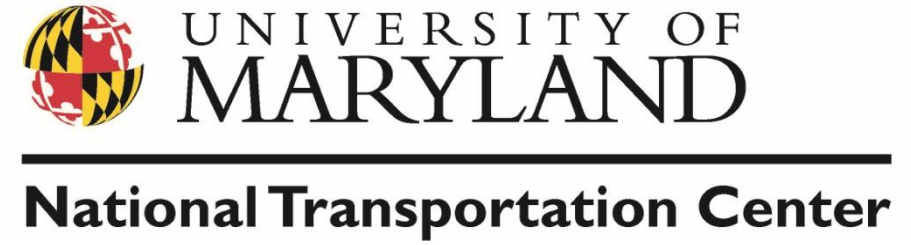

Project ID: NTC2015-MU-R-10

IMPROVING THE RELIABILITY OF FREIGHT TRANSPORTATION

Final Report

by

George F. List
gflist@ncsu.edu
919-515-8038
North Carolina State University
909 Capability Drive, Suite 3600
Research Building IV
Raleigh, North Carolina 27606

and

Elizabeth Byrom
Jeremy Addison
Atefeh Morsali
North Carolina State University

for

National Transportation Center at Maryland (NTC@Maryland)
1124 Glenn Martin Hall
University of Maryland
College Park, MD 20742

June 2018

ACKNOWLEDGEMENTS

The research team is deeply appreciative of the support received from colleagues Bo Yang, Mehdi Mashayekhi, Yu Lu, and Isaac Isukapati who helped prepare materials upon which the report is based. The team is also thankful for the collaboration with Drs. Paul Schonfeld at UMD who helped shape and focus the effort. The project was funded by the National Transportation Center @ Maryland (NTC@Maryland), one of the five National Centers that were selected in this nationwide competition, by the Office of the Assistant Secretary for Research and Technology (OST-R), U.S. Department of Transportation (US DOT).

DISCLAIMER

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

EXECUTIVE SUMMARY

This research report is focused on applying advanced tools and techniques to the analysis of reliability and efficiency for freight transportation. It deals primarily with truck-related pickup and delivery activities in urban areas, but the tools are applicable to other modes and multi-modal systems. The topic is important because of the economic value that results from minimizing the resource consumption associated with freight activity.

As is well recognized, a local carrier's objective is to accomplish pickups and deliveries within on-time windows (OTWs) a very high percentage of the time. Carriers measure their performance based on arrival, loading, and departure events, since all three relate to the perceived service quality by the customers, competitors, and other stakeholders, including society at large. These events are deemed to be "on-time", and therefore reliably executed, if they occur within specified, acceptable windows.

In the context of this research work, the implication is to find assignments of service requests (for pickup and/or delivery) to trucks and then service sequences and network paths (from one service request to another) that maximize on-time performance at minimal cost (or other impact). These two objectives are not necessarily co-linear, so a tradeoff may be involved. The challenge is to find vehicle assignment and routing solutions that achieve the best balance between these two objectives.

Without loss of generality, the network being examined can be assumed linear, as depicted in Figure ES-1. The servicing requests have both a location and a time. The challenge is to determine 1) how many trucks are needed and 2) what requests should be assigned to each truck. In the figure, 17 requests are shown, with locations and time windows indicated by the dark orange ovals. The length of the oval indicates the duration of the anticipated servicing window. The green box on the left indicates the earliest time when the trucks can leave the depot while the green box on the right shows the latest time they can return. (And hence necessitates the service time adjustments for R-01 and R-14.) In the solution shown, some tasks, like R-06 are accommodated when requested. (In fact, the truck arrives early and has schedule slack following.) In other cases, the servicing is either delayed, as in R-01, or done early, as in R-14. The solution shown uses five trucks, all of which depart from and return to the depot. There is no claim that this solution is either efficient or optimal. It is not likely to be either. But it is feasible; that is, all requests are accommodated. Obviously, it is not optimal because the servicing of four tasks is delayed (R-01, R-09, R-12, and R-17) and it is done early for two tasks (R-05 and R-14). It could be that a different assignment of requests to trucks could eliminate the adjustments to R-03, R-09, R-12, and R-17. It is not possible to fix the displacement of R-01 and R-14. The repositioning of those tasks is caused by consideration of how long it takes to reach those locations from the depot.

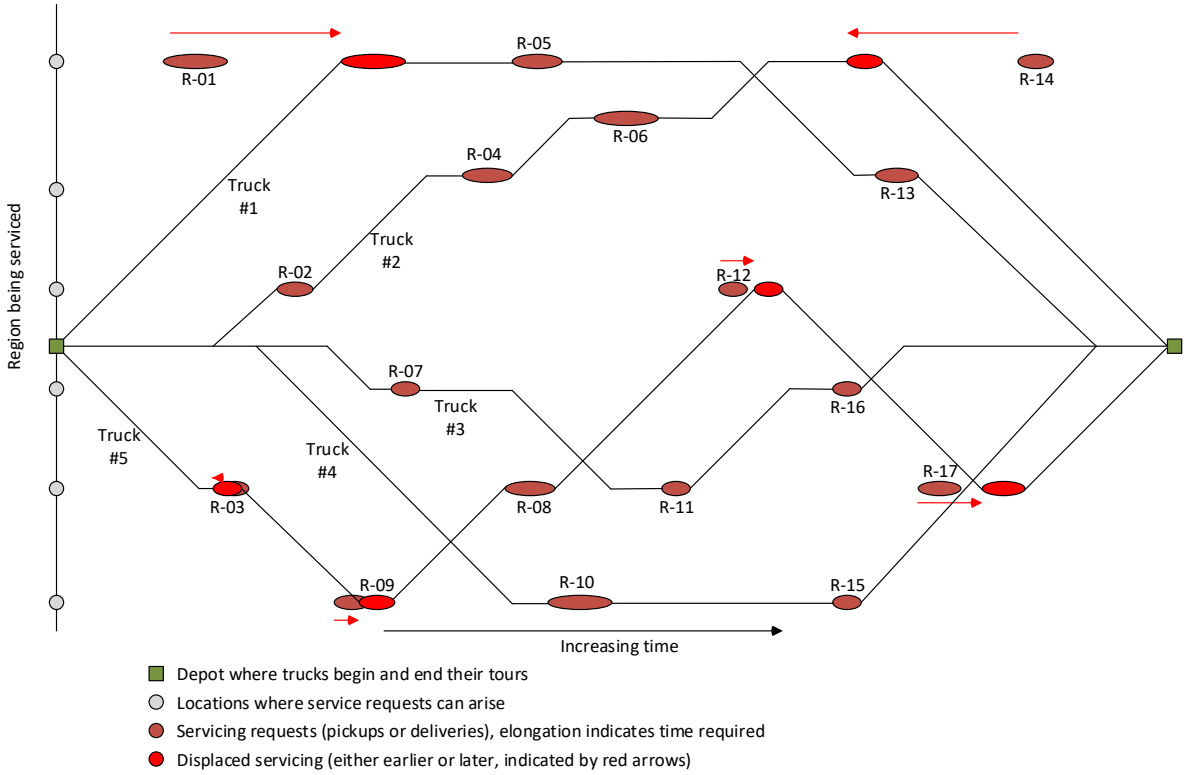

Figure ES.1: An example network with servicing requests and truck routings

Figure ES-1 makes the problem look relatively "simple", and in a sense, it is, if the travel times and service times are deterministic. Although the problem is still difficult to solve optimally, good heuristics exist to solve these problems in a deterministic setting. Many of these methods are described in the literature review.

