	5	1	0	12	12
	6	1	0	9	9
	7	1	1	0	0
4	0	1	1	0	0
	1	1	0	15	15
	2	1	0	17	17
	3	1	0	12	12
	4	1	0	13	13
	5	1	0	13	13
	6	1	1	0	0
	7	0	0	0	0
8	0	1	0	6	6
	1	1	0	15	15
	2	1	0	17	17
	3	1	0	8	8
	4	1	0	12	12
	5	1	0	2	2
	6	1	1	0	0
	7	0	0	0	0
11	0	1	0	9	9
	1	1	0	7	7
	2	1	1	1	0
	3	1	0	7	7
	4	1	0	10	10
	5	1	0	6	6
	6	1	1	0	0
	7	0	0	0	0

Figure 4.9 (a)-(c) respectively depicts the total benefit gained from the decisions at each time period, total tasks performed, and total repositions over the planning horizon. To evaluate the performance of the CAVE approximation in the presence of uncertainty, we compare our solutions with those of a benchmark greedy strategy (i.e., a rolling-horizon heuristic algorithm, which solves the problem at time t using what is known at time t).

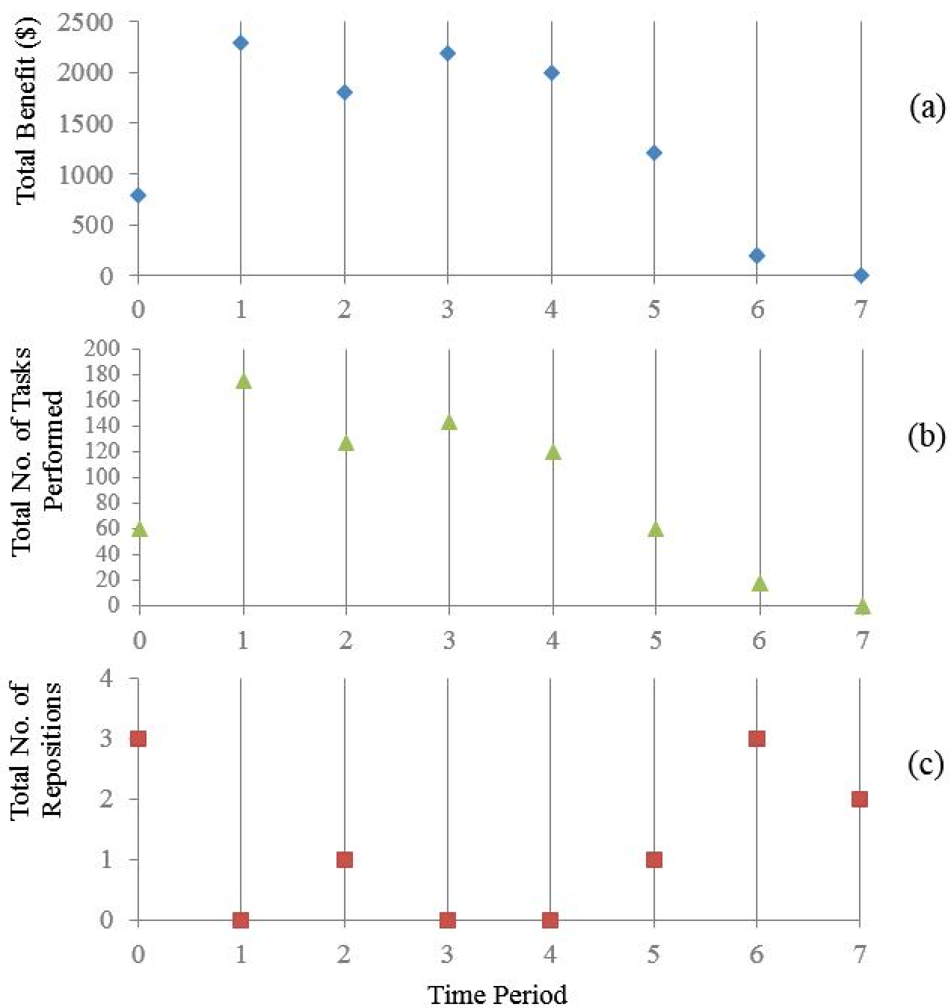


Figure 4.9: (a) Total benefit, (b) total number of repositions, and (c) total number of tasks performed at each time period t.

Furthermore, a set of sensitivity analyses is conducted on the value of the cost and revenue parameters in the model (See Figure 4.10 (a)-(c)). Each graph represents variations of the objective value (4.1a) based on the changes in (a) net revenue (per mile), (b) deadhead cost per mile, and (c) reposition cost per mile. Each graph is generated by only increasing one parameter value and fixing the other two parameters by the values presented in Table 4.1. Figure 4.10 (a) shows that the objective value (i.e., total benefit) increases when net revenue parameter grows from 1 to 1000 $\$/mile$. It is also observable from Figure 4.10 (b) that the objective value decreases by increasing the deadhead cost parameter value from 1 to 1000 $\$/mile$. Besides, Figure 4.10 (c) depicts a decreasing trend for the objective value with respect to the increase in the value of the repositioning cost parameter (similarly from 1 to 1000 $\$/mile$).

We can observe that the objective value is more sensitive to the deadheading cost parameter than repositioning. This trend occurs because the network forces a lot of deadhead links in between the maintenance task links, [2] while the number of repositionings in the network are often less as trucks often tend to take care of their assigned tasks unless unexpected conditions happen. In addition, the amount of decrease varies for different values of the deadheading as well as repositioning cost per mile, i.e., the slope of graphs (b) and (c) change while maintaining the decreasing trend. This is due to the stochasticities that we have in our problem (e.g., demand pattern) that affects our decisions (i.e., moving to the assigned routes (including the deadheads) or repositioning) at each location i and time t.

[2] Deadhead links are the links that are not under the jurisdiction of the winter maintenance agency (here, LCDOT) and cannot be salted/plowed but can be passed in the truck routing procedure.

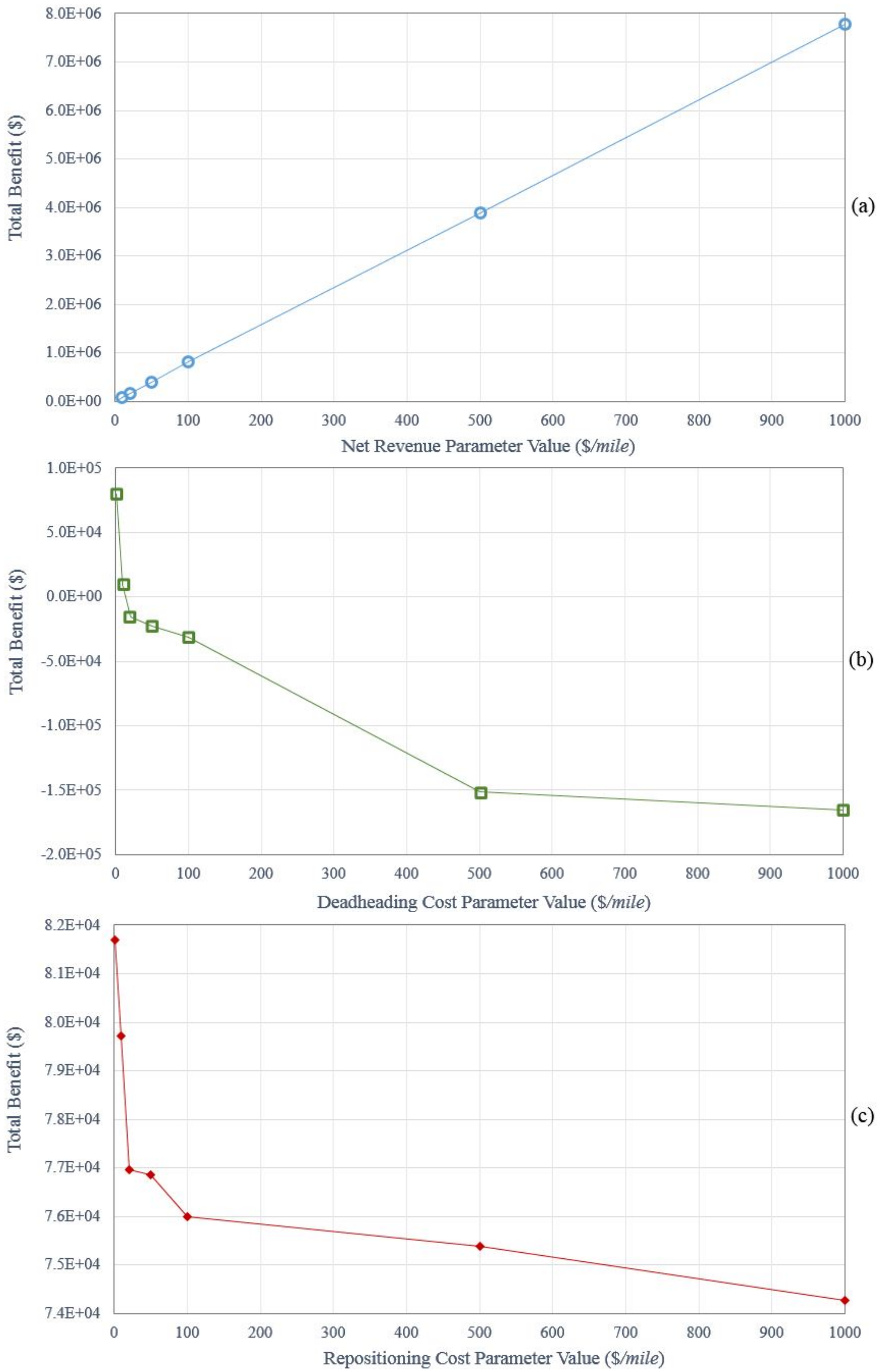


Figure 4.10: Total benefit with respect to revenue and cost parameter values: (a) net revenue ($\$/mile$), (b) deadheading cost ($\$/mile$), and (c) repositioning cost ($\$/mile$).

A comparison of total benefit is illustrated in Figure 4.11. This experiment shows that the proposed method outperforms the greedy algorithm by an average of 5.8 % over the planning horizon.

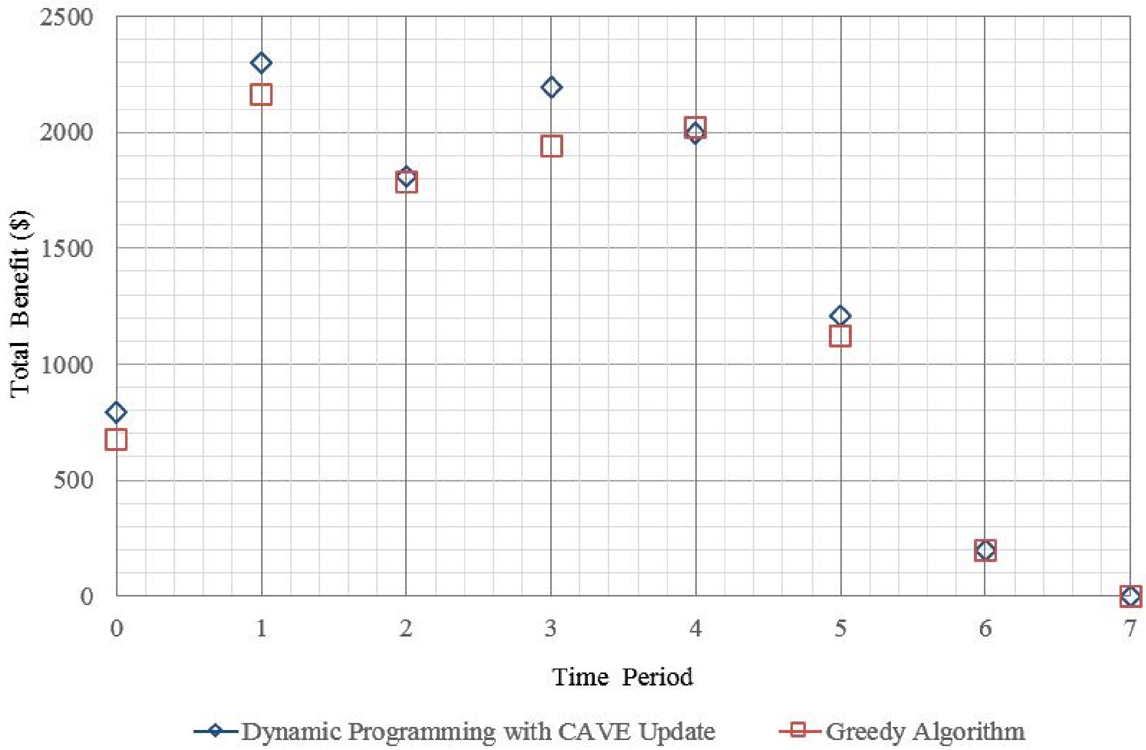


Figure 4.11: Comparison to an alternative algorithm at each time period t.

The advantage of the proposed solution approach lies primarily in the superior repositioning decisions made under resampling. Recall that the purpose of repositioning is basically to move trucks to routes with better future opportunities. However, if the future develops differently than expected, such movements incur penalties in two-fold: (i) the wasted repositioning cost and (ii) the opportunity cost of the wrong decision. Repositioning becomes more like a guessing game as the number of routes increases. Since resampling provides a richer view of the future than a single expectation, guessing under resampling is more likely to lead to better understanding of future opportunities. Furthermore, the gap between re-sampling and using a single expectation increases as the fleet size decreases, which is due to the more significant consequences of unnecessary repositioning.

4.5 Conclusion

This chapter presents a stochastic dynamic fleet management model for snow plow trucks during winter storms, where the snow maintenance demand is uncertain and service disruption may occur. The problem is formulated into a dynamic programming model, and an approximate dynamic programming solution procedure using a CAVE approximation algorithm for estimating the value function is introduced. The proposed solution technique is applied to a real-world case study in Lake County, Illinois. Besides, a set of sensitivity analyses has been conducted to capture the impact of the cost/revenue parameter values on the optimal decisions at each stage and total benefit over the planning horizon. Computational results show that the proposed algorithm is able to solve the problem effectively and outperforms a rolling-horizon heuristic solution. Alternatively, to compare the performance of the proposed methodology, a genetic algorithm (GA) can also be developed. However, the decision space in the GA framework will be very large in this case as our decision variables include time and location indices. Therefore, the chromosome length (i.e., the length of the string of our decision variables) will be very long and thus, we need to generate a large population for that chromosome length. This makes the algorithm inefficient; therefore, it is skipped in this dissertation.

Future research can be conducted in a few directions. A possible extension to our work is to develop more benchmark solution strategies and compare the numerical results following each technique. It is also interesting to consider different approximation methods for value function approximations. Furthermore, incorporating live traffic information as well as on-line location data from the service trucks can improve model realism and lead to more efficient snow plow fleet management during adverse winter conditions.

Acknowledgment

This research was financially supported in part by NSF CMMI 0748067. I would like to thank Martin Buehler, Al Giertych, Paula Trigg, Michael Scott, and Edwin Montano (Lake County Division of Transportation) for data support.

Chapter 5

Strategic Planning: Satellite Salt Replenishment Location Optimization

In this chapter we propose models and solution algorithms to determine the optimal number and location of satellite salt replenishment facilities prior to snow plow route design (see Figure 5.1). We simultaneously consider the facility location design, required distribution networks, and transportation plans as well as routing cost for snow plow trucks per network design. A version of this work has been done for the biofuel supply chain design and multi-modal transportation infrastructure expansion.[1] Although it is applied to another case study, mathematical models and solution algorithms can partially hold for this chapter.

[1] Hajibabai, L. and Ouyang, Y. (2013). Integrated Planning of Supply Chain Networks and Multi-modal Transportation Infrastructure Expansion: Model Development and Application to the Biofuel Industry. Computer-aided Civil and Infrastructure Engineering, 28(4): 247-259.

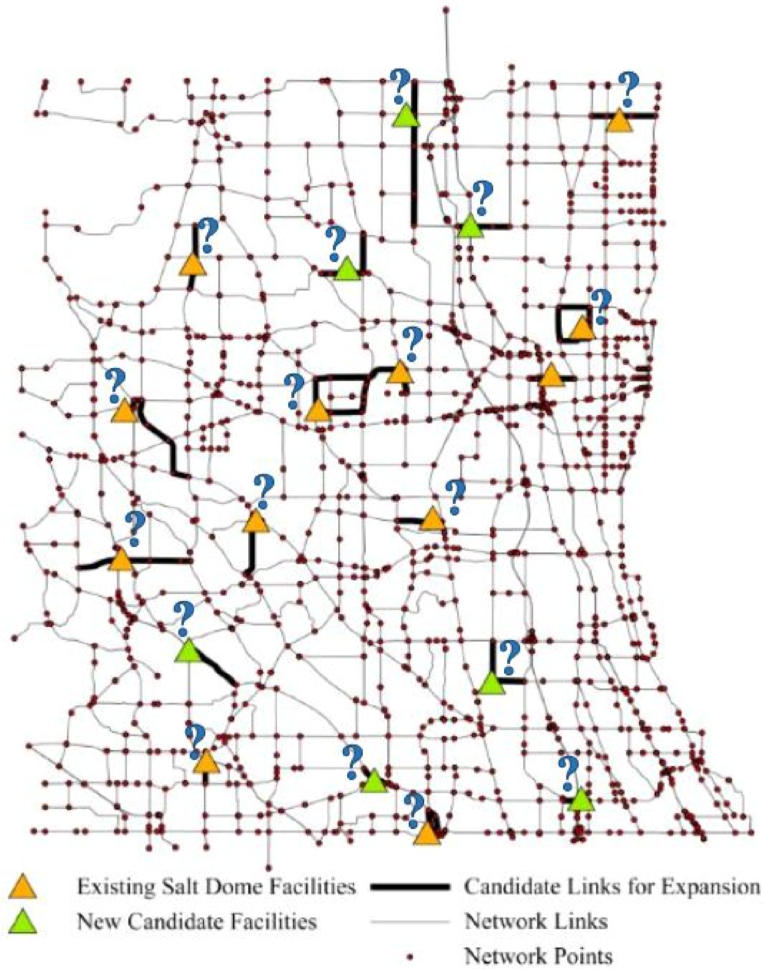


Figure 5.1: Strategic decisions: salt dome facility location design and roadway capacity expansion.

5.1 Introduction

This chapter proposes an integrated mathematical model for strategic planning of network design that encompasses salt dome number and location, transportation (both general roadway users and snow plow trucks' routing), and possible infrastructure capacity expansion. Figure 5.2 illustrates the relationship among these components. Our objective is to minimize the total cost including the transportation costs (for snow plow trucks and traveling public), the infrastructure investments (in both new salt dome construction and network capacity expansion), and routing cost. The snow plow trucks and existing users' transportation part of the problem can be modeled as a traffic assignment problem, the salt dome location part of the problem generally can be modeled as a fixed-charge facility location problem, and the snow plow trucks' routing is approximately modeled (instead of solving vehicle routing problem within the integrated framework) to determine the routing cost under optimal network design. The integration of facility location, public/truck transportation, truck routing, and infrastructure expansion makes the problem very difficult to solve. We propose a genetic algorithm to handle the facility location and infrastructure capacity expansion part of the problem, while using embedded convex combination algorithm to solve the traffic assignment decisions and continuous approximation algorithm to take care of the optimal routing cost.

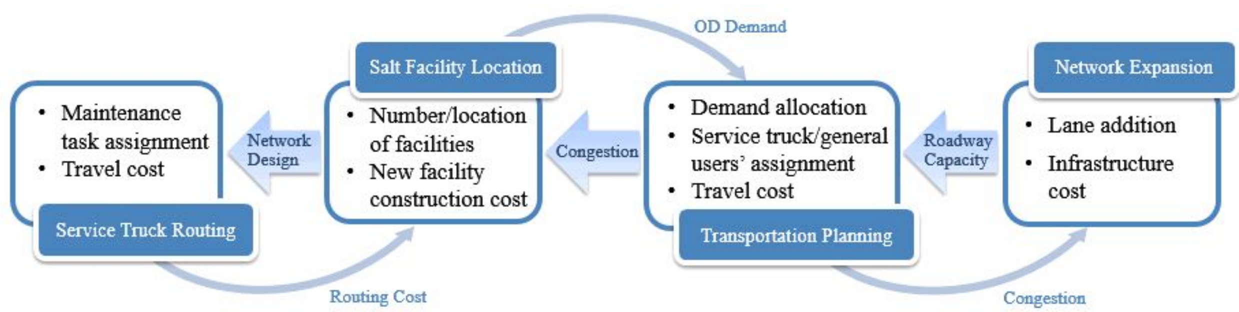


Figure 5.2: Interactions among transportation planning, facility location, and infrastructure capacity expansion.