When the travel times and service times are stochastic, however, the problem is more complex. No one assignment of the requests to the trucks is optimal. It depends on the values pertaining to a given problem realization.

Hence, when stochasticity is introduced, an optimal solution to the VRP does not exist. Rather, the knowledge contributions are: 1) how does the fleet size affect the robustness of the solutions obtained, and 2) are there request assignment patterns that occur frequently that constitute "thumb rules" for dispatching?

Since these problems are very difficult to solve analytically, a genetic algorithm-based method has been developed to search for increasingly better solutions; and that search procedure uses a truck fleet routing heuristic to develop implementable solutions. Effectively, the procedure synthesizes assignment solutions to test and the routing heuristic quantifies the performance of that assignment. Good assignments are kept. Poor ones are discarded. And new ones are synthesized by gene fusion of good solutions and gene mutations of good solutions.

The procedure works well for case studies that have been examined. It produces solutions that are reasonable and implementable, and whose logistics (what trucks are where, when, and doing what) is easily understood. The procedure has promise of being a technique that carriers could use to identify robust assessments of the fleet size needed to accommodate different ranges of service demands.

The report is organized as follows. Section 1 provides an overview of the problem. Section 2 reviews the pertinent literature. Section 3 presents a solution methodology based on block building, the strategy used by transit agencies to assign buses to blocks of route services. Section 4 presents a second method that develops solutions using a genetic algorithm-based method. Section 5 repurposes the genetic algorithm-based method to identify the best overall service request assignment given a specific set of requests and a fleet size. Finally, Section 6 summarizes the effort and identifies opportunities for future work.

1.0 INTRODUCTION

This research is focused on applying advanced tools and techniques to the analysis of reliability and efficiency for freight transportation. It deals primarily with truck-related pickup and delivery activities in urban areas although the tools are applicable to other modes and multi-modal systems. The topic is important because of the economic value that results from minimizing the resource consumption associated with freight activity.

Finding effective solutions to the vehicle assignment and routing problem in stochastic settings is very challenging. It is not only complex, but unique solutions do not exist. The solution is best when dependent on the operating conditions existent in the network at the time the decisions are made (e.g., the loads to be picked up and delivered, the fleet available, and the operating conditions of the network, as affected by weather, incidents, work zones, planned events, and other external factors).

As is well recognized, a carrier's objective is to accomplish the pickups and deliveries within on-time windows (OTWs) a very high percentage of the time. Carriers measure their performance based on arrival, loading, and departure events, since all three relate to perceived service quality by the customers, competitors, and other stakeholders, including society at large. These events are deemed to be "on-time", and therefore reliably executed, if they occur within specified, acceptable windows.

In the context of this research work, the objective is to find assignments of service requests (for pickup and/or delivery) to trucks and then service sequences and network paths (from one service request to another) that maximize on-time performance at minimal cost (or other impact). These two objectives are not necessarily co-linear, so a tradeoff can be involved, and the challenge is to find the vehicle assignment and routing problem solution that achieves the best balance between these two objectives.

1.1 DESCRIPTION

Without loss of generality, the network can be treated as linear, as depicted in Figure 1-1. The servicing requests have both a location and a time. The challenge is to determine 1) how many trucks are needed and 2) what requests should be assigned to each truck. In the figure, 17 requests are shown, with locations and time windows indicated by the dark orange ovals. The length of the oval indicates the duration of the anticipated servicing window. The green box on the left indicates the earliest time when the trucks can leave the depot while the green box on the right shows the latest time they can return. (And these constraints necessitate the service time adjustments for R-01 and R-14.) In the solution shown, some tasks, like R-06 are accommodated when requested. In fact, in the case of R-06, the truck arrives early and has schedule slack following. In other cases, the servicing is either delayed, as in R-05, or done early, as in R-14. The solution that is shown uses five trucks, all of which depart from and return to the depot. There is no claim that this solution is either efficient or optimal. It is not likely to be either. But it is feasible; that is, all requests are accommodated. Obviously, in some sense, it is not optimal because the servicing of four tasks is delayed (R-01, R-09, R-12, and R-17) and it is done early for two tasks (R-05 and R-14). It could

be that a different assignment of requests to trucks could eliminate the adjustments to R-03, R-09, R-12, and R-17. It is not possible to fix the displacement of R-01 and R-14, those changes are caused by how long it takes to reach those locations from the depot. But, as a trivial example, more trucks would fix the adjustments to the other two.