It shall be noted that although this chapter focuses specifically on salt replenishment facility location problem due to its importance, the modeling framework can be applied to a wide range of application contexts (e.g., urban land use development, biofuel supply chain design). Similar problems would arise as long as the origins, destinations, and paths of multiple types of commodity flow are determined simultaneously, while the capacity of network links could be expanded to mitigate congestion.

The exposition of this chapter is as follows. Section 5.2 focuses on the mathematical formulation and the notations used in the model. Section 5.3 introduces the solution approach that has been used to solve the problem. In the future, Section 5.5 will present the empirical case study for the Lake County network, Illinois.

5.2 Model Formulation

This section presents a mixed integer non-linear program (MINLP) that simultaneously addresses salt dome location, snow plow truck routing, and infrastructure expansion decisions under traffic congestion. The objective of our model is to minimize the total costs for the entire planning and operations including the investments in new salt dome construction and network capacity expansion, and routing and transportation cost (for snow plow trucks and public travel delay).

For transportation cost under congestion, we consider service truck movements from salt dome locations to demand areas. We let I^d represent the set of service truck demand locations. Let J denote the set of candidate locations for salt domes including the existing facilities as well as new candidates. Trucks dispatch from salt dome $j \in J$ with supply h_i^s to region $i \in I^d$ that has maintenance demand h_i^d. The unit of h_i^d is hourly passenger car equivalents (PCE), which are estimated based on full-truck load (salt capacity) in volume and hourly passenger car flow equivalent for trucks (HCM, 2000).

The construction of a salt dome at location $j \in J$ involves a fixed cost of m_j and a resource (e.g., salt, chemicals, etc.) capacity of C_j. In this study, we simply assume a fixed salt dome capacity for each candidate location. Sometimes, the refinery capacity at one given location could be selected from a discrete set. In such cases, the location part of the model should follow the multi-type facility location formulation (Akyüz et al., 2012; Klose and Drexl, 2005; Jen et al., 1968) by introducing additional decision variables that indicate the salt dome capacity (i.e., facility type) at each candidate location. The selection of locations for salt domes is determined by decision variables Y_j as follows:

$$Y_j = \begin{cases} 1 & \text{if a salt dome is built at } j \in J, \\ 0 & \text{otherwise.} \end{cases}$$

We assume each salt dome facility j dispatches trucks to visit maintenance task links at a fixed frequency; we use $\mathcal{X}$ to denote the number of visits of a task link in a year. Let γ_j be the total number of tasks served in a specific salt dome j neighborhood, λ be the number of tasks each truck can serve (i.e., truck capacity), $\mathcal{D}_{i'j}$ be the distance between task link $i' \in A$ and salt dome j. Shen and Qi (2007) show that the optimal VRP distance $\mathcal{V}_j$ can be approximated by the following formulation:

$$\mathcal{V}_j \approx 2 \left(\sum_{i' \in A} \frac{\mu_{i'}}{\mathcal{X}} \, \mathcal{D}_{i'j} \right) / \, \lambda \, + \, (1 - 1/\lambda) \, \Phi \, \gamma_j \, \left(\frac{\mathcal{A}'}{\mathcal{N}} \right)^{0.5} , \tag{5.1}$$

where $\mu_{i'}$ is the yearly maintenance demand at task link i', given total tasks $\mathcal{N}$ uniformly scattered in area $\mathcal{A}'$. Φ is a constant value that is assumed 0.75 for Euclidean metrics (Shen and Qi, 2007). Figure 5.3 illustrates the clusters for each salt dome neighborhood (truck routing is only shown in one of the clusters). Trucks will be dispatched from the main depot to each salt dome neighborhood and tasks will be performed within each salt dome neighborhood following a routing procedure. Trucks will then go back to the main depot. We estimate these truck travel costs using (5.1), which includes, in the first term, the weighted average maintenance demand in each cluster according to the distance of each task link to its assigned salt dome facility. The second term estimates the traveling salesman cost in each cluster, which is the major cost component (higher weight, as λ is large) in this estimation.

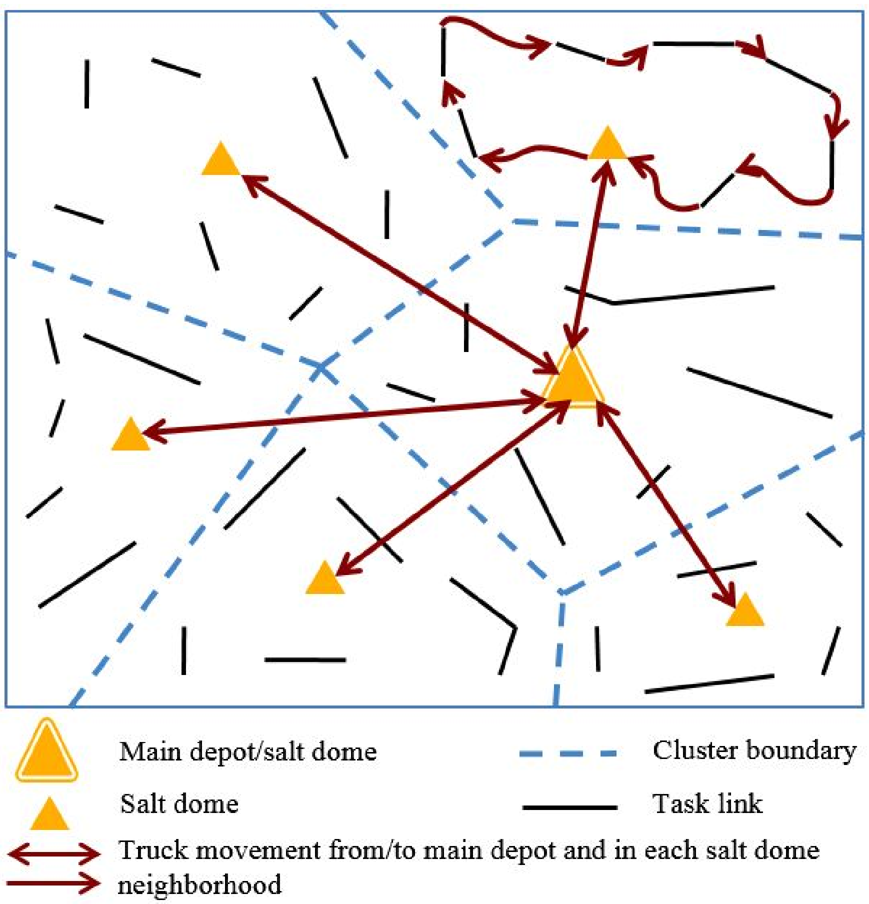


Figure 5.3: Routing cost approximation.

Suppose that trucks move through a transportation network containing a set of roadway links A. An imaginary node, S^d, is added as a source node for the demand transportation, which is connected by a set V of virtual links to candidate location $j \in J$ if there is an open facility (i.e., $Y_j = 1$). This can be interpreted as considering node S^d as the only origin of truck transportation to the set of demand points I^d as destinations. This ensures that all the flows from the source node will pass through at least one open salt dome. Figure 5.4 schematically illustrates the problem, which includes salt dome location, maintenance demand points, the transportation network, and origin/destination of the transportation flows (i.e., background traffic a well as service truck routing).

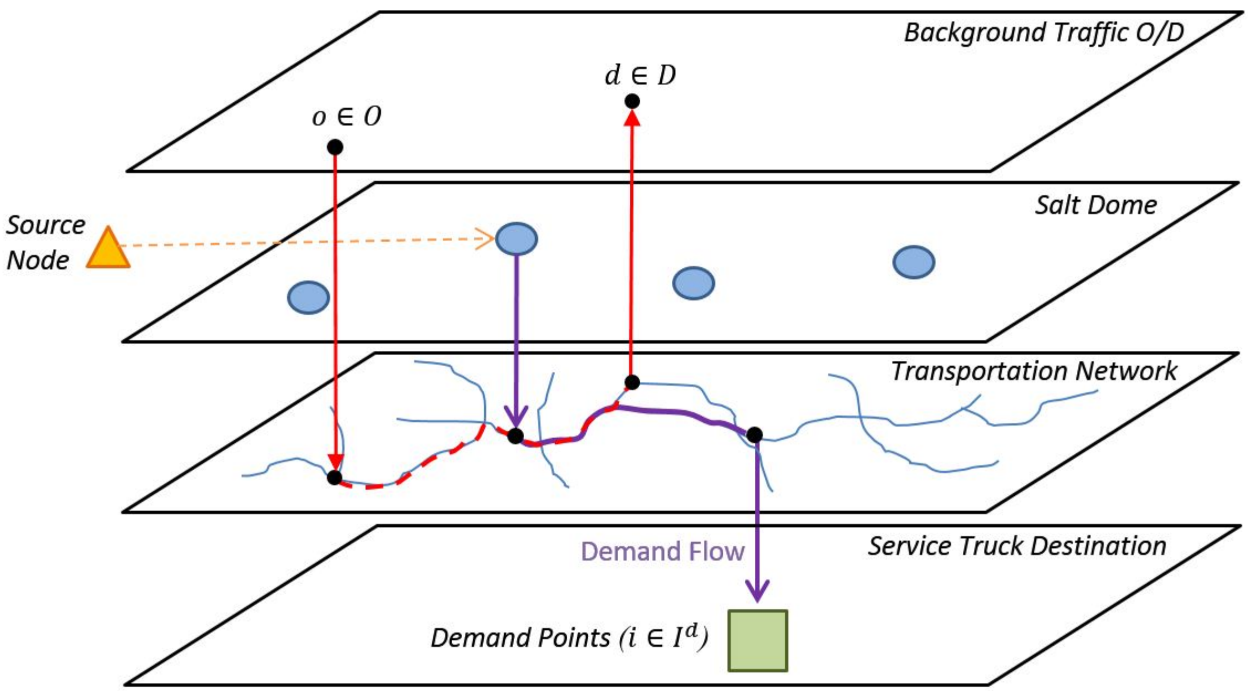


Figure 5.4: Salt dome candidate location, demand points, and transportation network.

Let $K^{d,i}$ denote the set of roadway paths from the source node S^d to a demand point $i \in I^d$. Demand flow $f_k^{d,i}$ travels from S^d to $i \in I^d$ on a possible path $k \in K^{d,i}$. Link-arc incidence parameter $\delta_{a,k}^{d,i}$ (Sheffi, 1985) is further introduced as follows:

$$\delta_{a,k}^{d,i} = \begin{cases} 1 & \text{if path } k \in K^{d,i} \text{ includes link } a \in A, \\ 0 & \text{otherwise.} \end{cases}$$

Similarly, parameter $\Delta_{j,k}^{d,i}$ is defined for the virtual links connected to roadway network:

$$\Delta_{j,k}^{d,i} = \begin{cases} 1 & \text{if path } k \in K^{d,i} \text{ includes node } j \in J, \\ 0 & \text{otherwise.} \end{cases}$$

Link flows on these virtual link v_j^d represent salt dome throughput at location $j \in J$. It can be expressed as $v_j^d = \sum_{i \in I^d} \sum_{k \in K^{d,i}} f_k^{d,i} \Delta_{j,k}^{d,i}$. In addition to the roadway maintenance demand flows, we assume that there are passenger traffic flows on the roadway transportation network from a set of origins, O to a set of destinations, D. Let $K^{o,d}$ denote the set of paths connecting a passenger flow origin $o \in O$ and destination $d \in D$ through the roadway network, and let $f_k^{o,d}$ represent the flow on path $k \in K^{o,d}$. On each roadway link $a \in A$, the total passenger flow is $\sum_{o \in O} \sum_{d \in D} \sum_{k \in K^{o,d}} f_k^{o,d} \delta_{a,k}^{o,d}$, where $\delta_{a,k}^{o,d} = 1$ if path $k \in K^{o,d}$ contains link $a \in A$, and 0 otherwise.

In summary, the total link flow x_a for all $a \in A$ is the summation of the background (passenger) flow and the demand transportation flow, that is,

$$x_a = \sum_{o \in O} \sum_{d \in D} \sum_{k \in K^{o,d}} f_k^{o,d} \delta_{a,k}^{o,d} + \sum_{i \in I^d} \sum_{k \in K^{d,i}} f_k^{d,i} \delta_{a,k}^{d,i}, \ \forall a \in A. \tag{5.2}$$

We consider the option of expanding roadway link capacity (especially in local areas close to salt dome facilities); let decision variable $Z_a \in \{0, 1, 2, \cdots\}$, $\forall a \in A$ be the number of lanes added to link a, and each additional lane yields a known extra capacity of q_a. The capacity of link a after the capacity expansion is the summation of the original link capacity Q_a and the additional capacity; that is, $Q_a + Z_a\, q_a$. The travel time on link $a \in A$, denoted by $t_a(x_a, Z_a)$, is assumed to take the following BPR function form based on the traffic volume and expanded link capacity $t_a(x_a, Z_a) = t_0 \left(1 + \alpha(\frac{x_a}{Q_a + Z_a\, q_a})^\beta\right)$, $\forall a \in A$, where constant parameters $\alpha = 0.15$ and $\beta = 4$ (BPR, 1970). The cost for roadway link capacity expansion, $c_a(Z_a)$, $\forall a \in A$, can be expressed as the product of link length l_a, the additional capacity $q_a\, Z_a$, and a cost coefficient w (Unnikrishnan et al., 2009), that is $c_a(Z_a) = w\, l_a\, q_a\, Z_a$, $\forall a \in A$.

The mathematical optimization model that integrates location, routing, and network design, and transportation decisions can be expressed as (5.3a)-(5.3h). The objective function (5.3a) minimizes the total system cost, which includes facility construction investment, infrastructure capacity expansion cost, costs for transportation, and public travel cost, respectively. Parameter ρ converts link travel time to travel cost and reflects a relative weight of total travel cost against the construction costs. This system optimal objective may be more suitable for centrally controlled trucks than for the public traffic (e.g., Aziz and Ukkusuri (2012)). Constraints (5.1) estimate the routing cost in each salt dome j neighborhood. Constraints (5.2) indicate that the traffic flow on each roadway network link is the sum of the background traffic and the passenger car equivalent demand truck flows on roadway network. Constraints (5.3b) ensure that the flow on each virtual link is the sum of the demand flow from all paths which contain node $j \in J$. Constraints (5.3c) show that the sum of all demand flows into a demand point should be equal to the demand at that point. Constraints (5.3d) ensure that the flow v_j^d can be any non-negative value no greater than the capacity of the salt dome at candidate location $j \in J$ (if there is a salt dome at that node). Constraints (5.3e) ensure that the total capacity of salt domes should exceed the total demand. Finally, constraints (5.3f)-(5.3h) define the binary and non-negative variables.

$$\text{minimize} \quad \sum_{j \in J} (m_j Y_j + \rho' \mathcal{V}_j) + \sum_{a \in A} (c_a(Z_a) + \rho \, x_a \, t_a(x_a, Z_a)) \tag{5.3a}$$

$$\text{subject to} \quad (5.1), (5.2), \text{ and}$$

$$v_j^d = \sum_{i \in I^d} \sum_{k \in K^{d,i}} f_k^{d,i} \, \Delta_{j,k}^{d,i} \, , \forall j \in J, \tag{5.3b}$$

$$h_i^d = \sum_{k \in K^{d,i}} f_k^{d,i}, \, \forall i \in I^d, \tag{5.3c}$$

$$v_j^d \leq C_j \, Y_j, \, \forall j \in J, \tag{5.3d}$$

$$\sum_{i \in I^d} h_i^d \leq \sum_{j \in J} C_j \, Y_j, \, \forall i \in I^d, \, j \in J, \tag{5.3e}$$

$$Y_j \in \{0, 1\}, \, Z_a \geq 0, \text{integer}, \, \forall j \in J, \, a \in A, \tag{5.3f}$$

$$f_k^{d,i} \geq 0, \forall i \in I^d, \, k \in K^{d,i}, \tag{5.3g}$$

$$f_k^{o,d} \geq 0, \forall o \in O, \, d \in D, \, k \in K^{o,d}. \tag{5.3h}$$

5.3 Solution Approach

The integrated mathematical model (5.1)-(5.2) and (5.3a)-(5.3h) involves non-linearity and mixed integer variables, and hence it is very difficult to solve this model to exact optimality. To overcome this challenge, we propose a hybrid solution approach that integrates the genetic algorithm (Goldberg, 1989), continuous approximation (Daganzo, 2005; Shen and Qi, 2007), and traffic assignment algorithm (Frank and Wolfe, 1956; Sheffi, 1985).

5.3.1 The GA Framework

The GA has been used to effectively solve transportation network design problems (e.g., Ukkusuri et al. (2007); Putha et al. (2012)). It begins with a population of individuals that develops through generations based on the principle of "survival of the fittest." Interested readers are referred to Goldberg (1989) for more details on GA.

The general framework for our GA approach is depicted in Figure 5.5. First, the basic parameter setting such as the size of population (n) and the probabilities of crossover and mutation must be initialized. We define a chromosome to be a binary vector representation of the integer variables $\{Y_j\}$ and $\{Z_a\}$, such that the length of the chromosome depends on $|J|$, $|A|$, and the maximum number of lanes we may add to a roadway link. If we only allow at most one lane addition to each link (i.e., $Z_a \in \{0, 1\}$, $\forall a \in A$), then the length of the chromosome is simply $|J| + |A|$.

In the initialization step, chromosomes are randomly generated for the first population, and each chromosome should satisfy constraints (5.3e) to ensure feasibility of the solution; that is, the total maintenance demand should not exceed the total capacity of the salt dome facilities. Each chromosome in the population contains information on the facility location and road expansion decisions. Based on this information we perform traffic assignment to determine near-optimal link traffic volumes on roadway network $\{x_a\}$. We also determine the optimal clusters to assign task links to salt replenishment facilities and estimate the truck routing cost $\mathcal{V}_j$. The detail of this step is explained in the following section. The fitness function that we use to evaluate and rank each chromosome is the inverse of the objective value.

GA creates chromosomes for new populations through a series of operations including selection, crossover, and mutation. The tournament selection technique is used to choose the chromosomes for later perturbations in crossover and mutation operations. The crossover uses a multi-point technique where cross points are randomly selected for each part of the chromosome (e.g., those corresponding to location decisions $\{Y_j\}$ and capacity expansion decisions $\{Z_a\}$). Then, a bit-wise mutation is used, that is, each cell of the chromosome (i.e., gene) is randomly flipped according to the probability of mutation. All parts of the chromosome are mutated in the same way but the cells representing the location decision are never switched with the cells representing the capacity expansion decision. Again, any newly generated chromosome should satisfy constraints (5.3e) or otherwise it will be discarded. Finally, the algorithm terminates either when the predetermined maximum number of generations is reached, or if the best solution has not been improved over a certain number of consecutive generations. The best chromosome over all generations is recorded as the solution to the problem.