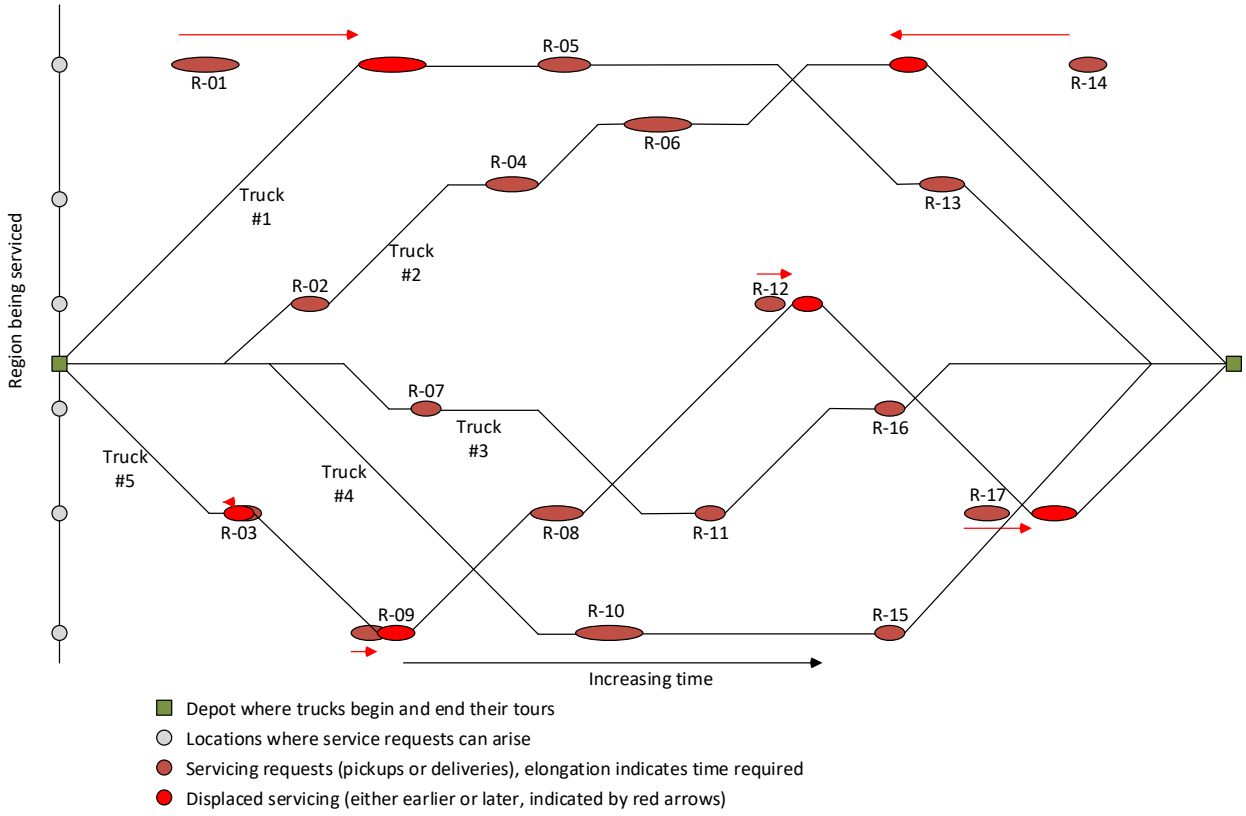

Figure 1.1: An example network with servicing requests and truck routings

When the travel times and service times are stochastic, however, the problem is more complex. No one assignment of the requests to the trucks is optimal. It depends on the values pertaining to a given problem realization. The main findings become: 1) how does the fleet size affect the robustness of the solutions obtained, and 2) are there request assignment patterns that occur frequently that constitute "thumb rules" for dispatching? Both issues are addressed.

Since these problems are very difficult to solve analytically, heuristics are often used. In this case, a genetic algorithm-based method (GAM) was developed. It synthesizes assignment solutions to test and the routing heuristic quantifies the performance of that assignment. Good assignments are kept, poor assignments are discarded, and new assignments are synthesized both by fusing the genes of good solutions and mutating the genes of those same good solutions.

The procedure works well for case studies that have been examined. It produces solutions that are reasonable and implementable. They are also solutions whose logistics (what trucks are where, when, and doing what) is easily understood. The procedure has promise of being a technique that

carriers could use to identify robust assessments of the fleet size needed to accommodate different ranges of service demands.

1.2 REPORT OVERVIEW

The remainder of the report is organized as follows. Section 2 reviews the pertinent literature. Section 3 presents a solution methodology based on block building, the strategy used by transit agencies to assign buses to blocks of route services. Section 4 presents a second method that develops solutions using a genetic algorithm-based method. Section 5 repurposes the genetic algorithm-based method to identify the best overall service request assignment given a specific set of requests and a fleet size. Finally, Section 6 summarizes the effort and identifies opportunities for future work.

2.0 PERTINENT RESEARCH

This section reviews prior research efforts that have focused on topics that are the same as or similar to the one addressed here. A prior project report, see List *et al*. (2017), provided a more comprehensive review of the literature focused on truck system reliability.

2.1 BASIC IDEAS ABOUT TRAVEL TIMES

Basic ideas about travel time reliability are presented in List *et al*. (2017). Trips are described as comprising 1) transport across links and 2) processing at nodes. Sometimes the processing is significant, as is the case of pickup and delivery times. At other times, it can simply be delay at junctions.

In Figure 2.1, the nodes are dots. They are surrounded by boxes. Each node has a letter designation (A through H). Connections between the nodes are shown as links. The word "link" means a two-way connection. Arc means one-way and implies directionality as in the arc DB, which originates at D and terminates at B.

When a shipment leaves a node, it passes through the box on the departing arc. It arrives at a node as it passes through the box on the arriving arc. Arc travel times arise between the boxes (on the arc). Processing times occur between the arriving and departing boxes at the node.

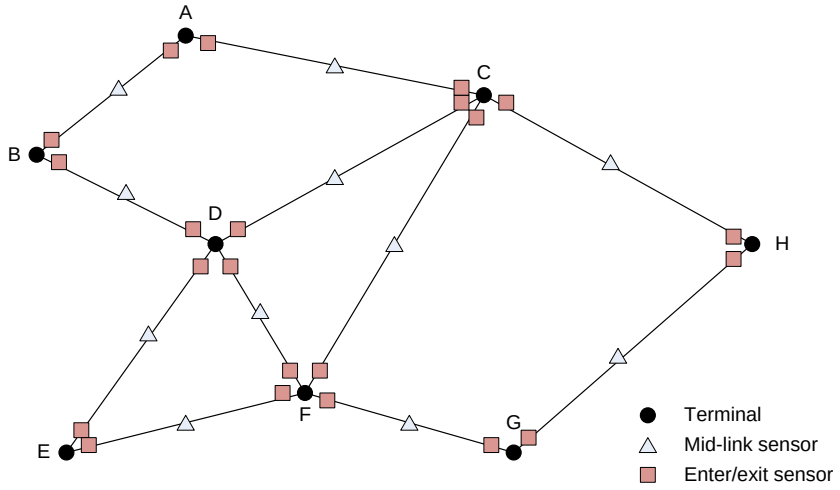

Figure 2.1: A hypothetical network and possible monitoring locations

For truck trips, there are arc transit times and nodal processing times. If the trip is from B to H with intermediate nodes D and C, there is an initial processing time at B, a travel time on arc BD, a processing time at D, a travel time on arc DC, a processing time at C, a travel time on arc CH, and a final processing time at H. The processing time at B is a loading time; the one at H is unloading; and the intermediate times are delays at junctions.

Insofar as the arcs are concerned, their travel rates (inverses of the space-based speeds) may vary temporally and spatially. The travel time from B to D on arc BD may be produced by travel rates

that vary by time and location. These rates vary because of changes in capacity, congestion, weather, incidents, maintenance work, etc. These "operating environment" variables determine the travel rates that are achieved. These variables describe the "operating condition."

At the nodes, the type of handling that occurs is important. Nodes that are just intersections or interchanges produce delays due to queueing and traffic control (e.g., signals or ramp metering). For example, if node D is a freeway interchange, then trucks see the time for traveling on a ramp from arc BD to arc DC If the node is an at-grade intersection (or an interchange that involves intersections), then the delay arises from making a turn, as in a left turn from arc BD to arc DC. Nodes where pickups or deliveries occur produce times that reflect the processing time associated with loading or unloading, and possibly re-arranging other shipments within the truck.

2.2 VEHICLE ROUTING AND SCHEDULING

The vehicle routing and scheduling problem (VRP or VRSP) is the main topic being addressed in this research. VRP is focused on assigning loads (service requests) to vehicles and then routing the vehicles so that performance metrics like total cost are optimized.

Three versions of the VRP are worth noting. In the first, the service requests involving taking complete truckloads from one place to another. The vehicles carry these loads in sequence, with intervening empty moves for repositioning. In the second case, the loads are either being picked up or delivered. The truck might leave the depot full and return empty (all deliveries) or leave empty and return full (all pickups). In some instances, both deliveries and pickups can occur within the same tour. These are the types of tours being studied in this research. In the third, loads can be both picked up and delivered during the tour, and multiple loads can be on-board at any given point in time as long as no single load consumes the vehicle's capacity. The objectives are often to 1) minimize total cost, 2) maximize on-time deliveries, 3) minimize the fleet size, and 4) maximize vehicle utilization. Other objectives include 5) maximizing on-time performance and 6) maximizing the lowest on-time performance among all the vehicles employed.