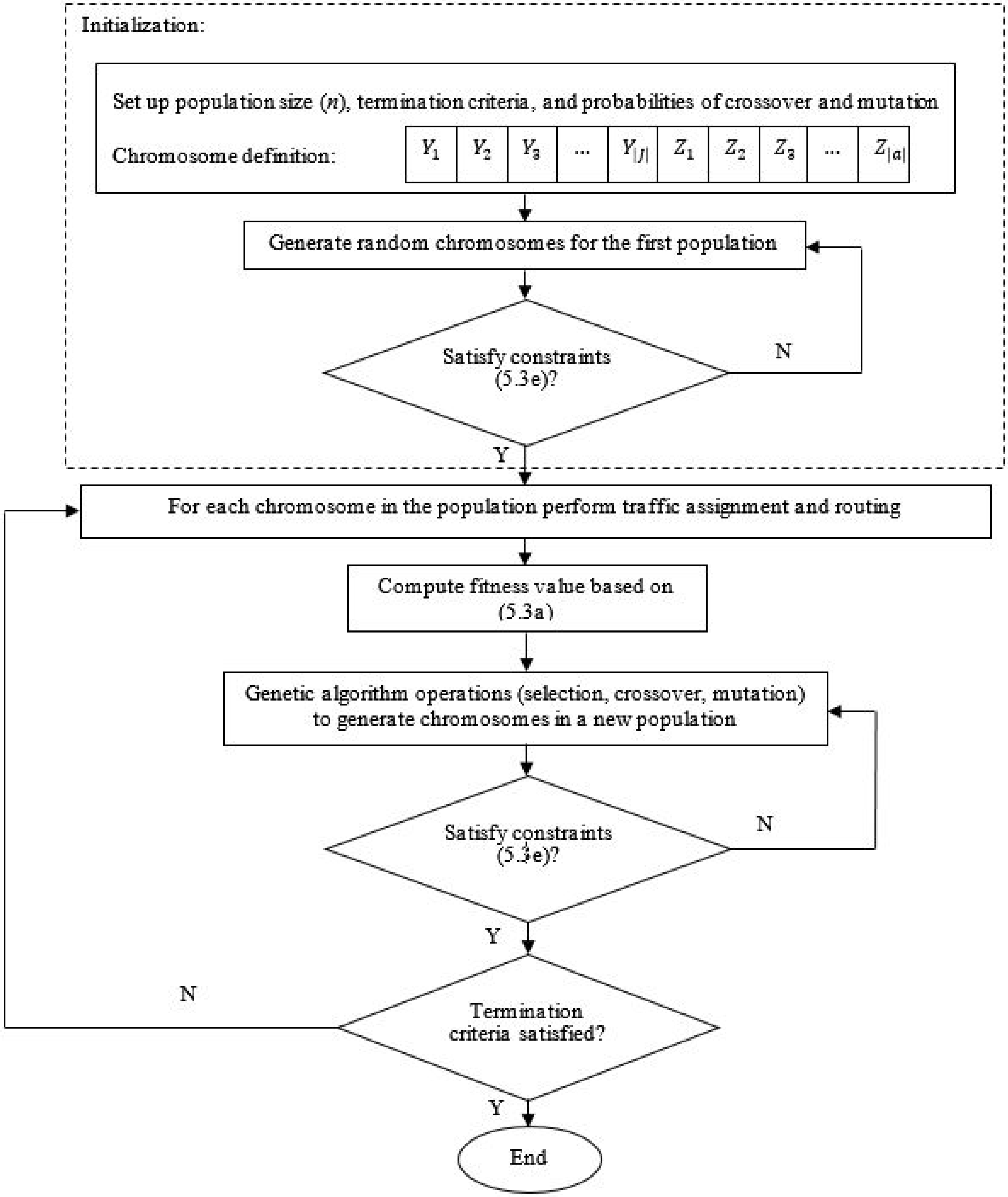


Figure 5.5: General Framework of the GA approach.

5.3.2 Routing Decisions for a Given Chromosome

As discussed above, the facility location and roadway capacity expansion decisions are determined by the chromosomes in the GA framework. We cluster task links $i' \in A$ into j optimal salt replenishment facility locations and estimate the routing cost for the current network design according to (5.1). We also use a convex combination algorithm to solve the remaining routing decisions (trucks and general roadway users traffic volume). For any given chromosome, the sets of decision variables $\{Y_j\}$ and $\{Z_a\}$ are known, therefore model (5.2) and (5.3a)-(5.3h) reduces to a simpler non-linear program. Then, the convex combination method can be applied to solve the routing problem.

5.4 Alternative Formulation with User-Equilibrium for Background Traffic

In this chapter, the congestion pattern and total transportation costs have been computed based on system optimal flows. In reality, user equilibrium flows could also be incorporated into the formulation to address alternative route choices of the background traffic b_a. To this end, formulation (5.1)-(5.2) and (5.3a)-(5.3h) can be rewritten into a mathematical program with equilibrium constraints (MPEC), as follows.

$$\text{minimize} \quad \sum_{j \in J} (m_j Y_j \, + \, \rho' \, \mathcal{V}_j) \, + \sum_{a \in A} (c_a(Z_a) \, + \rho \, x_a \, t_a(x_a, Z_a)) \tag{5.4a}$$

subject to (5.1) – (5.2), (5.3b) – (5.3g), and

$$x'_a = \sum_{i \in I^d} \sum_{k \in K^{d,i}} f_k^{d,i} \delta_{a,k}^{d,i}, \; \forall a \in A, \tag{5.4b}$$

and

$$b_a \in \text{argmin}_{b_a} \sum_{a \in A} \int_{x'_a}^{x'_a + b_a} t_a(\omega, Z_a) d\omega \tag{5.4c}$$

subject to (5.3h) and

$$b_a = \sum_{o \in O} \sum_{d \in D} \sum_{k \in K^{o,d}} f_k^{o,d} \delta_{a,k}^{o,d}, \; \forall a \in A, \tag{5.4d}$$

$$\sum_{k \in K^{o,d}} f_k^{o,d} = q^{od}, \; \forall o \in O, \; d \in D. \tag{5.4e}$$

The above MPEC model is generally difficult to solve. However, the literature has repeatedly shown that such MPEC models can often be effectively tackled by the GA approach (e.g., Unnikrishnan and Lin (2012); Lin (2011)). In such a framework, once the $\{Y_j\}$ and $\{Z_a\}$ variables are fixed, the remaining problem becomes a multi-class mixed traffic assignment problem, which can be solved effectively. Interested readers are referred to (Nie and Zhang, 2008) for more detailed information.

5.5 Case Study

The proposed solution algorithm in Section 3.3 is implemented and applied on a real-world case study for the Lake County, Illinois. The roadway network database of the Lake County Division of Transportation (LCDOT) contains 1,354 nodes and 3,350 links (See Figure 3.5). Each task link in the network is considered as a demand location for plowing and salting service. A dataset including 162 links for possible capacity expansion in the neighborhood of salt dome facilities is chosen. There are 19 candidate locations for salt replenishment facility construction, including 12 existing facility locations and 7 new locations that are selected based on socioeconomic factors such as access to major transportation facilities. If one of these candidates are selected for facility construction, a set of roadway segments will also be built to connect the facility to the existing roadway network. The demand for plowing and salting service is calculated based on the historical service demand (passenger cars per hour (pc/hr)).

Furthermore, the maximum capacity of salt dome facilities is considered to be 4,000 $tons/year$, except for the main depot that is 10,000 $tons/year$. The annual prorated cost of constructing new salt dome facilities of size 4,000 $tons/year$ and roadway link connectors to the existing network links will be $\$\,1,297,000$ (i.e., $\$\,10^6$ to build a facility of that size and $\$\,297,000$ to expand the roadway network by adding links for connecting the new facility to the existing roadway; the roadway capacity expansion cost is calculated according to Unnikrishnan et al. (2009) for $0.3\,miles$ roadway length and the cost coefficient factor of $10^3\,\$/lane - mile$). The salt dome facility construction cost is further prorated into hourly cost by assuming 20 years of service life and 60 days per year and 8 hours per day as the effective working time for the snow plowing/salting work shift.

Traffic flow is converted to passenger car equivalent (PCE) per hour. Information on the hourly passenger car traffic flow and the existing capacity of the roadway links are obtained from Illinois Department of Transportation (IDOT, 2011). The capacities of the interstate highways and local arterials are assumed to be 2,200 and 1,045 *pcphpl*, respectively (HCM, 2000). We set $\rho = 20\ \$/hr{-}PCE$, $\rho' = 10^3/plowspeed\ \$/hr{-}PCE$ (where 30 is assumed for the average speed of snow plow trucks' operation), and the cost coefficient factor for capacity expansion is $10^3\ \$/lane - mile$ (Unnikrishnan et al., 2009). Furthermore, we assume that at most one roadway lane will realistically be added to each link if road capacity expansion is required (i.e., $Z_a \in \{0, 1\}$).

Routing cost is estimated by (5.1), which uses parameter values as follows. We assume each task link is visited 100 times in a year (i.e., $\mathcal{X} = 100$). According to each GA solution for location decisions (i.e., Y_j), we cluster tasks into zones (i.e., assign each task to each salt dome facility j). The total number of tasks served in each zone (i.e., γ_j) is the number of task links in each cluster j to be served by a truck. Truck service capacity is determined based on LCDOT data; i.e., λ is assumed to be equal to total available tasks in the network that total trucks in LCDOT can serve per one operation cycle without salt replenishment along the way. We assume equal trucks with equal capacity of service. The yearly maintenance demand at task link i' (i.e., $\mu_{i'}$) is calculated by the product of number of visits per year, $\mathcal{X}$, length of each task link, and the salt application rate of trucks divided by average salt carrying capacity of trucks. Constant parameter Φ is assumed 0.75 (Shen and Qi, 2007). we derived the total routing area $\mathcal{A}'$ as well as total tasks, $\mathcal{N}$, from Lake County dataset.

5.5.1 Results and Discussion

This proposed algorithm is coded in Visual C++ and run on a desktop computer with 2.67 GHz CPU and 3.00 GB memory. In the GA framework, the selection pressure is chosen to be 20, the population size 300, probability of crossover 0.8, probability of mutation 0.01, chromosome length 181 (i.e., 19 candidate locations and 162 candidate links for expansion close to salt replenishment facilities), and random seed value 0.025. Figure 5.6 represents the structure of the GA population. Candidate facilities $1, 2, \ldots, 12$ are chosen from the existing facilities in LCDOT network and the rest are new candidate locations. The program terminates when the best fitness value does not improve across a number of generations. For all numerical cases, the GA converges within 100 generations, taking less than an hour of CPU time.

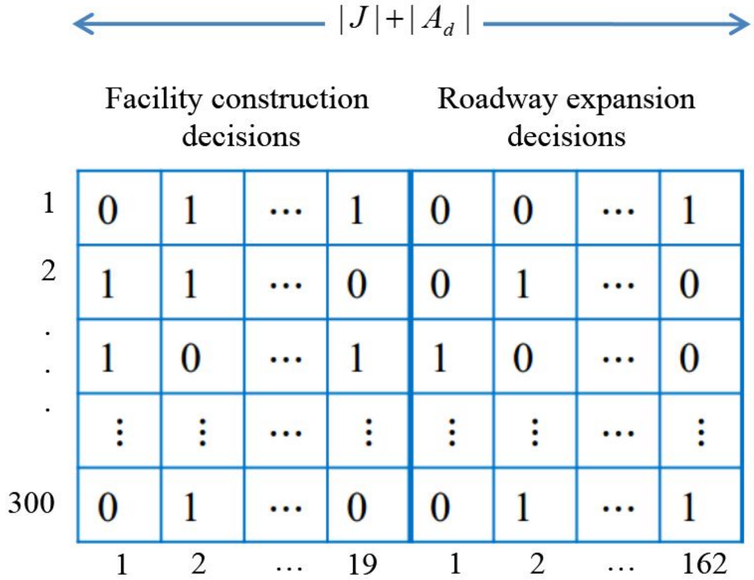


Figure 5.6: The structure of GA population.

It shall be noted that following our chromosome setting, constraints (5.3e) will be satisfied with a probability of 99.7% as follows. Constraints (5.3e) ensure that the total capacity of salt domes should exceed the total maintenance demand. This affects the salt dome facility location design. In this case study, enough number of facilities with average capacity of $4315.8\,tons$ should exist to satisfy the total demand ($\cong 14 \times 10^3\,tons$). The probability of selecting k' number of salt dome facilities by GA out of a total of n' candidate facility locations follows a binomial distribution as $\binom{n'}{k'} p^{k'} (1-p)^{n'-k'}$, where, p is the probability of selecting the candidate facilities. Therefore, the probability of selecting at least x' facilities (i.e., minimum number of required facilities to satisfy total maintenance demand: $14 \times 10^3/4315.8 = 3.2$) is represented as follows.

$$\sum_{k' \geq x'} \binom{n'}{k'} p^{k'} (1-p)^{n'-k'}. \tag{5.5}$$

As GA assigns 0's and 1's to the facility location decision variables with 0.5 probability, the probability of satisfying constraints (5.3e) is computed as $1 - \sum_{k' \in \{0,1,2,3\}} \binom{19}{k'} 0.5^{n'} = 99.7\%$. As such, these constraints are mostly satisfied and only in a few instances GA has to generate a new solution in the initialization step to satisfy the constraints. In the following generations, the constraints are hardly violated as GA pushes the solutions towards the regions of the feasibility area with lower costs that happen to have more facilities as shown in the numerical results.

Table 5.1 presents the computation results for $\rho = 20$ and $\omega = 10^3$. Figure 5.7 shows the locations selected for salt domes as well as the roadway links considered for expansion for this case. For comparison, we also compute the optimal solution of a benchmark scenario where the existing salt replenishment facilities are selected and the model excludes roadway link capacity expansion. The proposed solution selects 7 existing facilities and 4 new candidate locations for facility construction. In the benchmark solution, all facilities are the existing ones, thus the facility construction cost is equal to 0. The results show that our joint optimization model (i.e., proposed model including capacity expansion decisions in the model) reduces the transportation and routing costs, and in fact reduces the total system cost by $\$\,0.54 \times 10^6$. Figure 5.7 shows the facility location design and roadway capacity expansion decisions on the Lake County, Illinois map.

Table 5.1: Summary of computation results.

scenario	# of domes	loc. of domes	cost for domes ($\times 10^6$ \$)	# of added lanes	cost for exp. ($\times 10^6$ \$)	trans. cost ($\times 10^6$ \$)	rout. cost ($\times 10^6$ \$)	sys. cost ($\times 10^6$ \$)
proposed model	11	1,5,7,8 9,10,11,14 15,16,17	0.25	31	0.37	4.07	2.29	6.99
benchmark solution	12	1,2,3,4,5 6,7,8,9 10,11,12	0	–	–	4.87	2.66	7.53
difference	–	–	–	–	–	16.4 %	13.9 %	7.2 %

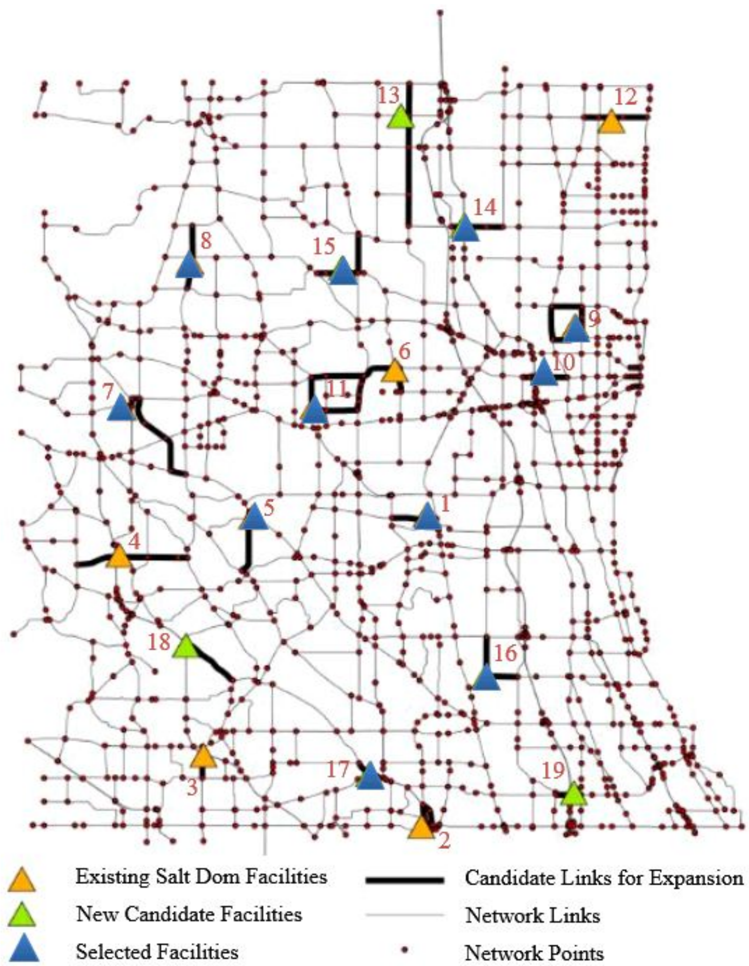


Figure 5.7: The algorithm results in the Lake County, Illinois network.

5.6 Conclusion

This study presents an integrated mathematical model for satellite salt replenishment facility design where the near-optimum number and location of facilities, the near-optimal routing of service trucks, and possible roadway capacity expansion are determined. The objective is to minimize the total cost for facility construction, roadway capacity expansion, transportation delay, and routing of service trucks. The congestion pattern and total transportation costs are determined based on traffic equilibrium flows on transportation routes under expanded capacity. To find the near-optimum solution to the proposed model, we develop a hybrid GA framework that incorporates convex combination and continuous approximation algorithms. A real-world case study for the Lake County, Illinois is conducted.

Future research can be generalized in a few directions. A possible extension is to develop a heuristic solution algorithm to evaluate the performance of our genetic algorithm framework with embedded traffic assignment technique. This research has assumed that the capital investment associated with roadway expansion is considered as part of the roadway maintenance operation investment plan. This is possible under potential public-private partnerships (Unnikrishnan et al., 2009). It may be interesting to expand our current framework to allow multiple stakeholders (e.g., public agency, winter maintenance agency, and public travelers) to have independent or conflicting objectives. This will probably require multi-level programs with equilibrium constraints. In addition, as we observe the impact of roadway capacity expansion on the total system cost, it would be interesting to study in the future how roadway expansion budget would affect the optimal solution. Another interesting extension is to develop efficient algorithms to solve location-routing problem within the general framework. It would also be interesting to apply our model and solution approach to a comprehensive list of diverse networks for additional real-world case studies.

Chapter 6

Decision-Support Software Design

A state-of-art snow route optimization software application is developed to help decision makers evaluate scenarios related to the roadway network, snow plow fleet, and to visualize the optimal results and the associated performance measures.

Snow control operations involve spatial information, and hence geographic information systems can provide a suitable platform for creating, maintaining and analyzing relevant data. Based on the mathematical model and solution algorithms described in Sections 3.2 - 3.3.2, we develop a snow route analysis and design software system, "Snow Route Optimizer," that is embedded in ESRI ArcGIS (see Appendix A for more details).

This software consists of two GIS user interfaces and a C++ optimization module. The user "Setting" interface in the GIS environment includes two sets of inputs, i.e., user-specified and fixed input parameters. These parameters are the inputs to the snow plow route analysis and optimization module.

Figure 6.1 shows the "Setting" user form interface for parameter and data settings, which provides various user options regarding the network, fleet data settings, and the optimization procedure. The first tab is designed for the "General" settings of the optimization application where the users can decide what scenario and optimization mode they prefer. In addition, the users can choose what time-of-day, cost metric, objective function weights, and maximum running time they favor for the optimized routes. Furthermore, the users can choose the long-storm scenario, in which each truck travels to the snow routes and makes multiple plow cycles before returning. To take this into account, the same path sequence is used for every cycle (for the same number of trucks and the same task links), while at the end of each cycle (except the last one) the trucks directly travel to the first task of the next cycle.

The "Network" tab allows users to specify the input data files for the points and links in the roadway network. This can be done by clicking the "Browse" button and choosing the "LCDOT_Data.mdb" file through the dialog box. The selected database file will be automatically converted into two text files ("LCDOTNetworkLinks.txt" and "LCDOTNetworkPoints.txt") suitable for the optimization program. In the next frame, the users can specify (*i*) whether the number of passes includes right turn lane, and (*ii*) the rounding threshold to compute the integer number of passes. In addition, salt application rate can also be entered by the users. The users can also choose whether they would like the trucks in the optimized route to share roadway links or whether they would like each truck to cover its own links.

Under the "Fleet" tab, the users can input the truck fleet size and composition. The users can make a list view to indicate the truck name (type), the number of each truck type in the fleet, each type of truck's salt capacity, and their fuel efficiency (i.e., miles per gallon). The users also has the option to remove any row from the list view by selecting this row and clicking the “Remove” button. On the other hand, the user will have the option to read the fleet specifications from the fleet database.

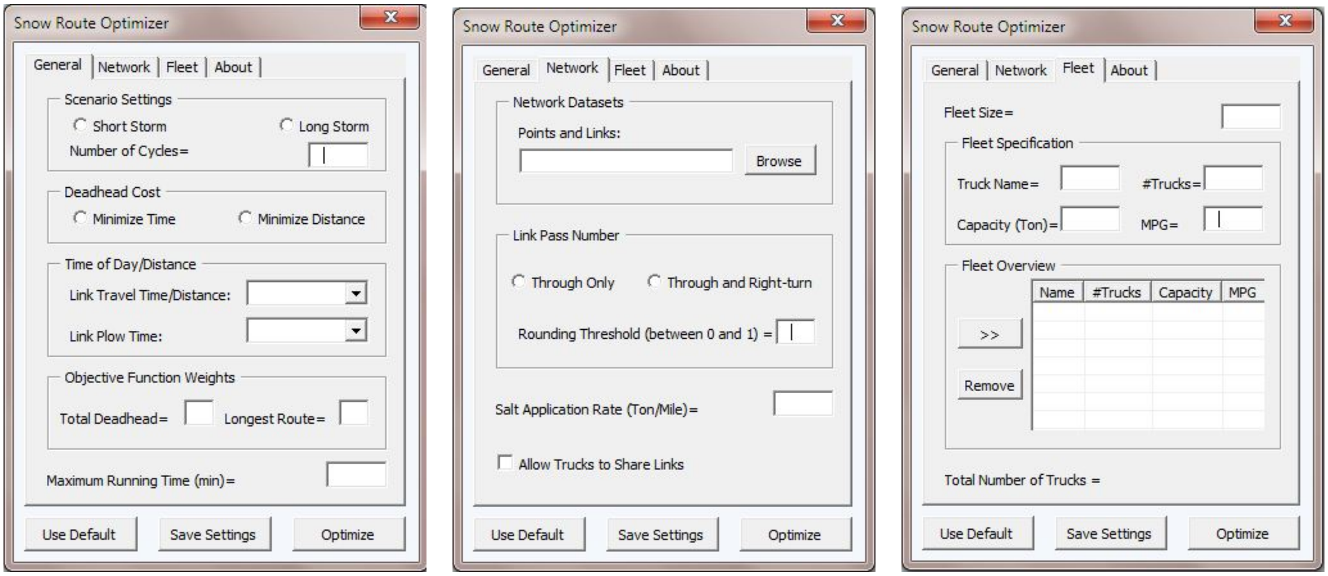


Figure 6.1: User form interface.

The users can easily change system parameters (e.g., vehicle fleet information, salt application rate). After saving the settings, the users can click on the “Optimize” button to run the optimization program. Progress of the optimization will be shown in a DOS window at selected major steps of the optimization algorithm (Figure 6.2).

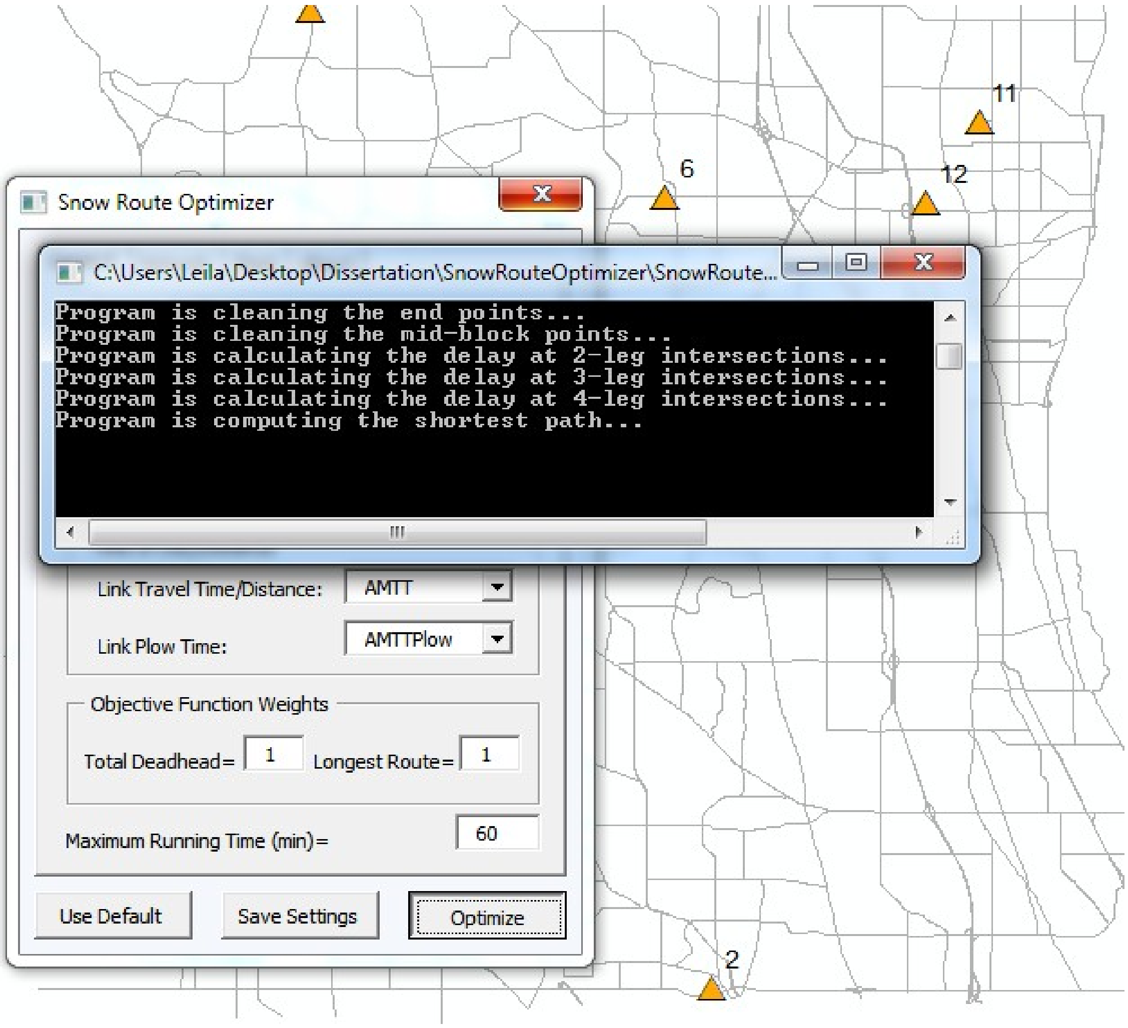


Figure 6.2: Optimize button and progress window.

After running the optimization module, the "Snow Route Optimizer Results" form (i.e., the second user interface) will appear (Figure 6.3). This module provides various GIS options to visualize the model output (e.g., optimized routes, performance statistics and summary) in tabular form. All the truck routes can also be shown on the map to provide insights on how balanced the snow routes are. Furthermore, the system is capable of detailed display of each truck's route, including step-by-step driver instructions and visual animations of the vehicle movements.

To generate these functionalities, first the output text results from the optimization module (i.e., all trucks' routes and step-by-step driver instructions) are read as tables in ArcGIS. Then a new spatial layer with the same properties of the original data (e.g., same coordinate system)is generated and new fields are created in the layer based on the fields' properties in the original data table and the output table (i.e., name, type, precision, length, etc.). Afterwards, the fields related to both tables in the new layer are populated by creating and writing records from the corresponding tables multiple times (due to the redundancies for each link in the step-by-step driver instructions, since a link may be visited more than once). This step cannot be done by a simple "join" process since we do force redundancies in the new layer. After running the optimization module, this new layer is automatically added in the list of GIS layers. This layer is connected to the map and includes all the information related to all the trucks, therefore all of the analyses (via the "Results" user form) are performed using the new spatial layer.

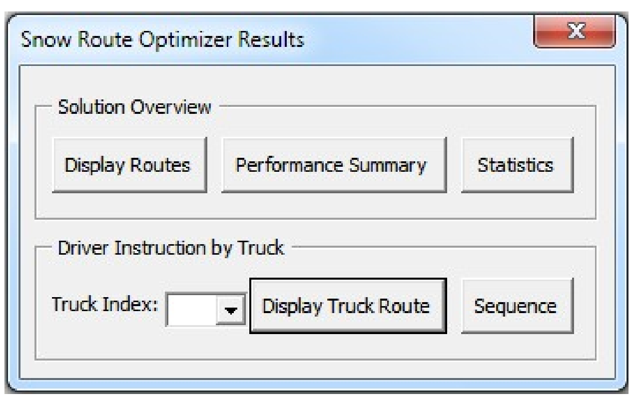


Figure 6.3: Results window.

The output snow plow routes will be highlighted on the GIS network map if the users click on "Display Routes" button, under solution overview; all the optimal truck routes will appear on the map with different colors (Figure 6.4).

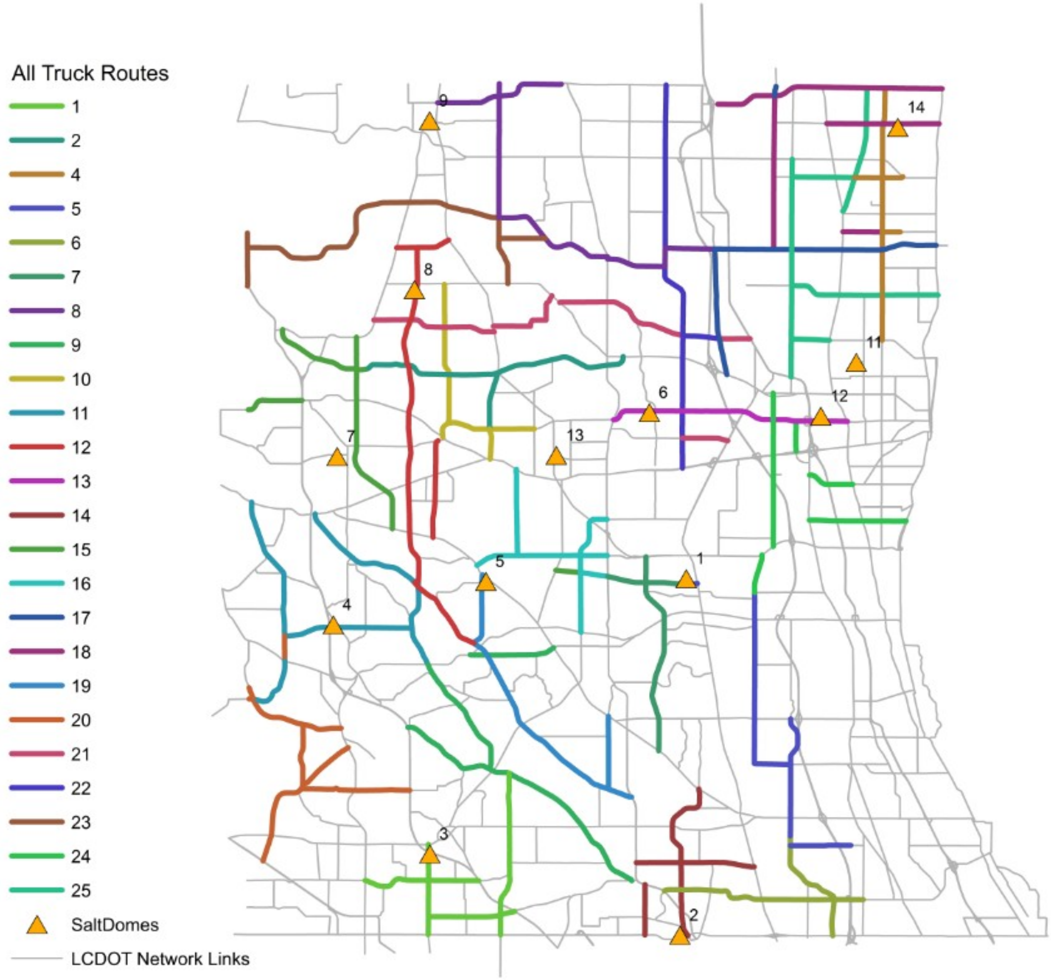
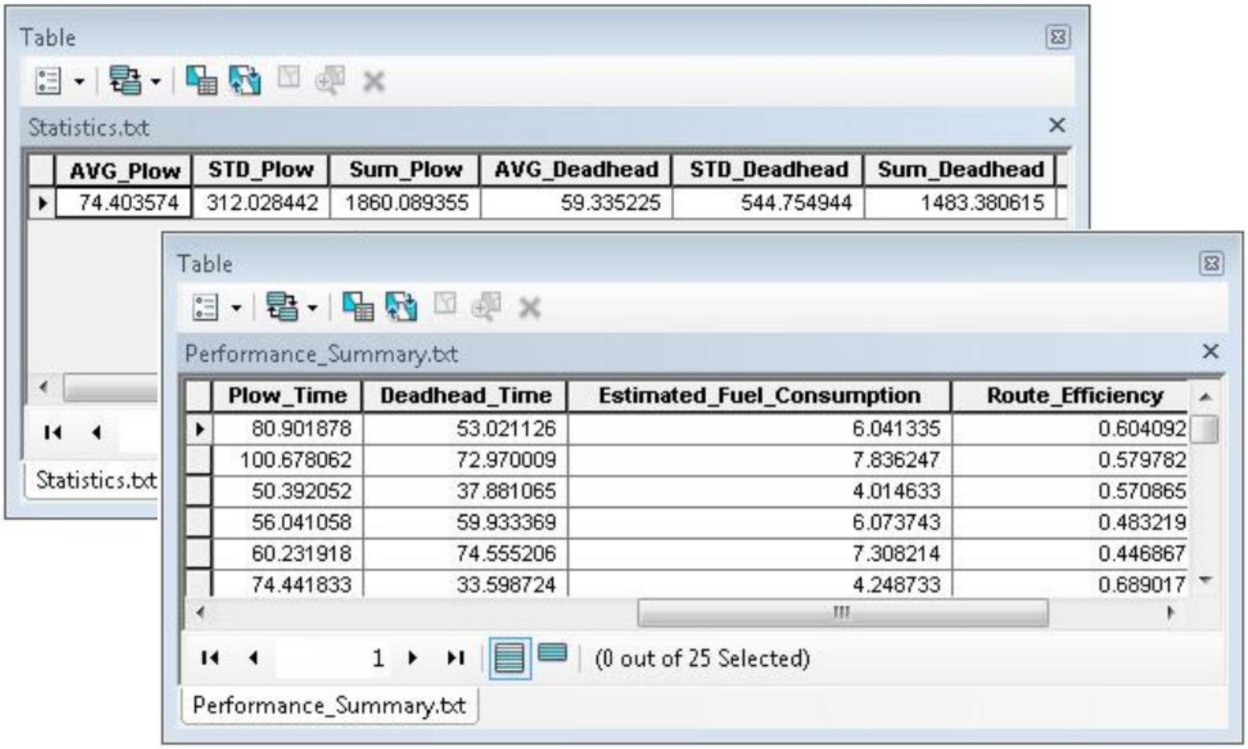


Figure 6.4: Display of all truck routes.

System performance and statistics will then be reported. If the users click on the "Performance Summary" button, the routing performance measures will appear in a table. This table indicates the total cycle time, plow time, deadhead time, estimated fuel consumption, etc. of each truck. If the users click on the "Statistics" button, the routing statistics will appear in a table. This table includes the mean and standard deviation of plow time, deadhead time, and total time across all the trucks in the fleet (Figure 6.5). The system performance and statistics provide a scientific basis for decision makers to conduct "what-if" analyses on fleet management and resource allocation, etc.



Table

Statistics.txt

AVG_Plow	STD_Plow	Sum_Plow	AVG_Deadhead	STD_Deadhead	Sum_Deadhead
74.403574	312.028442	1860.089355	59.335225	544.754944	1483.380615

Table

Performance_Summary.txt

Plow_Time	Deadhead_Time	Estimated_Fuel_Consumption	Route_Efficiency
80.901878	53.021126	6.041335	0.604092
100.678062	72.970009	7.836247	0.579782
50.392052	37.881065	4.014633	0.570865
56.041058	59.933369	6.073743	0.483219
60.231918	74.555206	7.308214	0.446867
74.441833	33.598724	4.248733	0.689017

1 (0 out of 25 Selected)

Performance_Summary.txt

Figure 6.5: Performance summary for each truck and statistics for all trucks.

The users can select a truck index and click the "Display Truck Route" button to check the detailed information of that truck's route sequence. The selected truck's route will be shown on the map as well as a separate table (Figure 6.6).

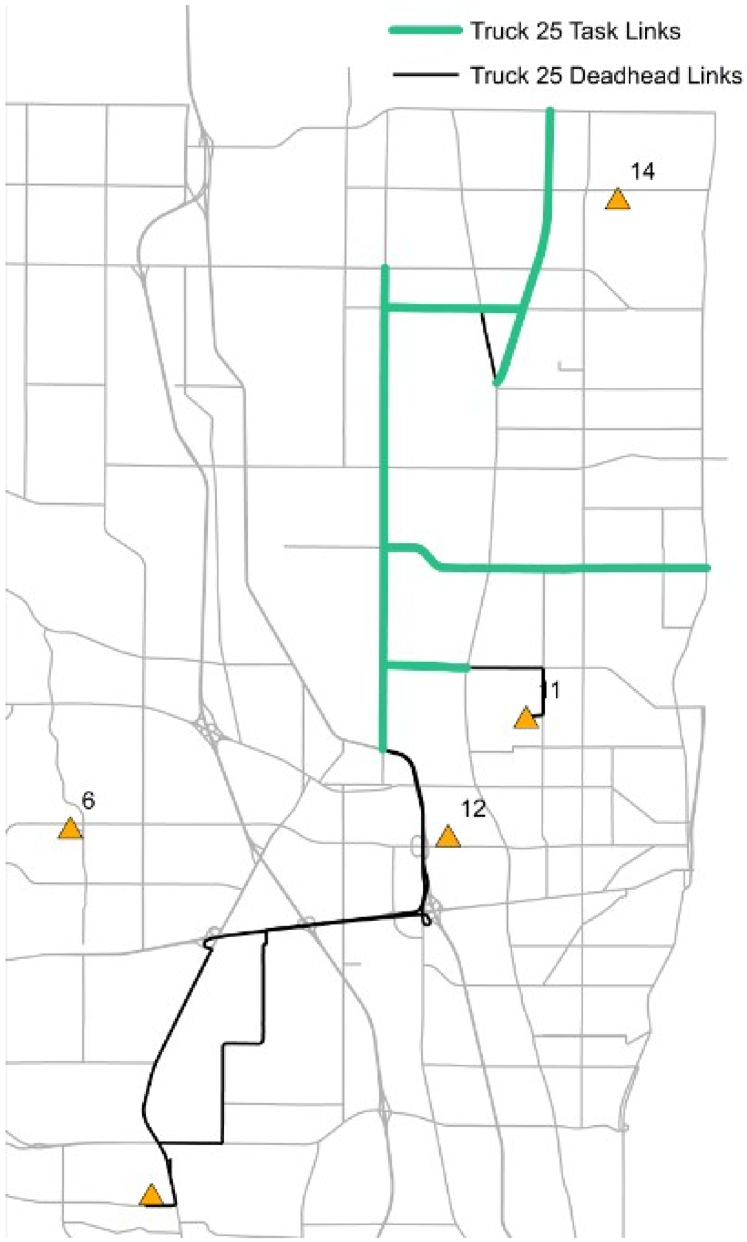


Figure 6.6: Route sequence for each truck.

The users can select a truck index and click the “Sequence” button to animate the truck’s roadway links in detail.

Chapter 7

Conclusions and Future Research

7.1 Conclusions

Winter storms and other snow events have a significant impact on traffic operations and safety. Ensuring safe mobility of passengers and travelers in adverse winter weather is a difficult task. It requires timely, expedient, and cost effective planning and operation of snow control activities. This dissertation addresses snow plow operations, their dynamic fleet management, and strategic planning for resource replenishment locations.

Winter maintenance operations on roadways involve truck routing, plowing, salting, and salt replenishment. Currently, planning of annual snow plow routes is not yet a fully-automated process for most public agencies. On the other hand, total operation and dead-head times of each truck is affected by the route choices and time-of-day traffic congestion. It is important and yet challenging to find the optimum solution to the snow plow routing problem. Such planning problems become even more complex when sets of extra operational constraints are required. There are a number of complicating implementation issues that must be considered, such as (a) salting distance for a route must match the truck capacity and (b) the trucks have only limited u-turn locations in the network. It is also common in practice to prioritize roadway segments (e.g., based on traffic volume) so that the plowing and salting operation maximizes certain objectives (e.g., safety). Similarly, higher level of priority can be assigned to roadway links that provide access to critical facilities (e.g., police stations, fire stations, hospitals).