The body of literature on vehicle routing and scheduling is vast. Bodin *et al.* (1981) were the first to provide a review, and they identified over 500 papers. Our review will not endeavor to present all this literature in detail.

The first paper appears to be that of Dantzig and Ramser (1959). They present a formulation of the truck dispatching problem that assigns full truckloads to multiple trucks based on truck capacity. The motivation was the scheduling of tank trucks delivering gasoline to filling stations. No direct treatment is given to the distances traveled by the trucks. The loads are sorted into a specific order and then assigned to trucks sequentially given the truck capacities.

A subsequent paper by Clarke and Wright (1964) describes a savings-based heuristic that develops solutions to the VRP. They describe the problem as finding tours for *K* trucks such that 1) all loads are carried, 2) the total distance traveled by the trucks is minimized, and 3) the capacities of the trucks are not exceeded. The procedure begins by creating individual tours for each load *i*. The resulting objective function value is $z = \sum_i 2t_{di}$ where t_{di} is the time between the depot and load *i*.

The algorithm combines these tours by replacing $t_{id} + t_{dj}$ with t_{ij} based on the savings $s_{ij} = (t_{id} + t_{dj}) - t_{ij}$ starting with the largest savings and working downward. The resulting grand tour is split each time vehicle capacity is reached. The procedure continues until no more savings can be achieved. The fleet size is ascertained when the algorithm terminates.

As the VRP is a combinatorial programming problem, it falls in the class of NP-Hard problems; therefore, it is unlikely that there exists a polynomial-time algorithm to solve VRP to optimality. Exact algorithms use formulations based on integer programming, including branch and bound, set partitioning, column generation, and network flow solution methods or formulations from dynamic programming with effective state-space relaxation (see the survey paper by Laporte (1992)). These formulations struggle to accommodate large-size problems. Heuristics can deal with much larger size problems and obtain solutions quickly. However, understandably, the answers may be suboptimal. Many heuristics use a 2-phase route construction and route improvement algorithm. The route construction stage often uses the Clarke-Wright Savings algorithm (1964) or the sweep algorithm described by Wren and Holliday (1972). When the Clarke-Wright algorithm is employed and the distances are Euclidean, further reductions in total route cost can often be achieved by limiting the domain of eligible arcs using an upper bound on arc costs as was done in Caccetta, Alameen, and Abdul-Niby (2013). The sweep algorithm uses a fixed number of vehicles when building routes. There is no gaurantee that vehicle capacity will not be exceeded. The solutions in either case can be further improved by using techniques such as k-opt switches of customers, route merging and splitting, and subroute deletion and insertion.

Over time, the VRP has been embellished to include constraints focused on capacity, time windows, nonhomogeneous travel times, stochastic demands and travel times, and pickup and delivery precedences. For instance, there is a formulation that includes hard time windows. Each request has a restriction that service *i* must occur within specified early and late times. A feasible routing must meet all time constraints. On the other hand, with soft time windows, the solution allows for violation of these time windows with a penalty cost.

The literature on VRP can be broken down into deterministic and stochastic groupings. The first assumes the travel times and servicing times are fixed. The second allows them to be stochastic.

2.2.1 Deterministic Formulations

The deterministic VRP can be stated as follows. Assume there are *N* service requests and each consumes truck capacity based on dem_i. Also, let *K* vehicles be available and assume each one has a capacity of cap_k. Choosing to use vehicle *k* is reflected by $z_{0k} = 1$ and assigning load *i* to vehicle *k* is designated by z_{ik}. The sequence for visits to the destinations is captured by x_{ijk} which indicates that load *i* is to be delivered before load *j* by vehicle *k*. If the distance between *i* and *j* for vehicle *k* is given by c_{ijk}, then the problem is:

Minimize:

$$\sum_k \sum_i \sum_j c_{ijk} x_{ijk} \tag{2.1}$$

Subject to:

$$\sum_i dem_i * z_{ik} \le cap_k * z_{0k} \quad \forall\, k \tag{2.2}$$

$$\sum_k z_{0k} \le K \tag{2.3}$$

$$\sum_k z_{ik} = 1 \quad \forall\, i \tag{2.4}$$

$$\sum_i x_{ijk} = z_{jk} \quad \forall\, j,k \tag{2.5}$$

$$\sum_j x_{ijk} = 1 \quad \forall\, i,k \tag{2.6}$$

$$\sum_{i\in S}\sum_{j\in S} x_{ijk} \le |S| - 1 \quad \forall\, k \quad \textit{where} \quad S \subseteq N(z_{ik}) \text{ is the set of all deliveries made by } k \tag{2.7}$$

The objective function specifies that the total vehicle miles (the cost in this case) should be minimized. Equation (2.2) ensures that the capacity of each truck used is not exceeded. Equation (2.3) ensures the selected fleet size is not greater than the fleet available. Equation (2.4) ensures that the loads get assigned, Equations (2.5) and (2.6) establish the load assignment sequences and equation (2.7) ensures that the number of arcs traversed by each truck is less than or equal to the number of deliveries made.

The VRP has also been applied to other domains. Bodin *et al.* (1978) studied the routing and scheduling of street sweepers. Ronen (2002) described the use of VRP in the context of cargo ships. Zografos and Androutsopoulos (2002), Meng, Lee, and Cheu (2005), Ghoseiri, Ghannadpour, and Seifi (2010) examined the problem of dispatching railroad locomotives, and Androutsopoulos and Zografos (2012) examined the domain of hazardous materials transport.

Software has also been developed to help with solving VRP problems in practice. One that is well known is RUCUS (Run Cutting and Scheduling), developed for the scheduling of transit buses, hence the name. It is described by Nussbaum (1975), Field (1976), and Hinds (1979).

Attention has also been given to finding procedures that can solve very large VRP problems. Agin (1975) described a large number of algorithms. Buxey (1979) explored the possibility of using Monte Carlo simulation to find solutions. Baker and Rushinek (1982) examined large-scale implementation issues.

Additional formulations have been developed to address other situations. For example, if the service requests include pickups and deliveries that are to occur for specific packages while the truck is enroute, constraints are needed that ensure things like the pickup occurs before the delivery, and the capacity of the truck is not exceeded by the packages on-board. This problem is often called the “Dial a Ride” problem because it was developed to address the transit service needs of handicapped individuals. The problem is as follows. A set of requests are made for trips to and from specific locations with specific departure and arrival times, like doctor’s appointments and shopping trips. The task is to determine how to assign these trips to the dial-a-ride vehicles, and how many vehicles to use. The solution becomes the pick-up and delivery schedule. Unlike delivering loads, multiple people can be on-board the vehicle at any given point in time. Bruck

(1969) was one of the first to present a formulation. He also described a tool called CARS (Computer Aided Routing and Scheduling) that was intended to be a decision support system for solving the dial-a-ride problem. Papers with a similar focus include Howson (1970), Deleuw, Cather and Company (1971), Arthur D. Little (1971), Roos (1971), Roos and Porter (1971), Roos and Wilson (1971), and Fielding (1977).