In order to find the optimum solution to the snow plow routing problem (Chapter 3), this dissertation proposes mathematical models that aim to not only minimize the total snow plow truck travel times but also balance the distribution of such travel times for multiple classes of priority. The objective of the formulation includes the weighted sum of the total deadhead travel time and longest individual snow plow truck cycle time. A set of customized construction and local search methods is developed to effectively solve the problem. Empirical case study with real-world data shows that the proposed solution approach is able to optimize snow routes (with or without considering priorities of plowing tasks) in a short amount of time. Furthermore, our model results outperform the current solution in practice. We also develop a state-of-art snow routing software with optimization modules and user-friendly GIS interfaces for snow route analysis and design. This decision-support software (Chapter 6) optimizes a set of snow plow routes based on a set of user input parameters, and it can help stake-holders, engineers, and planners evaluate snow plow options (such as salt usage, vehicle capacities, fleet size, plowing time during the day) and provide recommendations on vehicle assignments to snow routes. It also includes sufficient flexibility such that experts can further fine-tune the results before field implementation.

On the other hand, winter maintenance operations such as snow and ice control services are themselves subject to the adverse impacts of snow storms, and hence they require careful planning. Advance planning of such activities, however, is often very challenging, since (i) storm events are stochastic in nature as their start time, severity, impact areas, and duration generate uncertain maintenance demand, and (ii) maintenance trucks may not be readily available at all times as a result of possible mechanical failures or traffic congestion. These issues affect the overall service reliability, especially in high priority regions where timely snow removal is critical. Therefore, fleet management that accounts for uncertainties in a flexible and demand-responsive fashion is appealing.

This dissertation (i.e., Chapter 4) addresses a fleet assignment problem for snow plow trucks under service disruptions and demand uncertainty. A stochastic dynamic fleet management model is developed to assign available trucks to cover uncertain snow plowing demand. The objective is to simultaneously minimize the cost for truck deadheading and repositioning, as well as to maximize the benefits (i.e., level of service) of plowing. The problem is formulated into a dynamic programming model and solved using an approximate dynamic programming algorithm. Piece-wise linear functional approximations are used to estimate the value function of system states (i.e., snow plow trucks location over time). We apply our model and solution approach to a snow plow operation scenario for Lake County, Illinois. Besides, a set of sensitivity analyses is conducted to capture the impact of the cost/revenue parameter values on the optimal decisions at each stage and total benefit over the planning horizon. Numerical results show that the proposed algorithm can solve the problem effectively and outperforms a rolling-horizon heuristic solution.

Furthermore, snow shall be plowed in a manner so as to minimize traffic obstructions during the service trucks' operations in transportation networks. However, snow plow trucks are relatively heavy in weight and often have difficulty turning at intersections and thus, affect the traffic operations and may contribute to additional congestion during their service. Thus, expanding the roadways (e.g., widening roadway shoulders) can improve roadway capacity and increase truck driver comfort, especially near resource replenishment facilities that trucks visit during maintenance operations. On the other hand, number and location of satellite resource replenishment facilities can also affect the efficiency of maintenance trucks' operations. Hence, it is beneficial to simultaneously consider a strategic plan for facility location design as well as transportation network expansion (especially in the neighborhood of the salt replenishment locations) to facilitate traffic operations and maintain the service level. Furthermore, routing cost of service trucks under these strategic network decisions shall also be considered.

This dissertation (i.e., Chapter 5) develops an integrated mathematical model for salt dome facility location design, which determines the near-optimum number and location of the salt domes, the near-optimal traffic assignment (both general roadway users and snow plow trucks), snow plow trucks routing cost based on near-optimal network design, and possible roadway capacity expansion. The objective is to minimize the total cost for salt dome facility construction, transportation infrastructure expansion, transportation delay (for both snow plow truck movements and public travel), as well as deadhead travel. A genetic algorithm framework (with embedded traffic assignment algorithm and continuous approximation (CA) model for truck routing cost estimation) is developed. The numerical results show that the integrated solution technique can solve the problem effectively. It shall be noted that although this research focuses on the strategic network design for salt dome facilities and snow plow roadway transportation, the model and solution techniques are suitable for a number of application contexts that simultaneously involve network traffic equilibrium, truck routing, infrastructure expansion, and facility location choices (which determine the origin/destination of multi-commodity flow).

7.2 Future Research

This research in general provides effective decision-making tools to facilitate complex strategic planning as well as operational management level problems. Future research in these areas can be generalized in various directions (see Figure 7.1). One extension can be on developing further models and solution techniques (or extending our existing models/algorithms) to determine transportation infrastructure decisions (e.g., roadway capacity expansion) and their impacts on multi-modal travels. This can be applied on various other case studies than service trucks operations such as transportation planning for freight logistics. Another extension in this area is to determine the optimal pavement rehabilitation plan as a result of traffic load frequency and heavy vehicles movements on the roadways (see Hajibabai et al. (2014a)). Furthermore, the developed models and solution techniques for dynamic snow plow trucks' management can be extended for variety of other studies such as freight truck operations, port management, task assignments in an industry (e.g., activity schedules as well as construction equipments' task assignment in job sites), etc. Several integrated models can be developed in the future to include different components of this research. It will be interesting to study the environmental impacts of such transportation systems (e.g., pollutants and greenhouse gas emissions) in the context of environmental sustainability. Integrated models can be developed with embedded modules to reduce those impacts while operations of trucks/passenger traffic are being optimized.

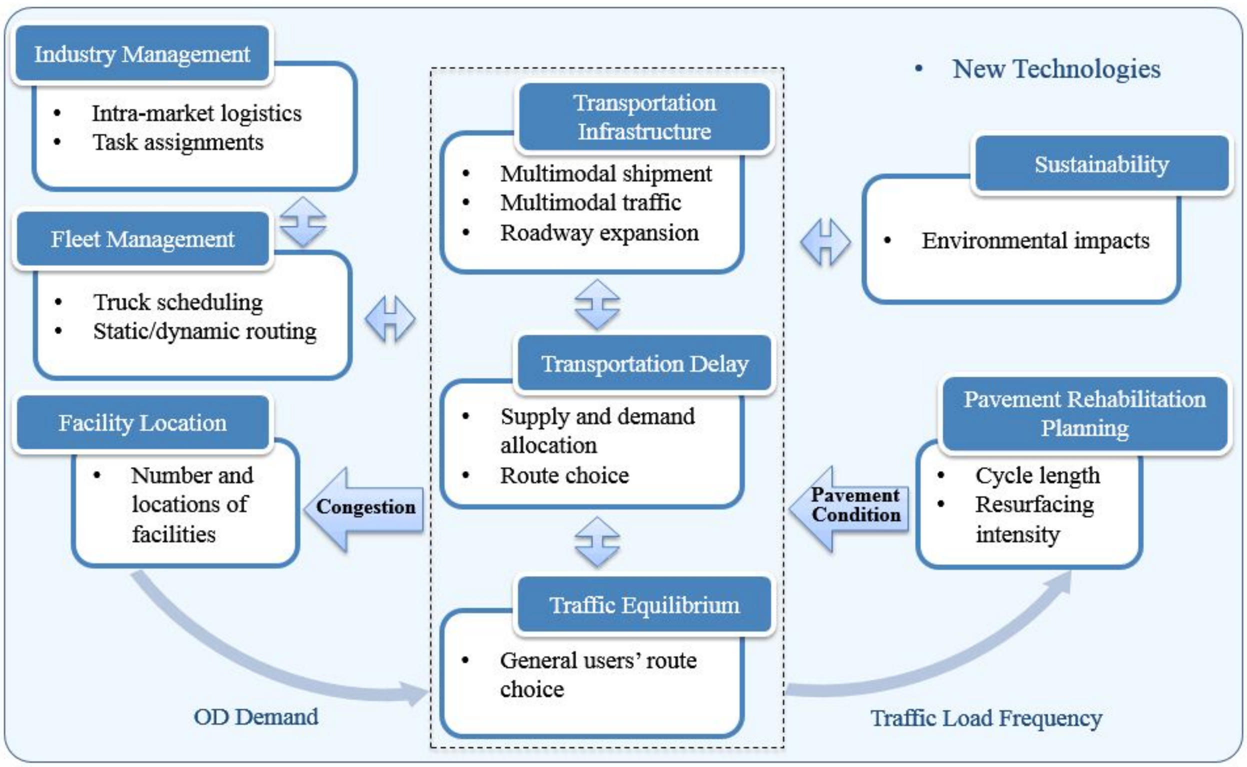


Figure 7.1: Route sequence for each truck.

Future research in the area of network routing (i.e., Chapter 3) can be generalized in a few directions. In this dissertation, only one cycle routing is formulated for each vehicle in a short snow storm. A possible extension is to develop a model formulation for a long-storm scenario under two possible options: (i) having exactly the same truck path sequence in different cycles, and (ii) enforcing a minimum time separation between two consecutive plowing of the same task links. In addition, in this dissertation the possibility of system breakdown (e.g., satellite salt depot disruption or truck failure) is ignored; it would be interesting in the future to solve snow plow routing problem under stochastic reliability constraints. It would also be interesting to apply our model and solution approach to a comprehensive list of diverse networks for additional real-world case studies.

Future research in the area of dynamic fleet management (i.e., Chapter 4) can be conducted in a few directions. Our proposed model is based on several cost and reward parameters (i.e., deadhead cost, repositioning cost, and task performance reward). One of the assumptions in the proposed model is that the time required for a maintenance truck to move between any pair of routes (i.e., repositioning time) is a single time period. This assumption may not be true in large networks where the distance between the links is large and/or the time period duration is short. In this research, an acceptable length for the time period is selected based on the size of the study network. It will be interesting to study the impact of the time period length on the performance of the proposed model. Besides, a possible extension to this work is to develop more benchmark solution strategies and compare the numerical results following the proposed techniques. This will help demonstrate the performance of each alternative algorithm based on the properties of our proposed model. It can also provide valuable insights for practical purposes. It is also interesting to consider different approximation methods for value function approximations. Furthermore, incorporating live traffic information as well as on-line location data from the service trucks can improve model realism and lead to more efficient snow plow fleet management during adverse winter conditions. As the practical side of this research, it will be interesting in the future to provide the results of this study to the corresponding winter maintenance agencies with some opportunity/mechanism to use the system in their fleet management under uncertain winter conditions and further validate the results and conclusions. Another valuable extension will be on modeling other constraints for driver behavior (e.g., arrival time of drivers, level of information we provide for drivers while they are driving, etc.) as well as work shift/schedule (e.g., maximum in-vehicle time for a driver before a break, etc.). Constraints can also be added to the model to ensure that drivers are not re-routed before finishing their current re-route; this makes sure that we are not enforcing an unnecessary stop and go condition as it is not as efficient as keep going condition (i.e., having drivers performed their assigned tasks instead of re-routing them all the time). The extended model will also be treated as stochastic. It is also interesting in the future to account for different truck capacities in our model as we assume all truck are equal in our existing models. Accounting for incidents is another interesting direction as they involve in traffic congestion which affects the operations of the service trucks (e.g., either the trucks are involved in the incidents, i.e., truck failure, or they are stuck in the traffic queue).

Future research in the strategic planning area (i.e., Chapter 5) can be generalized in the following directions. A possible extension is to develop a heuristic solution algorithm to evaluate the performance of our genetic algorithm framework with embedded traffic assignment technique and routing approximation module. This research has assumed that the capital investment associated with roadway expansion is considered as part of the roadway maintenance operation investment plan. This is possible under potential public-private partnerships (Unnikrishnan et al., 2009). It may be interesting to expand our current framework to allow multiple stakeholders (e.g., public agency, winter maintenance agency, and public travelers) to have independent or conflicting objectives. This will probably require multi-level programs with equilibrium and routing constraints. In addition, as we observe the impact of roadway capacity expansion on the total system cost, it would be interesting to study in the future how roadway expansion budget would affect the optimal solution. Another interesting extension is to develop efficient algorithms to solve dynamic location-routing problem and account for stochasticities (i.e., truck availability, etc.) within the general framework. We can also include different truck specifications (e.g., truck capacity) in our model. In addition, several combinations of the cost components in the objective function can be examined to draw some managerial insights on the performance/practicability of the proposed approaches. On the other hand, the background traffic is assumed fixed in this research. It will be very interesting to develop efficient algorithms to include traffic equilibrium for general roadway users' route choice. This will definitely add another level of complexity to our current problem. It would also be interesting to apply our model and solution approach to a comprehensive list of diverse networks for additional real-world case studies.

References

Akyüz, M. H., Öncan, T., Altınel, İ. K., 2012. Efficient approximate solution methods for the multi-commodity capacitated multi-facility weber problem. Computers & Operations Research 39 (2), 225–237.

Anez, J., De La Barra, T., Perez, B., 1996. Dual graph representation of transport networks. Transportation Research Part B: Methodological 30 (3), 209–216.

Aziz, H., Ukkusuri, S. V., 2012. Integration of environmental objectives in a system optimal dynamic traffic assignment model. Computer-Aided Civil and Infrastructure Engineering 27 (7), 494–511.

Bai, Y., Hwang, T., Kang, S., Ouyang, Y., 2011. Biofuel refinery location and supply chain planning under traffic congestion. Transportation Research Part B 45 (1), 162–175.

Bai, Y., Ouyang, Y., Pang, J., 2012. Biofuel supply chain design under competitive agricultural land use and feedstock market equilibrium. Energy Economics 34 (5), 1623–1633.

Bertsekas, D., Tsitsiklis, J., 1996. Neuro-Dynamic Programming. Athena Scientific, Belmont, MA.

Bertsimas, D. J., Simchi-Levi, D., 1996. A new generation of vehicle routing research: robust algorithms, addressing uncertainty. Operations Research 44 (2), 286–304.

Birge, J. R., 1985. Decomposition and partitioning methods for multistage stochastic linear programs. Operations Research 33 (5), 989–1007.

Birge, J. R., Louveaux, F., 2011. Introduction to stochastic programming. Springer.

BPR, 1970. U.S. Bureau of Public Roads. Urban Transportation Planning. General Information and Introduction to System 360. Washington. D.C. (U.S. Department of Transportation).

Bräysy, O., Gendreau, M., 2005. Vehicle routing problem with time windows, part i: Route construction and local search algorithms. Transportation science 39 (1), 104–118.

C2Logix, 2011. "c2logix" software @ONLINE.
URL `http://www.c2logix.com/`

Carvalho, T. A., Powell, W. B., 2000. A multiplier adjustment method for dynamic resource allocation problems. Transportation Science 34 (2), 150–164.

Chen, Z.-L., Powell, W. B., 1999. Convergent cutting-plane and partial-sampling algorithm for multistage stochastic linear programs with recourse. Journal of Optimization Theory and Applications 102 (3), 497–524.

Cheung, R. K., Powell, W. B., 1996. An algorithm for multistage dynamic networks with random arc capacities, with an application to dynamic fleet management. Operations Research 44 (6), 951–963.

Christofides, N., Campos, V., Corberán, A., Mota, E., 1986. An algorithm for the rural postman problem on a directed graph. In: Netflow at Pisa. Springer, pp. 155–166.

Clarke, L., 2004. Embedding CPLEX using ILOG concert technology. Prezentacja. ILOG Inc.

Daganzo, C. F., 2005. Logistics systems analysis. Springer.

Dantzig, G. B., 1955. Linear programming under uncertainty. Management science 1 (3-4), 197–206.

Dantzig, G. B., Infanger, G., 1991. Large-scale stochastic linear programs: Importance sampling and benders decomposition. Tech. rep., DTIC Document.

Daskin, M., 1995. Network and Discrete Location: Models, Algorithms, and Applications. John Wiley and Sons, New York.

Drezner, Z., 1995. Facility Location: A Survey of Application and Methods. Springer, Berlin.

Eathington, L., Swenson, D., 2007. Dude, wheres my corn? constraints on the location of ethanol production in the corn belt. Department of Economics, Iowa State University.

Eiselt, H. A., Gendreau, M., Laporte, G., 1995. Arc routing problems, part ii: The rural postman problem. Operations Research 43 (3), 399–414.

Fisher, M., 1981. The lagrangian relaxation method for solving integer programming problems. Management science 27 (1), 1–18.

Frank, M., Wolfe, P., 1956. An algorithm for quadratic programming. Naval research logistics quarterly. Wiley Online Library 3 (1-2), 95–110.

Frantzeskakis, L. F., Powell, W. B., 1990. A successive linear approximation procedure for stochastic, dynamic vehicle allocation problems. Transportation Science 24 (1), 40–57.

Funke, B., Grünert, T., Irnich, S., 2005. Local search for vehicle routing and scheduling problems: Review and conceptual integration. Journal of heuristics 11 (4), 267–306.

Gao, L., Xie, C., Zhang, Z., Waller, S. T., 2012. Network-level road pavement maintenance and rehabilitation scheduling for optimal performance improvement and budget utilization. Computer-Aided Civil and Infrastructure Engineering 27 (4), 278–287.

Gendreau, M., Hertz, A., Laporte, G., 1992. New insertion and postoptimization procedures for the traveling salesman problem. Operations Research 40 (6), 1086–1094.

Gendreau, M., Laporte, G., Séguin, R., 1996. Stochastic vehicle routing. European Journal of Operational Research 88 (1), 3–12.

Glover, F., 1992. New ejection chain and alternating path methods for traveling salesman problems. Computer Science and Operations Research 1992, 449–509.

Godfrey, A., Powell, W. B., 2001. An adaptive, distribution-free algorithm for the newsvendor problem with censored demands, with applications to inventory and distribution. Management Science 47 (8), 1101–1112.

Godfrey, A., Powell, W. B., 2002a. An adaptive dynamic programming algorithm for dynamic fleet management, i: Single period travel times. Transportation Science 36 (1), 21–39.

Goldberg, D., 1989. Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley, Boston.

Graves, S. C., Willems, S. P., 2005. Optimizing the supply chain configuration for new products. Management Science 51 (8), 1165–1180.

Gregory, A., Powell, W. B., 2002b. An adaptive dynamic programming algorithm for dynamic fleet management, ii: multiperiod travel times. Transportation Science 36 (1), 40–54.

Haghani, A., Qiao, H., 2001. Decision support system for snow emergency vehicle routing: Algorithms and application. Transportation Research Record: Journal of the Transportation Research Board 1771 (1), 172–178.

Hajibabai, L., Bai, Y., Ouyang, Y., 2014a. Joint optimization of freight facility location and pavement infrastructure rehabilitation under network traffic equilibrium. Transportation Research Part B: Methodological 63, 38–52.

Hajibabai, L., Nourbakhsh, S., Ouyang, Y., Peng, F., 2014b. Network routing of snow plows with resource replenishment and plowing priorities: Formulation, algorithm, and application. Transportation Research Record: Journal of the Transportation Research Board, In Press.

Hajibabai, L., Ouyang, Y., 2013. Integrated planning of supply chain networks and multimodal transportation infrastructure expansion: Model development and application to the biofuel industry. Computer-Aided Civil and Infrastructure Engineering 28 (4), 247–259.

Hanna, A., Maze, T., Albrecht, C., Kroeger, D., Wiegand, J., 2009. Performance measures for snow and ice control operations. NCHRP Research Results Digest (335).

HCM, 2000. Highway Capacity Manual. Transportation Research Board, National Research Council. Washington, D.C. (0-309-06681-6).

Higle, J. L., Sen, S., 1991. Stochastic decomposition: An algorithm for two-stage linear programs with recourse. Mathematics of operations research 16 (3), 650–669.

IDOT, 2011. Illinois department of transportation (idot) statistical maps of illinois-average annual daily traffic maps. Accessed on May 29, 2014.
URL `http://idot.ms2soft.com/tms/`

Infanger, G., 1994. Planning under uncertainty: solving large-scale stochastic linear programs. Boyd & Fraser Danvers, MA.

Ioannou, G., Kritikos, M., Prastacos, G., 2001. A greedy look-ahead heuristic for the vehicle routing problem with time windows. Journal of the Operational Research Society 52 (5), 523–537.

Jaillet, P., Lu, X., 2011. Online traveling salesman problems with service flexibility. Networks 58 (2), 137–146.

Jaillet, P., Wagner, M. R., 2008. Generalized online routing: New competitive ratios, resource augmentation, and asymptotic analyses. Operations research 56 (3), 745–757.

Jayaram, K., 2005. The impact of road traffic congestion on supply chain: insights from auckland, new zealand. International Journal of Logistics 8 (2), 159–180.

Jen, F. C., Pegels, C. C., Dupuis, T. M., 1968. Optimal capacities of production facilities. Management Science 14 (10), B–573 – B–580.

Jordan, W. C., Turnquist, M. A., 1983. A stochastic, dynamic network model for railroad car distribution. Transportation Science 17 (2), 123–145.

Kall, P., Wallace, S. W., 1994. Stochastic programming. John Wiley and Sons Ltd.

Kanungo, T., Mount, D., Netanyahu, N., Piatko, C., Silverman, R., Wu, A., 2002. An efficient k-means clustering algorithm: Analysis and implementation. Pattern Analysis and Machine Intelligence, IEEE Transactions on 24 (7), 881–892.

Kirby, R. F., Potts, R. B., 1969. The minimum route problem for networks with turn penalties and prohibitions. Transportation Research 3 (3), 397–408.

Klose, A., Drexl, A., 2005. Facility location models for distribution system design. European Journal of Operational Research 162 (1), 4–29.

Konur, D., Geunes, J., 2011. Analysis of traffic congestion costs in a competitive supply chain. Transportation Research Part E 47 (1), 1–17.

Laporte, G., 1992. The vehicle routing problem: An overview of exact and approximate algorithms. European Journal of Operational Research 59 (3), 345–358.

Li, Z., Madanu, S., Zhou, B., Wang, Y., Abbas, M., 2010. A heuristic approach for selecting highway investment alternatives. Computer-Aided Civil and Infrastructure Engineering 25 (6), 427–439.

Lin, D.-Y., 2011. A dual variable approximation-based descent method for a bi-level continuous dynamic network design problem. Computer-Aided Civil and Infrastructure Engineering 26 (8), 581–594.

López, E., Monzón, A., 2010. Integration of sustainability issues in strategic transportation planning: A multi-criteria model for the assessment of transport infrastructure plans. Computer-Aided Civil and Infrastructure Engineering 25 (6), 440–451.

MTP, 2006. Metropolitan Transportation Plan 2035. Road expansion, Sacramento region, Blueprint for Sustainable Communities 2006. Accessed on June 7, 2010.
URL `http://www.sacog.org/2035`

Ng, M., Lin, D. Y., Waller, S. T., 2009. Optimal long-term infrastructure maintenance planning accounting for traffic dynamics. Computer-Aided Civil and Infrastructure Engineering 24 (7), 459–469.

Nie, Y. M., Zhang, H. M., 2008. A variational inequality formulation for inferring dynamic origin–destination travel demands. Transportation Research Part B 42 (7), 635–662.

Novoa, C., Storer, R., 2009. An approximate dynamic programming approach for the vehicle routing problem with stochastic demands. European Journal of Operational Research 196 (2), 509–515.

Or, I., 1976. Traveling salesman-type combinatorial problems and their relation to the logistics of regional blood banking. Ph.D. thesis, Northwestern University, Evanston, Illinois.

Ouyang, Y., 2007. Pavement resurfacing planning on highway networks: A parametric policy iteration approach. ASCE Journal of Infrastructure Systems 13 (1), 65–71.

Pallottino, S., Scutella, M. G., 1998. Shortest path algorithms in transportation models: classical and innovative aspects. In: Equilibrium and advanced transportation modelling. Springer, pp. 245–281.

Peng, F., Kang, S., Li, X., Ouyang, Y., Somani, K., Acharya, D., 2011. A heuristic approach to the railroad track maintenance scheduling problem. Computer-Aided Civil and Infrastructure Engineering 26 (2), 129–145.

Peng, F., Ouyang, Y., 2012. Track maintenance production team scheduling in railroad networks. Transportation Research Part B: Methodological 46 (10), 1474–1488.

Peng, F., Ouyang, Y., 2014. Optimal clustering of railroad track maintenance jobs. Computer-Aided Civil and Infrastructure Engineering 29 (4).

Peng, F., Ouyang, Y., Somani, K., 2014. Scheduling of periodic inspections in large-scale railroad networks. Journal of Rail Transport Planning & Management. In press. DOI: 10.1016/j.jrtpm.2014.02.003.

Pereira, M., Pinto, L. M., 1991. Multi-stage stochastic optimization applied to energy planning. Mathematical Programming 52 (1-3), 359–375.

Perrier, N., Langevin, A., Amaya, C., 2008. Vehicle routing for urban snow plowing operations. Transportation Science 42 (1), 44–56.

Perrier, N., Langevin, A., Campbell, J., 2006. A survey of models and algorithms for winter road maintenance. part i: system design for spreading and plowing. Computers and Operations Research 33 (1), 209–238.

Perrier, N., Langevin, A., Campbell, J., 2007a. A survey of models and algorithms for winter road maintenance. part iii: Vehicle routing and depot location for spreading. Computers and Operations Research 34 (1), 211–257.

Perrier, N., Langevin, A., Campbell, J., 2007b. A survey of models and algorithms for winter road maintenance. part iv: Vehicle routing and fleet sizing for plowing and snow disposal. Computers and Operations Research 34 (1), 258–294.

Pillac, V., Gendreau, M., Guéret, C., Medaglia, A. L., 2013. A review of dynamic vehicle routing problems. European Journal of Operational Research 225 (1), 1–11.

Potvin, J., Rousseau, J., 1993. A parallel route building algorithm for the vehicle routing and scheduling problem with time windows. European Journal of Operational Research 66 (3), 331–340.

Potvin, J., Rousseau, J., 1995. An exchange heuristic for routeing problems with time windows. Journal of the Operational Research Society 46 (12), 1433–1446.

Powell, W. B., 1986. A stochastic model of the dynamic vehicle allocation problem. Transportation science 20 (2), 117–129.

Powell, W. B., 1987. An operational planning model for the dynamic vehicle allocation problem with uncertain demands. Transportation Research Part B 21 (3), 217–232.

Powell, W. B., Carvalho, T. A., 1998. Dynamic control of logistics queueing networks for large-scale fleet management [1]. Transportation Science 32 (2).

Prékopa, A., 1995. Stochastic Programming. Springer.

Puterman, M. L., 2009. Markov Decision Processes: Discrete Stochastic Dynamic Programming. Vol. 414. John Wiley & Sons.

Putha, R., Quadrifoglio, L., Zechman, E., 2012. Comparing ant colony optimization and genetic algorithm approaches for solving traffic signal coordination under oversaturation conditions. Computer-Aided Civil and Infrastructure Engineering 27 (1), 14–28.

Regan, A. C., Mahmassani, H. S., Jaillet, P., 1996. Dynamic decision making for commercial fleet operations using real-time information. Transportation Research Record: Journal of the Transportation Research Board 1537 (1), 91–97.

Rockafellar, R. T., Wets, R. J.-B., 1991. Scenarios and policy aggregation in optimization under uncertainty. Mathematics of operations research 16 (1), 119–147.

RouteSmart, 2011. "routesmart" software @ONLINE.
URL `http://www.routesmart.co.uk/`

Russell, R., 1977. An effective heuristic for the m-tour traveling salesman problem with some side conditions. Operations Research 25 (3), 517–524.

Salazar-Aguilar, M., Langevin, A., Laporte, G., 2012. Synchronized arc routing for snow plowing operations. Computers & Operations Research 39 (7), 1432–1440.

Salim, M., Timmerman, M., Strauss, T., Emch, M., 2002. Artificial-intelligence-based optimization of the management of snow removal assets and resources. Midwest Transportation Consortium (MTC): Iowa State University and University of Missouri-Columbia and Lincoln University and University of Missouri-Kansas City and University of Missouri-St. Louis and University of Northern Iowa, Tech. Report.

Savelsbergh, M. W., 1992. The vehicle routing problem with time windows: Minimizing route duration. ORSA journal on computing 4 (2), 146–154.

Schenk, L., Klabjan, D., 2008. Intramarket optimization for express package carriers. Transportation Science 42 (4), 530–545.

Schenk, L., Klabjan, D., 2010. Intra market optimization for express package carriers with station to station travel and proportional sorting. Computers & Operations Research 37 (10), 1749–1761.

Secomandi, N., 2001. A rollout policy for the vehicle routing problem with stochastic demands. Operations Research 49 (5), 796–802.

Secomandi, N., Margot, F., 2009. Reoptimization approaches for the vehicle-routing problem with stochastic demands. Operations Research 57 (1), 214–230.

Sheffi, Y., 1985. Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Methods. Prentice Hall, England Cliffs, NJ.

Shen, M., Qi, L., 2007. Incorporating inventory and routing costs in strategic location models. European Journal of Operational Research 179 (2), 372–389.

Sochor, J., Yu, C., 2004. A heuristic method for routing snowplows after snowfall. Master thesis, Division of Optimization, Department of Mathematics, Linkoping Institute of Technology, Linkoping, Sweden,.

Solomon, M., 1987. Algorithms for the vehicle routing and scheduling problems with time window constraints. Operations research 35 (2), 254–265.

Sugumaran, R., Salim, M., Strauss, T., Fulcher, C., 2005. Web-based implementation of a winter maintenance decision support system using GIS and Remote Sensing. Midwest Transportation Consortium (MTC): University Transportation Centers Program, U.S. Department of Transportation, Tech. Report.

Sutton, R. S., Barto, A. G., 1998. Introduction to reinforcement learning. The MIT Press, Cambridge, MA.

Taillard, É., Badeau, P., Gendreau, M., Guertin, F., Potvin, J., 1997. A tabu search heuristic for the vehicle routing problem with soft time windows. Transportation science 31 (2), 170–186.

Thompson, P., Psaraftis, H., 1993. Cyclic transfer algorithms for multivehicle routing and scheduling problems. Operations research 41 (5), 935–946.

Ukkusuri, S. V., Mathew, T. V., Waller, S. T., 2007. Robust transportation network design under demand uncertainty. Computer-Aided Civil and Infrastructure Engineering 22 (1), 6–18.

Unnikrishnan, A., Lin, D.-Y., 2012. User equilibrium with recourse: continuous network design problem. Computer-Aided Civil and Infrastructure Engineering 27 (7), 512–524.

Unnikrishnan, A., Valsaraj, V., Damnjanovic, I., Waller, S. T., 2009. Design and management strategies for mixed public private transportation networks: A meta-heuristic approach. Computer-Aided Civil and Infrastructure Engineering 24 (4), 266–279.

Wang, J., Kandula, P., Wright, J., 1995. Evaluation of computer-generated routes to improved snow and ice control. Transportation Research Record (1509), 15–21.

Wang, X., Ouyang, Y., Yang, H., Bai, Y., 2013. Optimal biofuel supply chain design under consumption mandates with renewable identification numbers. Transportation Research Part B 57, 158–171.

Wilson, M., Dadie-Amoah, K., Zhang, Y., 2003. Snow plow operations and resource management. Office of Research Services, Minnesota Department of Transportation, Tech. Report.

Yang, J., Jaillet, P., Mahmassani, H., 2004. Real-Time Multivehicle Truckload Pickup and Delivery Problems. Transportation Science 38 (2), 135–148.

Appendix A

User Guide for Snow Route Optimizer

The developed software tool is embedded in the ArcGIS environment, which is named *Snow Route Optimizer*. This user guide briefly explains its set-up and functionalities. Section A.1 below explains the software installation procedure; Section A.2 describes different tabs of the graphical user interface for user settings; and Section A.3 shows the functionalities and result visualization.

A.1 Software Installation

This software is developed for ArcGIS Version 10.0 and it has been tested on personal PCs with Windows XP (64 bits) and 7 operating environments. We have not tested the software on other environments.

First, the software in the .rar folder ("SnowRouteOptimizer" software package) includes the optimization program (.exe file) and the GIS program (.mxd file). They should be saved in any suitable folder on the user?s computer.

This package includes two versions of the software, for Windows XP and Windows 7, respectively. Each version contains the following key files:

- a batch file that will install the required library files in support of running
- the optimization module, .exe file, and
- the “SnowRouteOptimizer” .mxd file, and

The package also includes the required library files for each computer operating environment and a text file containing the password to the VBA source code. The software should be installed in the following sequence: First, double-click the batch file to make sure that all the required library files exist; second, open the .mxd project file; third, prepare input data “LCDOTNetworkLinks.shp” (i.e., exporting the GeoDB file into a shape file); fourth, run the program by clicking the proper buttons.

Note: Since the graphical user interface (GUI) creates intermediate data files during the optimization process and the visualization procedure, the directory/path of these files as well as that of the .exe file must be the same.

A.2 Graphical User Interfaces

Upon successful installation, a toolbar is provided in the ArcGIS environment, which includes setting and results tools (see Figure A.1).

Figure A.1: Snow Route Optimizer toolbar: “Setting” and “Results” Tools.

When the “Setting” button is clicked, the Snow Route Optimizer settings window will pop up for the users to enter appropriate system parameters. The tabs and buttons will be described in detail in the following sections.

A.2.1 “General” Tab

This tab is for the “General” settings of the optimization application where the users can decide what scenario and optimization mode they prefer. In addition, the users can choose what time-of-day, cost metric, objective function weights, and maximum running time they favor for the optimized routes (Figure A.2).

In the “General” tab, under the “Deadhead Cost” frame, the users can choose whether they would like the optimizing software to minimize deadhead time or deadhead distance. Under the “Time of Day/Distance” frame, the users are able to choose the time-of-day and the appropriate traffic delay, i.e., morning/evening, minimum or maximum travel times, etc. (Figure A.3 (a)-(b)).

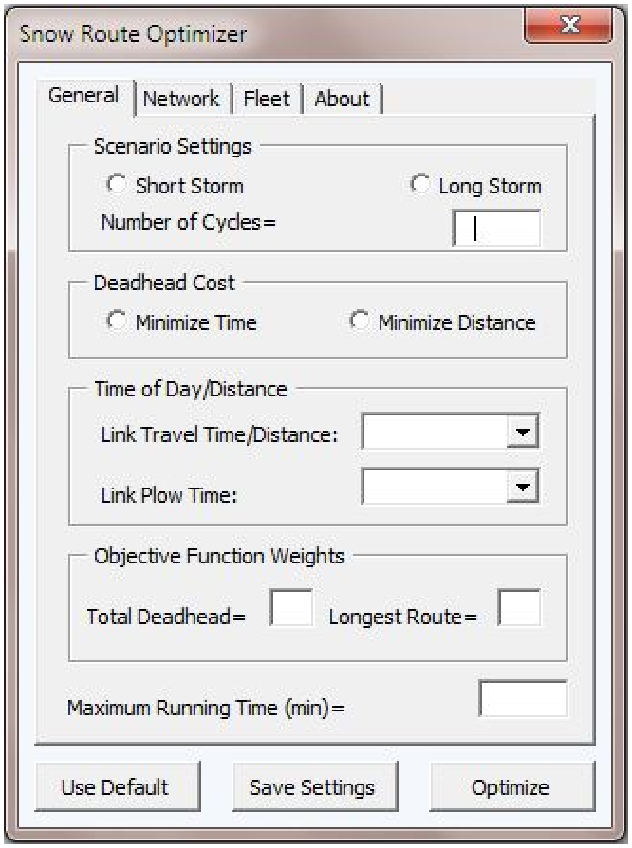
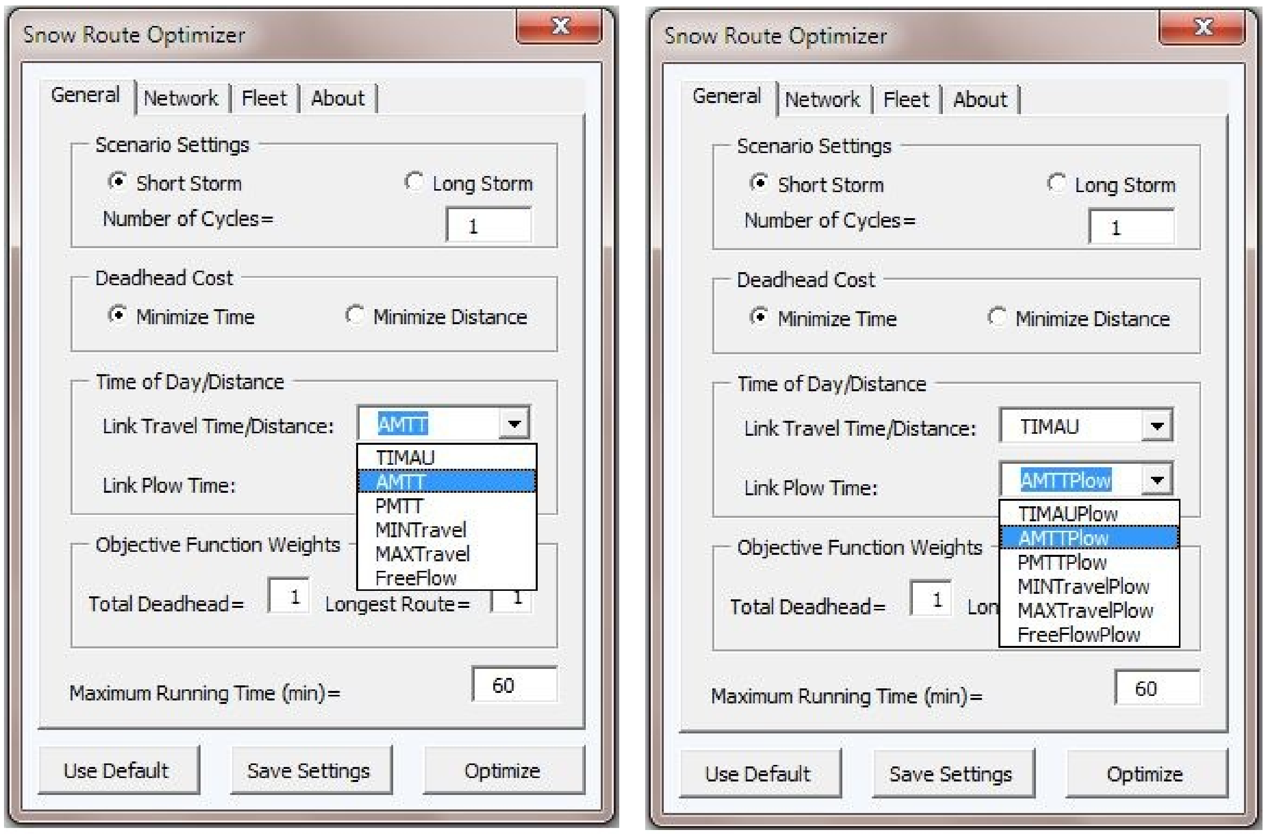


Figure A.2: General tab.

A.2.2 "Network" Tab

The "Network" tab allows users to specify the input data files for the points and links in the roadway network. This can be done by clicking the "Browse" button and choosing the "LCDOT_Data.mdb" file through the dialog box. The selected database file will be automatically converted into two text files ("LCDOTNetworkLinks.txt" and "LCDOTNetworkPoints.txt") suitable for the optimization program.

In the next frame, the users can specify (*i*) whether the number of passes includes right turn lane, and (*ii*) the rounding threshold to compute the integer number of passes. In addition, salt application rate can also be entered by the users. The users can also choose whether they would like the trucks in the optimized route to share roadway links or whether they would like each truck to cover its own links (Figure A.4).



(a) (b)

Figure A.3: (a) Link travel time/distance and (b) Link plow time.

A.2.3 "Fleet" Tab

Under the "Fleet" tab, the user can input the truck fleet size and composition. The users can make a list view to indicate the truck name (type), the number of each truck type in the fleet, each type of truck?s salt capacity, and their fuel efficiency (i.e., miles per gallon). The users also have the option to remove any row from the list view by selecting this row and clicking the "Remove" button. On the other hand, the users will have the option to read the fleet specifications from the fleet database (Figure A.5).