Cordeau (2006) describes the problem as follows. Let n be the number of users (or requests) to be served. Sets P and D contain the pick-up and drop-off nodes, respectively and they are combined to create set N for which the the 1 to n nodes are pick-up nodes while those from $n + 1$ to $2n$ are drop-off nodes. A set of arcs A is a fully connected directional graph on N. Each vehicle $k \in K$ has a capacity Q_k and a maximum allowable tour duration T_k. Associated with each node $i \in N$ is a load q_i and a non-negative service duration d_i. A time window $[e_i, l_i]$ is also associated with node $i \in N$ where e_i and l_i represent the earliest and latest time, respectively, at which service may begin at node i. For each arc $(i, j) \in A$ there is a routing cost c_{ij} and a travel time t_{ij}. Finally, L is the maximum ride time allowed by policy for any user. For each arc $(i, j) \in A$ and each vehicle $k \in K$, $x_{ij}^k = 1$ if vehicle k travels from node i to node j. For each node $i \in N$ and each vehicle $k \in K$, let B_i^k be the time at which vehicle k begins service at node i, and Q_i^k be the load (number of people) on vehicle k after visiting node i. Finally, for each user i, let L_i^k be the ride time of user i on vehicle k. The formulation is as follows:

Minimize:

$$\sum_{k\in K}\sum_{i\in N}\sum_{j\in N} c_{ij}^k x_{ij}^k \tag{2.8}$$

Subject to:

$$\sum_{k\in K}\sum_{j\in N} x_{ij}^k = 1 \qquad \forall\, i \tag{2.9}$$

$$\sum_{j\in N} x_{ij}^k - \sum_{j\in N} x_{n+1,j}^k = 0 \qquad \forall\, i \in i,k \tag{2.10}$$

$$\sum_{k\in K}\sum_{j\in N} x_{0j}^k = 1 \qquad \forall\, k \tag{2.11}$$

$$\sum_{j\in N} x_{ji}^k - \sum_{j\in N} x_{ij}^k = 0 \qquad \forall\, i \in i,k \tag{2.12}$$

$$\sum_{i\in N} x_{i,2n+1}^k = 1 \qquad \forall\, k \tag{2.13}$$

$$B_j^k \geq (B_i^k + d_i + t_{ij}) x_{ij}^k \qquad \forall\, i, j, k \tag{2.14}$$

$$Q_j^k \geq (Q_i^k + q_j) x_{ij}^k \qquad \forall\, i, j, k \tag{2.15}$$

$$L_i^k = B_{n+i}^k - (B_i^k + d_i) \qquad \forall\, i, k \tag{2.16}$$

$$B_{2n+1}^k - B_0^k \leq T_k \qquad \forall\, k \tag{2.17}$$

$$e_i \le B_i^k \le l_i \qquad \forall i,k \tag{2.18}$$

$$t_{i,n+i} \le L_i^k \le L_i \qquad \forall i,k \tag{2.19}$$

$$\max\{0, q_i\} \le Q_i^k \le \min\{Q_k, Q_k + q_i\} \qquad \forall i,k \tag{2.20}$$

$$x_{ij}^k \in \{0,1\} \qquad \forall i, j, k \tag{2.21}$$

The objective function (2.8) minimizes the total routing cost. Constraints (2.9) and (2.10) ensure that each request is served exactly once and that the origin and destination nodes are visited by the same vehicle. Constraints (2.11) - (2.13) guarantee that the route of each vehicle *k* starts at the origin depot and ends at the destination depot. Consistency between the time and load variables is ensured by constraints (2.14) and (2.15). Equalities (2.16) define the ride time of each user which is bounded by constraints (2.19). The latter also act as precedence constraints because the non-negativity of the L_i^k variables ensures that node *i* will be visited before node *n* + *i* for every user *i*. Finally, the inequality (2.17) bounds the duration of each route while constraints (2.18) and (2.20) impose time windows and capacity constraints, respectively. This formulation is non-linear because of constraints (2.14) and (2.15) but there are ways to convert it to a mixed integer LP.

2.2.2 Stochastic Formulations

The second subgroup of research assumes the travel times and servicing times are stochastic. Techniques like stochastic optimization and simulation in combination with optimization (search routines) are used to find solutions. This work is more recent, spawned by the advent of computers that can simulate the movement of large fleets of trucks in reasonable time.

In Stochastic Vehicle Routing Problems (SVRPs), the expected travel times and service times are used as surrogates for the more realistic stochastic variability arguing that a "expected value" problem is being solved. This is expedient, but the resulting solutions provide little or no insight about system performance when the vector of values is significantly different from these mean values.

Treatment of the problem from a stochastic standpoint starts about 1990. Laporte *et al.* (1992) addressed the problem of finding solutions to the vehicle routing and scheduling problem when stochastic travel times are present. A chance-constrained programming formulation is presented along with two stochastic optimization formulations and a branch-and-cut algorithm for solving all three formulations. The chance constrained formulation performs well as should be expected since it is a variant on the mixed integer LP formulation. Of the two stochastic optimization formulations, the one that more explicitly represents the problem formulation does much better. The authors conclude that such problems can be solved for significant size problems in reasonable time. Powell (1988) described algorithms that can be used to solve the dynamic (time-based) routing of vehicles in response to known and anticipated, but unknown loads.

Many papers focused on solving stochastic vehicle routing problems followed Laporte *et al.* (1992). There is the notable paper by Bertsimas *et al.* (1995) and the proceedings paper by Taniguchi *et al.* (1999). Campbell (2004) described heuristics for considering a variety of

complicating constraints not typically included in the traditional formulations. Yamada, Yoshimura, and Mori (2004) is an interesting paper because it endeavors to use VRP procedures to study and assess road network reliability.

Other papers focused on making decisions about re-routing vehicles in real-time in response to evolving network conditions. Taniguchi, Yamada, and Tamaishi (2001) presented a formulation of the problem. Taniguchi and Nakanishi (2003) gave another. Slater (2002) provided an approach to the problem as does Kim (2003). Dessouky, Ioannou, and Jula (2004) examined strategies that can build partial tours (in time) and then update those tours as more information becomes available. Mitrovic-Minic and Laporte (2004) explored the use of waiting strategies. Hejazi and Haghani (2009) investigated ways for less-than-truckload services to optimize their services considering evolving conditions in the highway network. Kanturska, Trozzi, and Bell (2013) presented the idea of hyperpaths to help drivers select optimal delivery routes and schedules in response to evolving patterns of network travel times. The hyperpaths are sets of possible paths plus a path selection logic.

The advent of optimization schemes such as genetic algorithms, simulated annealing, and tabu search has motivated explorations of ways to use these techniques to solve VRP problems. The earliest investigation appears to be Garcia and Arunapuram (1993) who explored the use of tabu search. Potvin (2007) provided a survey of evolutionary algorithms that have been applied to VRP. Included in the review are genetic algorithms, evolutionary strategies, and swarm optimization. Weise, Podlich, and Gorldt (2009) completed a similar, newer review.

Subsequent research efforts focused on using a variety of techniques to address specific problems. Cordeau and Laporte (2003) used tabu search to solve a multi-vehicle dial-a-ride problem that is dynamic. Bell (2004) explored the use of game theory to address VRPs where the occurrence of incidents is of concern. Vidal *et al.* (2012) presented a hybrid genetic algorithm for solving multi-depot and periodic VRPs. Lin, Yu, and Chou (2011) employed simulated annealing and Xu, He, and Li (2009) explored a hybrid procedure that integrates genetic algorithms, stochastic simulation, and neural networks.

3.0 BUILDING BLOCK METHOD

This research explores three methods for developing VRP solutions to the stochastic problem. The first one, described here, is the Block Building Method, or BBM. It was originally presented in List *et al*. (2017). The BBM employs a vehicle assignment heuristic structured much like RUCUS, described in Section 2.

3.1 METHODOLOGY

The BBM makes a single pass through the set of requests, in chronological order, and identifies an assignment of trucks which is both feasible and efficient. It is feasible in that the chronology of events for every truck ensures that the servicing requests are accommodated and each truck has sufficient time to travel from one servicing request to the next. It is efficient in that for every request the truck that is nearest for accommodating that request is the one selected. The BBM is based on List *et al.* (2003) and List and Turnquist (1993). Those formulations, in turn, are based on Goeddel (1975) as enhanced by Ball, Bodin, and Greenberg (1985) and described by Wren and Rousseau (1995).

The problem formulation used by BBM is as follows, linguistically:

Maximize the on-time performance for the services provided
Minimize the total cost

Subject to:

Completing all pickups or deliveries (one or the other)
Not exceeding the capacity of any vehicle
Not exceeding the maximum tour duration allowed for any vehicle

Find this solution while working with the following assumptions:

- The fleet size is fixed.
- The vehicles are always available (maintenance spares exist).
- There is a single depot where trucks originate and terminate their tours.
- Every servicing has an arrival window (AW) at the beginning and a departure window (DW) at the end. The DW starts at the same time as the AW, but it ends at a time specified by the customer. A servicing is "on-time" if both the arrival and the departure fall within these windows.
- Both pick-ups and deliveries can be made by the same vehicle during a given tour. But delivered shipments must originate at the depot and picked-up shipments must be carried to the depot. Said another way, the depot must be one end of the trip for each shipment.
- Trucks have a limited capacity and each shipment uses some of that capacity.
- The total time for each truck tour is the sum of the random variables that describe the travel times between stops and the times spent in pick-up or delivery. This means the travel times are stochastic as are the pick-up and delivery times.

- An upper limit exists for the duration of any tour (measured in minutes).
- If trucks arrive earlier than the AW, then they wait until the beginning of the AW to start loading or unloading.
- Trucks can depart as soon as they are finished with the loading/unloading task; they incur no penalty for doing so.
- The cost equation has five components: a) the number of trucks used, b) truck hours, c) truck miles, d) the penalty cost for arriving early, and e) the cost for departing late.
- On-time performance for arrivals is measured by the probability of arriving within the AW. The same pertains to the probability of departing during the DW.
- The probability of an OTA can be improved by adding slack time to the schedule; that is, by arriving early, off-site, near a customer's location, at the cost of adding time to the tour and potentially increasing the fleet size.
- The probability of an OTA can also be improved by increasing the fleet size. A larger fleet reduces the number of customer stops per truck and adds more resource availability.
- All shipments are accommodated, either directly, or by outsourcing (at a significant cost).
- Trucks are interchangeable.
- Drivers are always available.
- A load and a shipment are the same thing. The two words can be used interchangeably. The same is true for the words customers, consignees, and stops.

The AWs indicate when the loading or unloading dock is available for use. A truck is "early" if it arrives before the AW begins. It is "late" if it arrives after the AW ends. It is "delayed" if it leaves after the DW ends. When the truck arrives early, it waits until the beginning of the AW. There are costs for being early or delayed, but not late. (This could be changed. The lateness is monitored.) The cost of being early is less than the cost of being delayed.

Two objectives apply. The first is to maximize the on-time performance for the customers. This is done by minimizing the combination of the average delay for all customers (a mini-avg or mini-sum objective which is an L-1 norm) and the greatest lateness among all customers (which is an L-∞ norm). The second objective is to minimize total cost. Other "objectives" such as minimizing the fleet size and the duration of time required to complete all deliveries (the makespan) are treated parametrically.

The choice variables are: 1) the assignment of requests to trucks, 2) the sequencing of visits to customers, and, 3) in the case of the Type 2 AWs, the start times. The fleet size is an input as well as the start times for the Type 1 AWs. The sequencing and assignment introduces slack to minimize late arrivals, ensuring early or OTAs. Adding slack lengthens the tour durations. Larger fleet sizes improve on-time performance but also increase cost.

The search procedure tracks the performance of various routing and scheduling solutions. The run-cutting procedure develops the solutions. The procedure can deal with large-scale problems and is easy to understand. Optimally cannot be assured, but the procedure's performance can be compared to optimal solutions for small problems.

The equivalent math programming formulation is a stochastic version of Bender's decomposition. Each sub-model is a scenario realization of the travel times and load-unload times. It has a

probability that the scenario arises. These probabilities are used as weights in computing the overall objective function. The overarching choice variable is the fleet size. The assignment of loads to trucks and the tour sequences vary by sub-model. The overriding purposes are to 1) identify a fleet size that can accommodate the stochasticity in an acceptable manner and 2) seek general patterns in the assignment of loads to trucks.

The pseudo-code representation of the BBM is shown in Figure 3.1. Based on the results for each fleet size provided by the algorithm, determine which solution is optimal: the trade-off between delay performance and fleet size. The truck tours are developed using a greedy heuristic that assigns the best available truck to the loads in chronological order, seen prominently in lines 15-21 of the pseudo code.