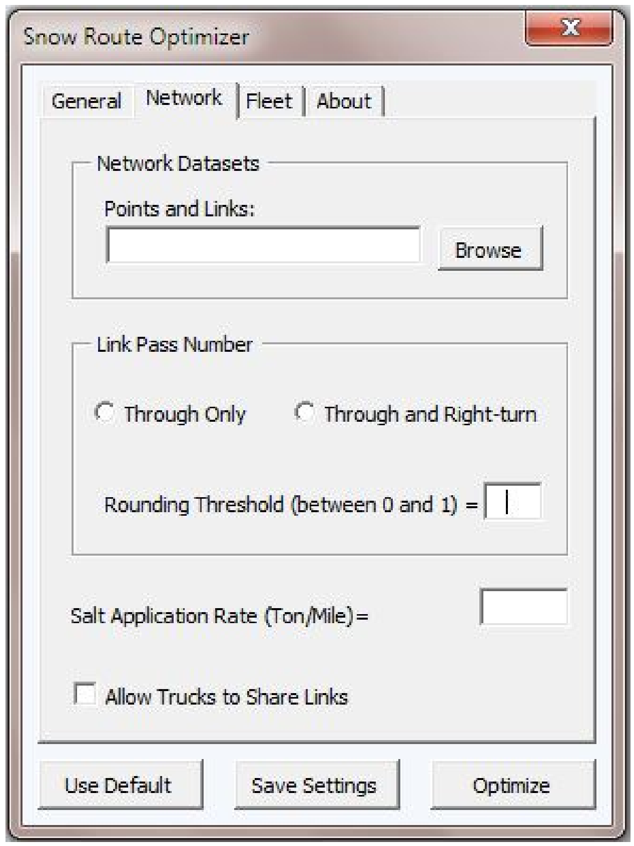


Figure A.4: Network tab.

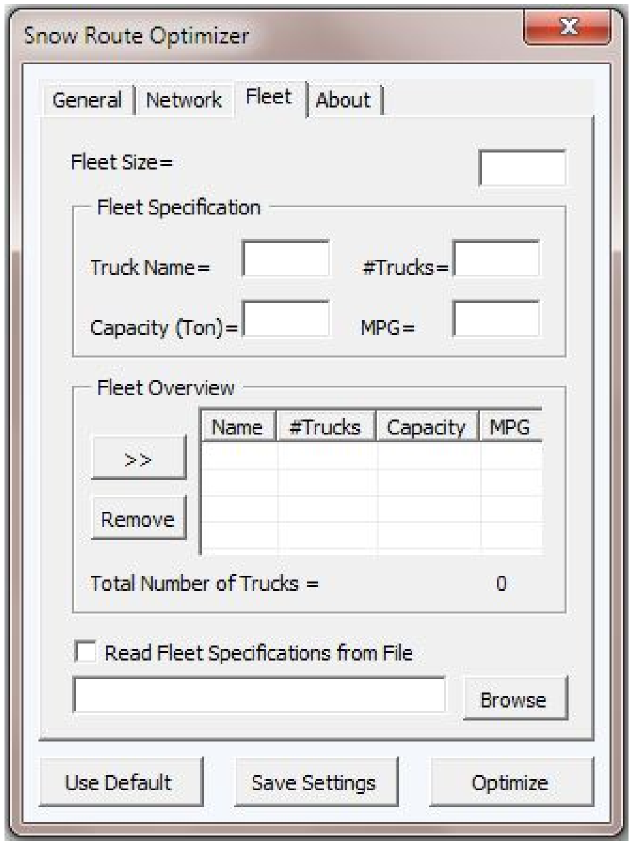


Figure A.5: Fleet tab.

A.2.4 "About" Tab

The "About" module displays acknowledgment and copyright information (version, release date, etc.), as shown in Figure A.6.

A.2.5 "Use Default" Button

The "Use Default" button allows the users to load the default settings that have already been stored in the system. Users can also change some values in the default setting and then save their settings (Table A.1). When the users click the "Use Default" button, the "Select Project Directory" window will be open; then, the users need to select the software folder (where all files are stored) to save the input files to the optimization program. Note that this window only allows the users to select a folder (not a file) (see Figure A.7).

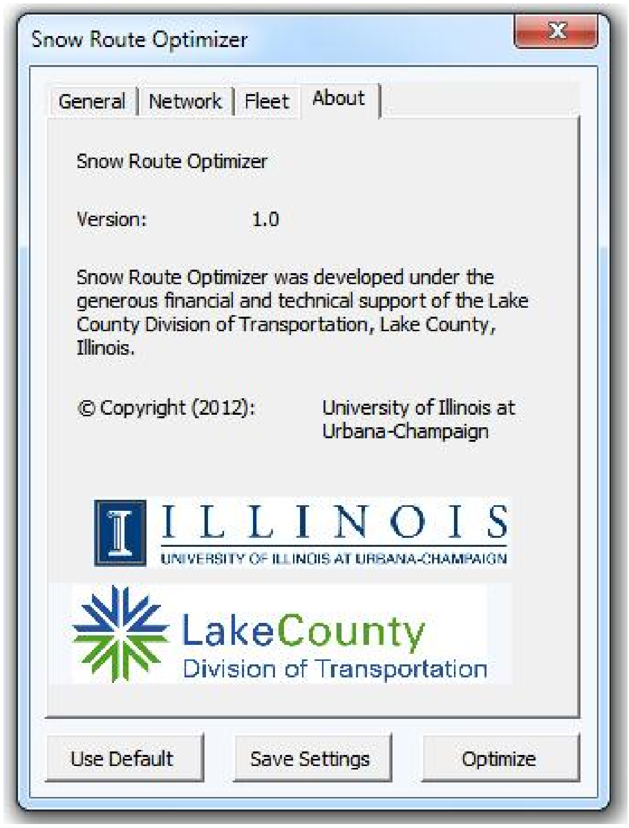


Figure A.6: About tab.

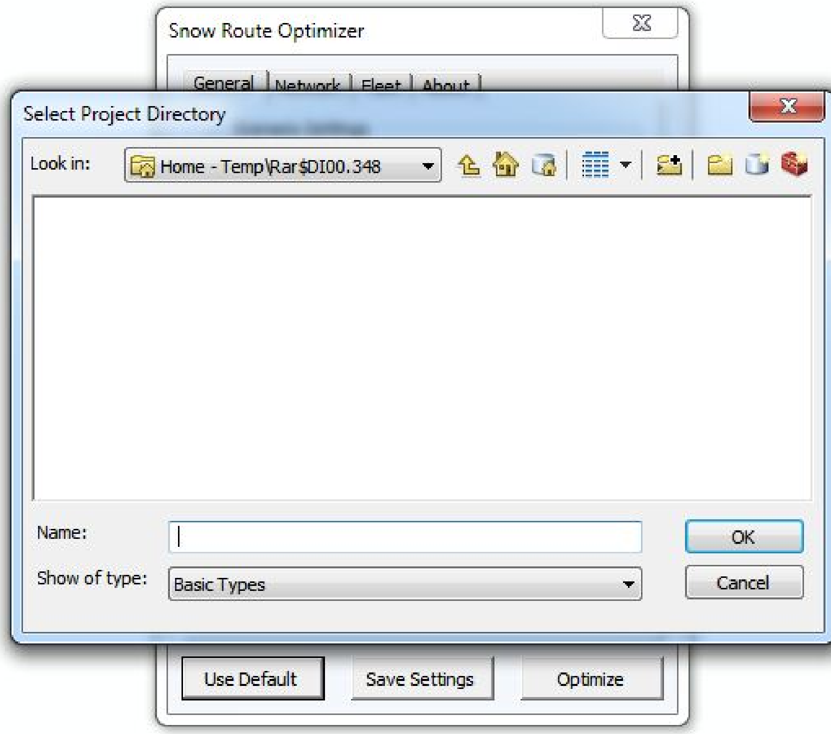


Figure A.7: Select Project Directory for Use Default Button.

A.2.6 "Save Settings" Button

After entering input parameters, the user needs to "Save Settings" before clicking the optimize button. This will help save all the user inputs in text files, which the optimization program will read as input files. Similar to the "Use Default" button, when the users click the "Save Settings" button, the "Select Project Directory" window will be open and the users should select the software folder to save the input files.

If the time-of-day that the users have chosen for travel and plow times are different, a dialog box will pop up to remind the users about this discrepancy (see Figure A.8 (a)-(b)). However, the users may choose to continue with that setting. If the users select yes, the current setting will be saved; otherwise another message box will pop up to remind the users

Table A.1: The default parameter values.

Default Setting	Value
Storm Scenario	Short Storm
Deadhead Cost Mode	Minimize Time
Link Travel Time	AMTT
Link Plow Time	AMTTPlow
Total Deadhead	1
Longest Route	1
Maximum Running Time (min)	60
Right-turn (min)	0.1
Left-turn (min)	0.5
U-turn (min)	1.5
Link Pass Number	Through with Right-turn
Rounding Threshold	0.3
Salt Application Rate (Ton/Mile)	0.125
Allow Trucks to Share Links	No
Fleet Size	25
Truck Name	A, B, C, D
# of Trucks	6, 7, 8, 4
Truck Capacity (Ton)	8, 7, 7.5, 11
Truck MPG	10, 12, 11, 9

to reset the time-of-day for travel and plow times.

It should be noted that this only happens when the users select "Minimize Time" under the "Deadhead Cost" portion; for the "Minimize Distance" mode, the plow time-of-day and Link "Travel Time/Distance" are obviously different and those dialog boxes will not interrupt saving the user settings.

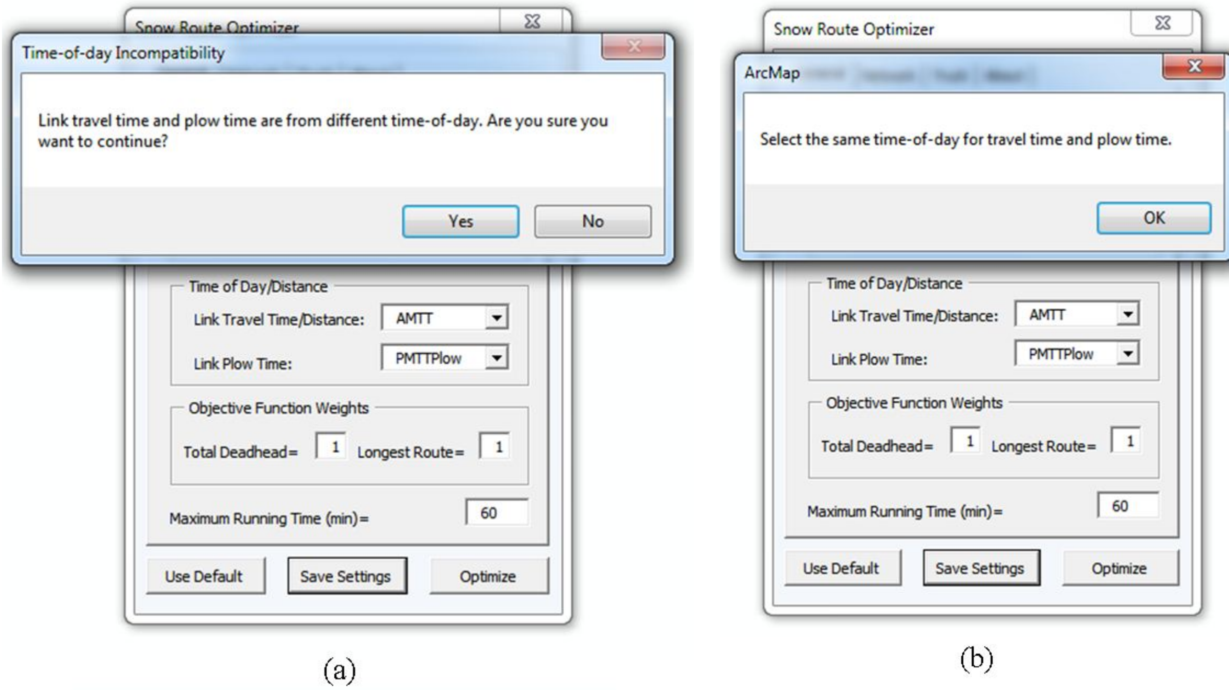


Figure A.8: Save Setting dialog boxes when different Travel and Plow times of day are selected.

A.2.7 "Optimize" Button

After saving the settings, the users can click on the "Optimize" button to run the optimization program. If the users click the "Optimize" button when some required information is missing in the "Setting" user form, the software alerts the users about the missing information (see Figure A.9).

If there are errors in the input network data, an error message will pop up. Then, the users need to go through the database and modify the data and update the "LCDOT_Data.mdb" file. Progress of the optimization will be shown in a DOS window at selected major steps of the optimization algorithm (Figure A.10).

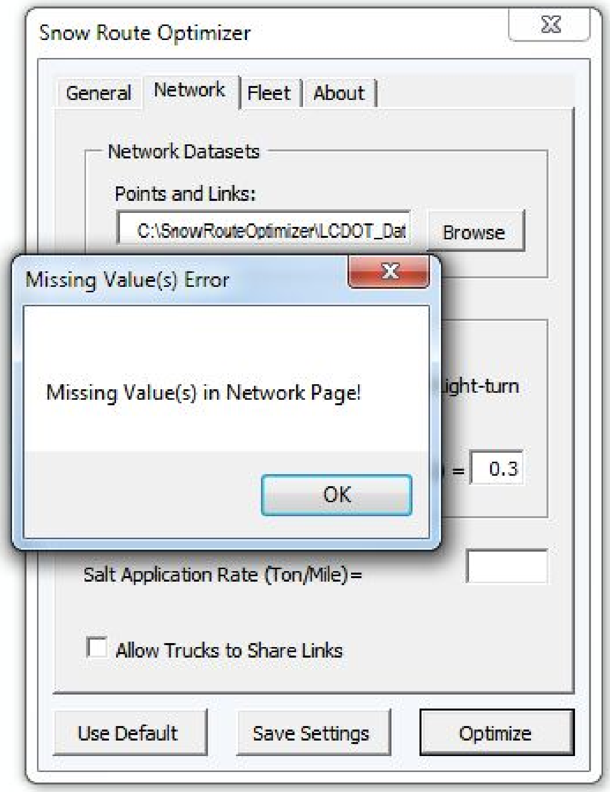


Figure A.9: Missing Value(s) Error: Salt Application Rate is missing.

A.3 Snow Route Optimizer Results

After running the optimization program, the "Snow Route Optimizer Results" form will appear (Figure A.11).

A.3.1 "Display Routes" Button

If the users click on "Display Routes" button, under solution overview, all the optimal truck routes will appear on the map with different colors (Figure A.12).

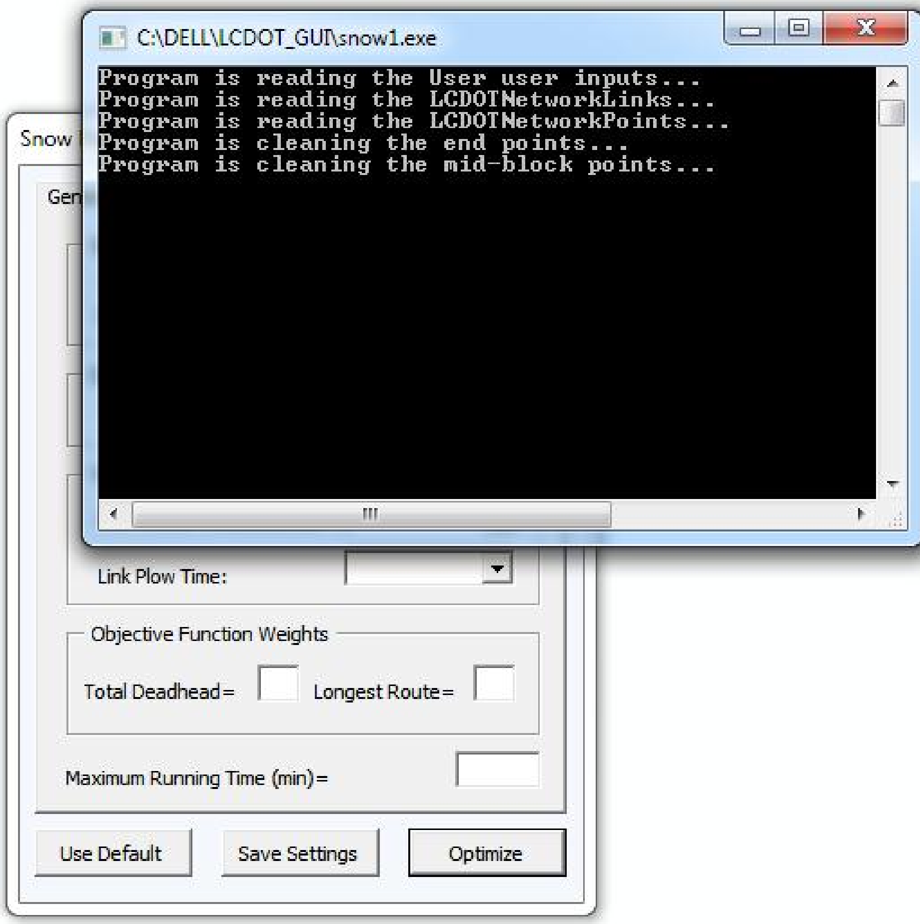


Figure A.10: Optimize button and progress window.

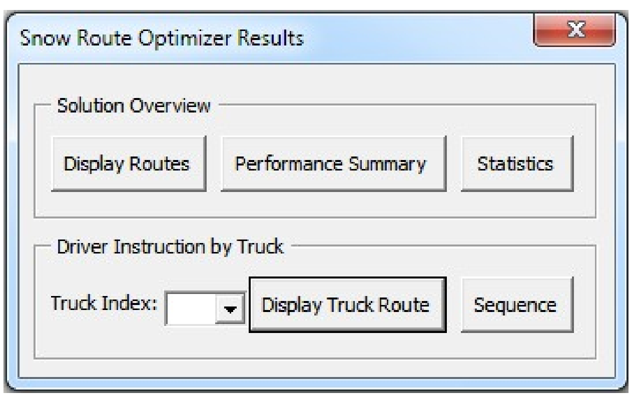
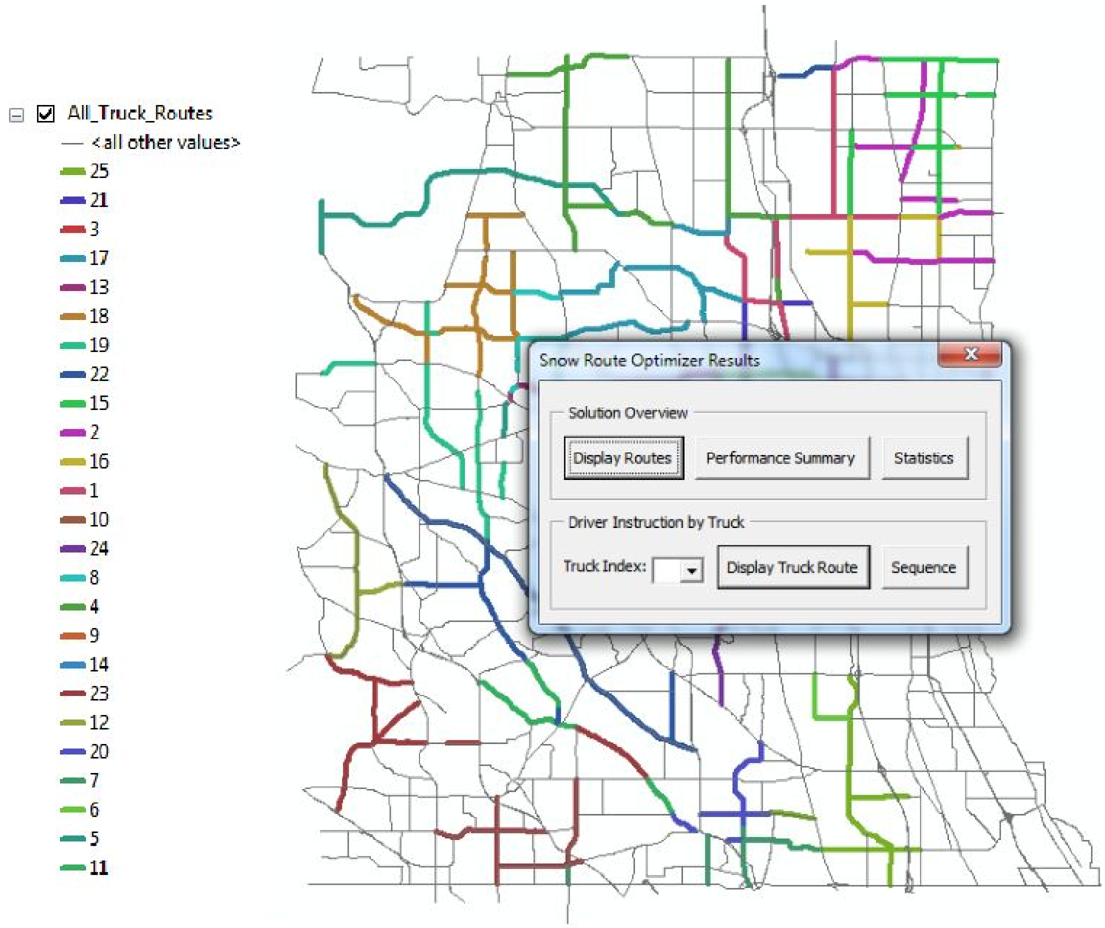


Figure A.11: Results window.