Building Block Model: Assign service request sequences (stops) to trucks

```
1: function Assign and Sequence (Graph G, Stop info array Stop, Fleet sizes array M, Simulation specifications S)
2: for all Fleet sizes k ∈ M do
3:     for all n = 1,...,S.numRuns do
4:         for all i,j node pairs in G do
5:             Sample for travel time t[i,j]
6:         for all i Stops in Stop do
7:             Sample for service time Stop[i].tSvc
8:         Create array VI containing Stop indices ordered by earliest arrrival window
9:         for all trucks j ∈ 1,...,k do
10:            veh[j].arrive,veh[j].begin,veh[j].early,veh[j].depart,veh[j].delay ← 0
11:            veh[j].loc ← 0 where location 0 is depot
12:        maxDelay,aveDelay,timesDelayed ← 0
13:        for all Indices i ∈ V I do
14:            vehTemp ← veh
15:            for all trucks j ∈ 1,...,k do
16:                vehTemp[j].arrive ← t[veh[j].loc,Stop[i].node]
17:                vehTemp[j].early ← max(Stop[i].bWin − vehTemp[j].arrive,0)
18:                vehTemp[j].begin ← max(vehTemp[j].arrive,Stop[i].bWin)
19:                vehTemp[j].depart ← max(vehTemp[j].begin+Stop[i].tSvc,Stop[i].eWin)
20:                vehTemp[j].delay ← max(vehTemp[j].depart−(Stop[i].eWin+OTW),0)
21:        Select jBest such that vehTemp[jBest].arrive ≤ vehTemp[j].arrive and vehTemp[jBest].delay ≤
           vehTemp[j].delay ∀j ∈{1,...,k},j 6= jBest
22:            if veh[jBest].delay > 0 then
23:                aveDelay ← aveDelay + veh[jBest].delay
24:                timesDelayed ← timesDelayed + 1
25:                if veh[jBest].delay > maxDelay then
26:                    maxDelay ← veh[jBest].delay
27:                vehTemp[jBest].delay ← veh[jBest].delay + vehTemp[jBest].delay
28:            veh[jBest] ← vehTemp[jBest]
29:            load[i] ← veh[jBest]
30:            load[i].veh ← jBest
31:        for all Indices i ∈ V I do
32:            delayStops[n,i] ← load[i].delays
33:        for all trucks i ∈ 1,...,k do
34:            delayVehicles[n,i] ← veh[i].delay
```

35: *maxDelays*[*n*] ← *maxDelay*
36: *aveDelays*[*n*] ← *aveDelay/timesDelayed*
37: Provide summary of results through *aveDelays,maxDelays,latenessStops,* and *latenessTrucks* before iterating to next fleet size

Figure 3.1: Block building method

3.2 CASE STUDIES

The case study is hypothetical. It is based on a 10-node network. Node #1 is the depot. A single business day is examined using 200 realizations. The day is assumed to be 10 hours long (600 minutes). The fleet size can range from 1 to 10 trucks. That value is set before each solution is obtained and its impact on the objectives is explored parametrically. All trucks originate and terminate tours at the depot (node #1). An upper bound is imposed on the capacity of each truck.

The hypothetical network is developed using a set of stochastic equations. The equation for the nominal travel times is $t_{ij} = a_{ij} + b_{ij} r_{ij}$ where t_{ij} is the travel time, r_{ij} is a uniform random variable on the interval [0,1] and a_{ij} and b_{ij} are constants. Table 3.1 shows the set of values used. The unit of travel time is one minute. The travel times are assumed symmetric. That is, t_{ij} is the same as t_{ji}.

					Times					
	1	2	3	4	5	6	7	8	9	10
1	1	20	10	13	15	21	16	15	28	19
2	20	1	21	22	18	7	17	17	22	21
3	10	21	1	19	17	14	17	20	5	30
4	13	22	19	1	22	28	30	29	30	7
5	15	18	17	22	1	12	16	27	16	22
6	21	7	14	28	12	1	28	14	20	10
7	16	17	17	30	16	28	1	6	28	23
8	15	17	20	29	27	14	6	1	27	15
9	28	22	5	30	16	20	28	27	1	12
10	19	21	30	7	22	10	23	15	12	1
10	= #Loc			1 Time unit = 1 minute						

Table 3.1: Example network travel times

Twenty service requests exist. Each has a location (2.. 10), a beginning time for servicing/ visitation (*bWin*), an ending time for servicing (*eWin*), and a service time (*tSvc*). The locations of the service requests are determined quasi-randomly. They are placed so that each location has between one to three service requests. The equation for the nominal service times is $t_s = a_s + b_s r_s$ where t_s is the service time for stop *s*, r_s is a uniform random variable on the interval [0,1] and a_s and b_s are constants.

The case study set of service requests is shown in Table 3.2. The unit of time is one minute. As indicated before, there is an initial stop #0 at the depot which begins the tour and a stop #21 at the depot which ends the tour.

Stop	Loc	bWin	eWin	tSvc
1	8	135	165	18
2	10	240	270	20
3	2	195	255	23
4	7	15	45	20
5	4	180	225	23
6	3	240	255	16
7	3	105	150	23
8	6	195	225	20
9	8	285	315	23
10	9	435	480	21
11	8	195	240	16
12	6	240	255	21
13	10	285	345	20
14	3	360	390	23
15	5	405	450	18
16	9	300	360	15
17	4	255	300	16
18	8	345	375	15
19	6	255	285	24
20	2	300	315	24

Table 3.2: Example set of service requests and their characteristics

For each of the 200 scenario realizations (set of actual travel times and service times for which the problem is solved), stochasticity is introduced (for a given, randomly generated problem setting, described above) by pre-multiplying the nominal time values t_{nom} by a random variable r_k. The value of r_k follows a four-point discrete distribution. The possible values of r_k are 0.8, 1.0, 1.5, and 2.0 and they have probabilities of 10%, 50%, 20% and 10%. To illustrate, this means there is a 20% chance that r_k will be 1.5 and that t_k will be 1.5 times the nominal value. The travel times and load/unload times are both treated this way. They are assumed to be determined independently.

Trucks that arrive before bWin (see Figure 3.1) must wait until bWin to start their servicing. They are considered late if they start servicing after bWin + OTW where OTW is the duration of the on-time window. OTW is set to 10 minutes. Trucks cannot leave before eWin. They are considered delayed if they depart later than eWin + OTW. The same value of 10 minutes is employed for this OTW.

Figure 3.2 shows the distribution of delays for 200 scenario realizations based on the set of nominal travel times and service times previously shown in Tables 3.1 and 3.2. The number of realizations examined is 200. Three trucks are employed.

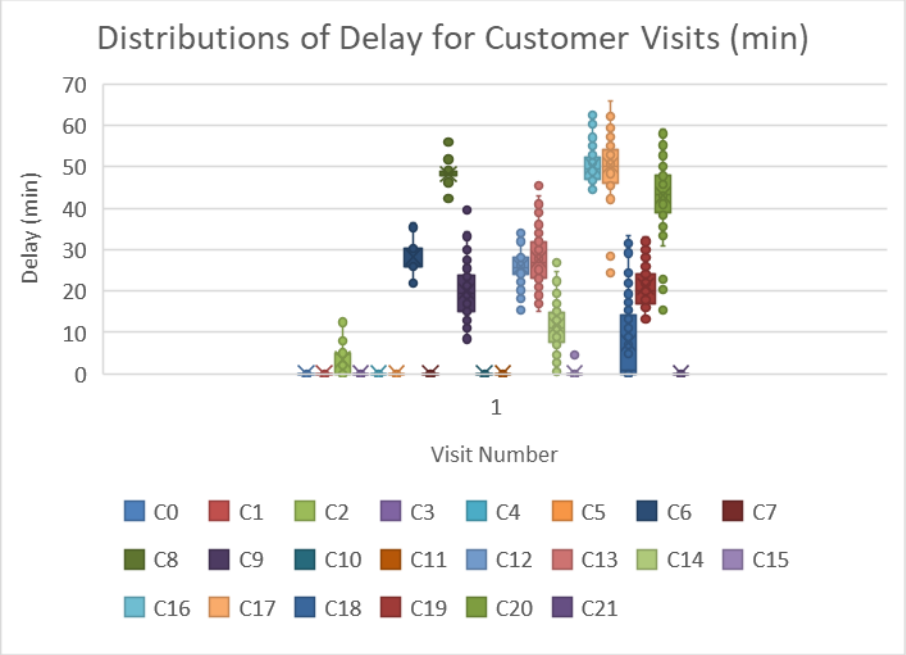

Figure 3.2: Distributions of delays for customer visits – three trucks employed

As can be seen, for about 10 of the service requests, the delay is almost zero in every scenario. For four others, requests 8, 16, 17, and 20, the delay varies up to 40 minutes or more. Of these, request 18 typically has delays of 10 minutes or less, but the value can reach up to 30.