(a)

Figure A.12: Display all truck routes.

A.3.2 "Performance Summary" Button

If the users click on the "Performance Summary" button, the routing performance measures will appear in a table. This table indicates the total cycle time, plow time, deadhead time, estimated fuel consumption, etc. of each truck (Figure A.13).

A.3.3 "Statistics" Button

If the users click on the "Statistics" button, the routing statistics will appear in a table.

This table includes the mean and standard deviation of plow time, deadhead time, and total time across all the trucks in the fleet (Figure A.14).

A.3.4 "Display Truck Route" Button

The user can select a truck index and click the "Display Truck Route" button to check the detailed information of that truck?s route sequence. The selected truck?s route will be shown on the map (Figure A.15).

A.3.5 "Sequence" Button

The users can select a truck index and click the "Sequence" button to animate the truck?s roadway links in detail.

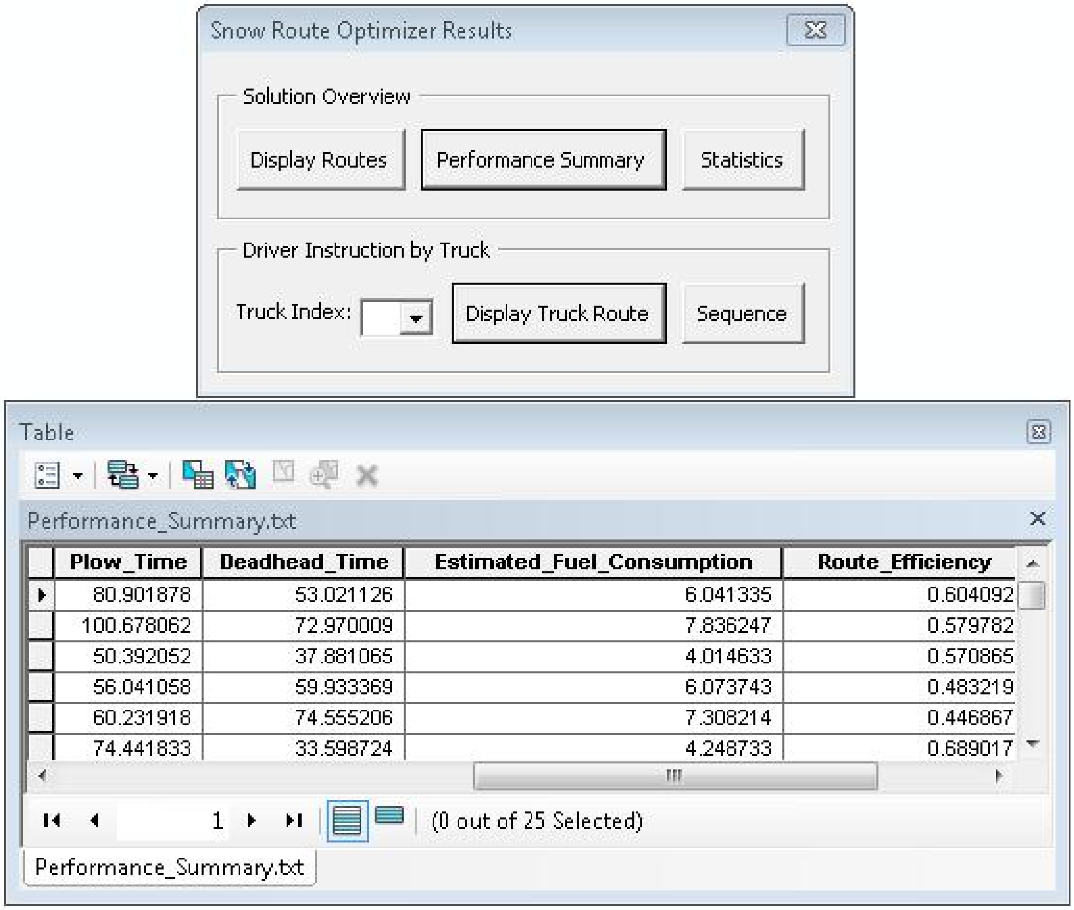


Figure A.13: Performance summary for each truck.

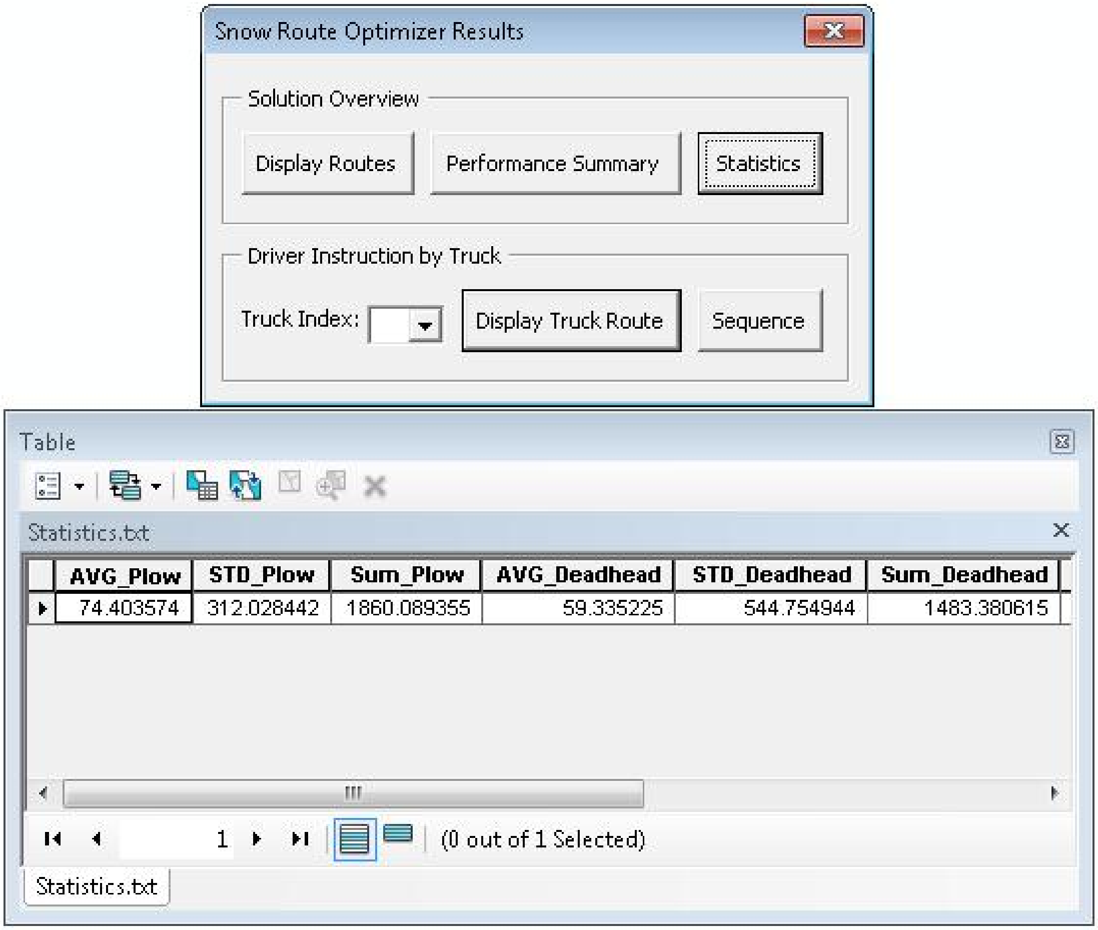


Figure A.14: Statistics for all trucks.

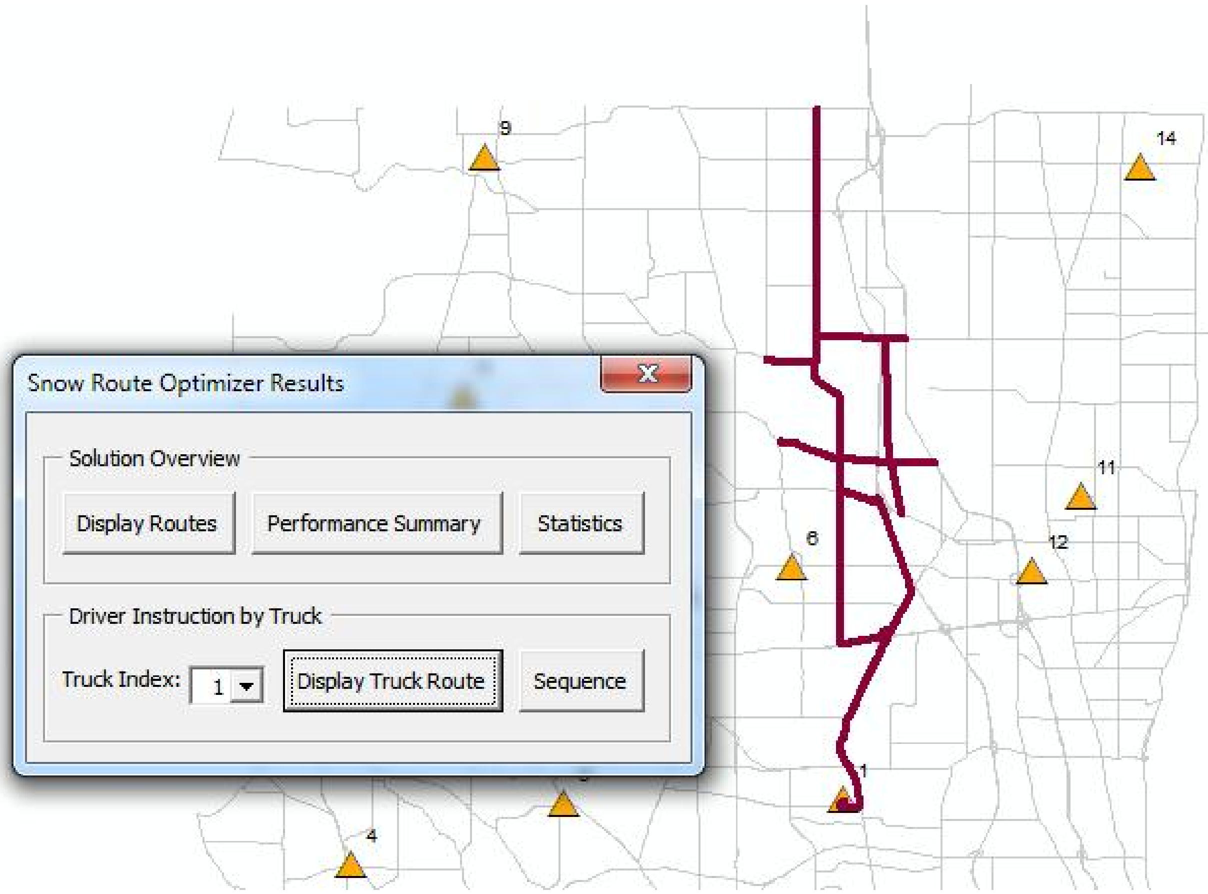


Figure A.15: Route sequence for each truck.

A.4 Software Update: Flexible User Setting Interface

The software can either start optimizing the snow routes from LCDOT's manual clusters (as an initial solution for the optimization) or optimize the routes from automatic clustering method embedded in the software. To provide the flexibility of choosing the clustering method for the users, the updated user setting interface includes "Optimization Method" with a check box to take into account the user preference by either starting the optimization from the manual clusters or making the optimization completely automatic (see Figure A.16).

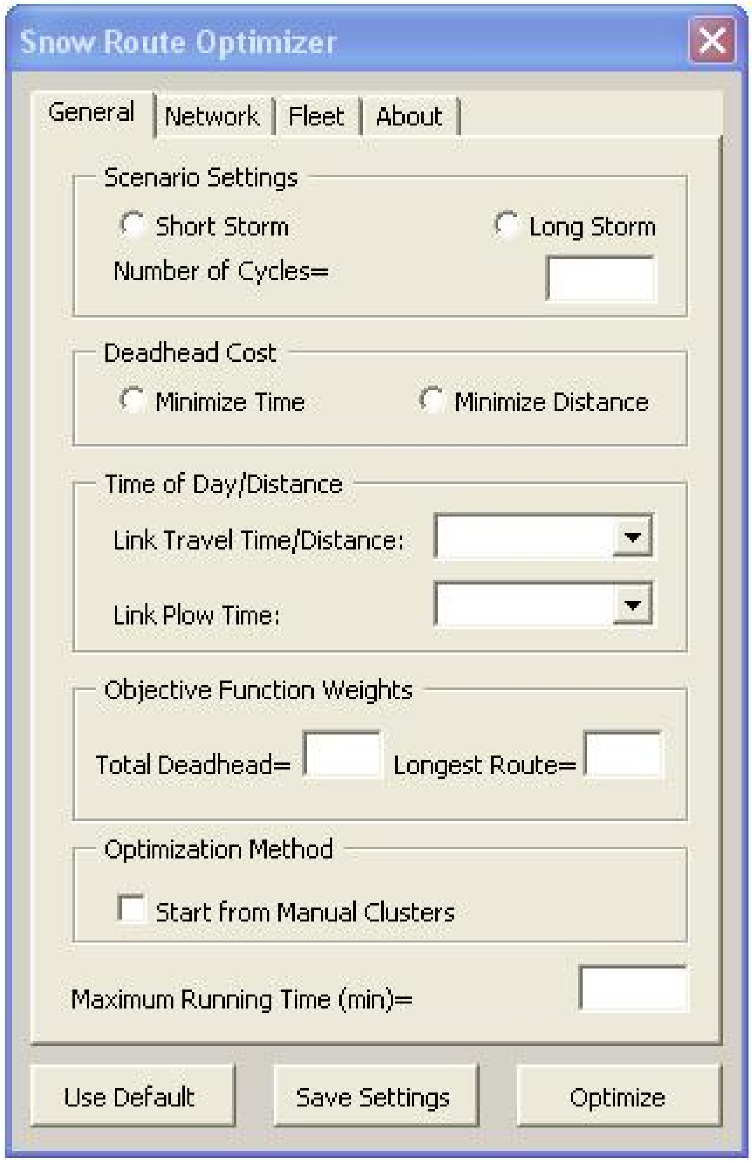


Figure A.16: Updated user setting interface.

If the users prefer not to check the box near "Start from Manual Clusters", the optimization program creates truck clusters based on 2 and then solves the network routing according to 2.

However, when the users check the box near "Start from Manual Clusters" , the optimization program reads the "LCDOT.txt" file provided in the software folder, which includes the manual clusters from LCDOT, and solves the network routing based on 2. Figure A.17 shows part of the "LCDOT.txt" file.

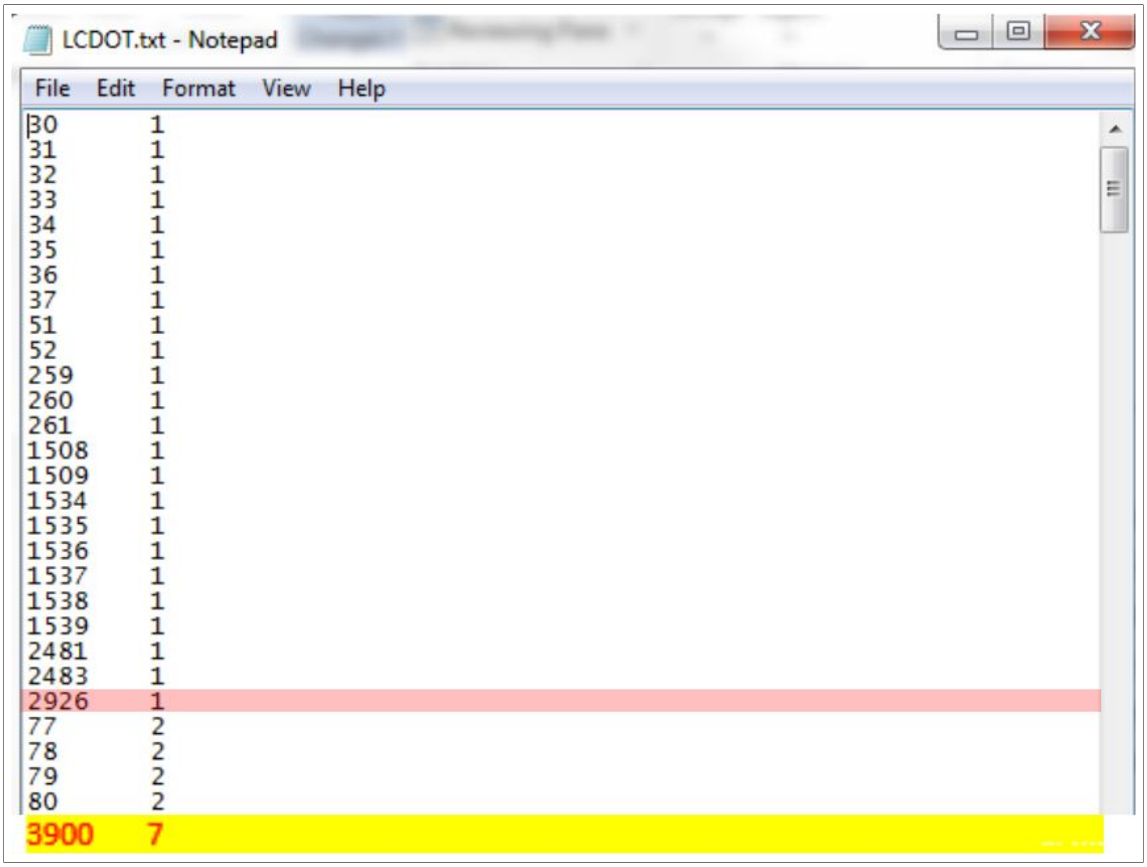


Figure A.17: Updating LCDOT manual solution.

This file contains two columns; the first column includes the OBJECTID of the task links and the second one shows the truck ID's that have been assigned to the task links based on LCDOT's clustering solution. The truck ID's range from 1 to the number of trucks (e.g., they range from 1 to 25 if 25 trucks have been set in the user setting interface).

A.4.1 Updating Manual Clusters in Case of Network Updates

If any changes occurs in the current LCDOT network (i.e., a set of task links is added to/removed from the network), the corresponding OBJECTID's need to be added/removed in the first column of "LCDOT.txt". Similarly, the truck ID's that have been assigned to the task links are required to be updated in the second column. The procedure of updating the "LCDOT.txt" file is provided as follows:

Step 1. Update the OBJECTID's of the task links

- Add/remove the OBJECTID's in the first column; for example, if task OBJECTID=2926 does not exist in the updated network, we simply remove the row related to this task OBJECTID (see the high-lighted row in pink in Figure A.17). Or, for instance, if a task link with OBJECTID=3900 is added to the new network, we just add this OBJECTID and the corresponding truck ID as a new row in this file (see the last row in Figure A.17, high-lighted in yellow).

Step 2. Update the truck ID's

- In the second column, double check the corresponding truck ID's that have been assigned to the updated task links based on LCDOT's updated solution.
- Truck ID's should range from 1 to the truck numbers; for example, if there were only 7 trucks for the task assignments, we should update the second column by ID's between 1 to 7.