Figure 3.3 shows the distribution of tour times for the three trucks employed. All of them are about 400 minutes (about 5 hours) although the range is 380 to 440. This means a solution involving three trucks is feasible although the delays may be greater than desirable.

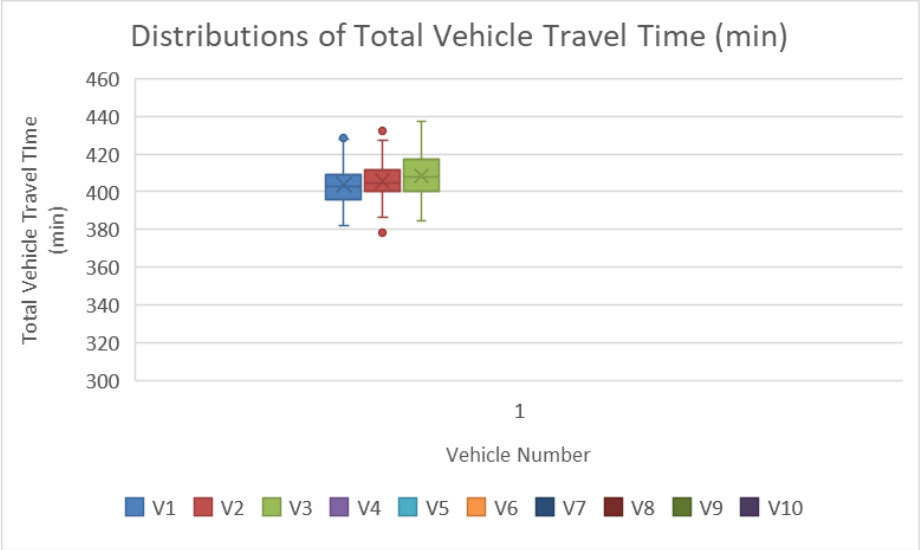

Figure 3.3: Tour Durations for the three trucks in the scenario analyzed

Since the travel times and service times are stochastic, no one assignment of requests to the trucks is optimal. The one that is best in each scenario (realization) depends on the values that pertain for

the travel times and servicing times. Table 3.3 shows how frequently (out of 200 realizations) a given truck is assigned to a specific request. For example, truck #1 is always assigned to request #7 (Veh 1 and L7). For request 11, there is one instance in which the request is assigned to Truck #1. In the remaining 199 realizations, it is assigned to Truck #3. This shows the strength of BBM's ability to adjust the assignment of requests to trucks depending upon the nuances of each scenario realization.

	Vehicles by Load - Block Building Solution																					
Veh	L0	L1	L2	L3	L4	L5	L6	L7	L8	L9	L10	L11	L12	L13	L14	L15	L16	L17	L18	L19	L20	L21
1	0	0	0	199	0	0	193	200	1	23	48	1	7	9	18	87	22	173	78	2	100	65
2	0	0	87	0	200	200	6	0	113	108	78	0	116	71	83	65	76	12	72	85	52	57
3	200	200	113	1	0	0	1	0	86	69	74	199	77	120	99	48	102	15	50	113	48	78

Table 3.3: Example set of service requests and their characteristics

The results for L0 and L21, the departure from and return to the depot are not relevant. The focus is on L1 through L20. The color shading gives a sense of frequency of assignment. Green means not at all. Red means all the time. There are some requests like L7 where the assignment is always to truck #1, but for others, like L15, the assignment varies.

As would be expected, the fleet size has a major impact on the quality of the solution. Figure 3.4 shows the performance that results from fleet sizes ranging from 1 to 6 trucks. The figure presents data for 1) the early arrivals and 2) delays, both average and maximum values. A visit to a customer is considered early if the truck arrives before the beginning of the servicing window. The visit is "late" if the truck arrives later than 10 minutes after the beginning of the service window. The visit is "delayed" if the truck leaves later than the end of the service window.

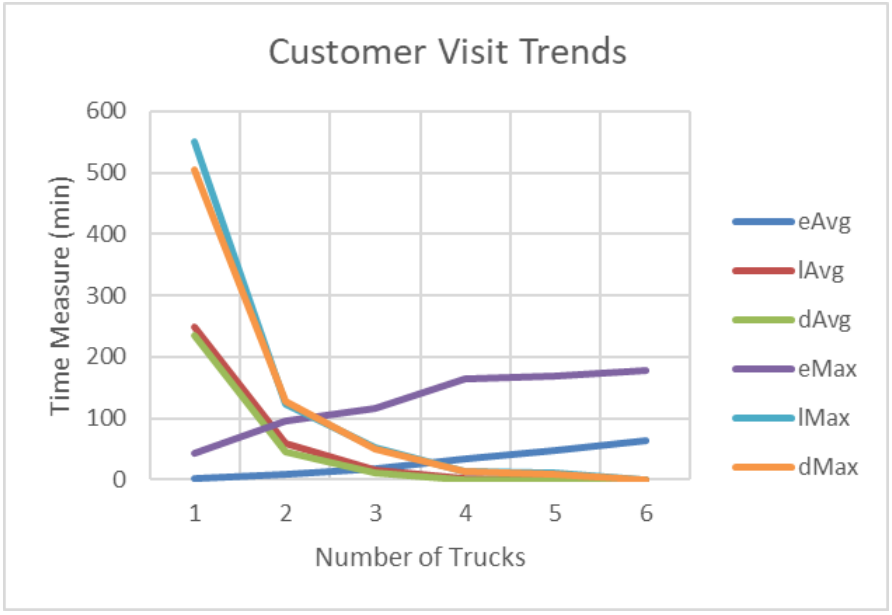

Figure 3.4: Trends in early arrivals and delays

It is easy to see that the delays fall toward zero rapidly as the fleet size increases from 1 to 3 trucks. At 6 trucks, the delays are all zero. Hence, for this setting, a fleet size of 4 or more trucks ensures that reasonably high-quality service will be provided.

It is also important to view these trends from the perspective of the trucks. Figure 3.5 presents trends in 1) the number of visits assigned per truck, 2) the extent to which the trucks are early and 3) the extent to which they are delayed; both average and maximum values.

It is easy to see that truck performance improves dramatically as the fleet size grows from 1 to 4 trucks. Beyond that, the improvement is very gradual. The delay performance improves most dramatically, from an initial value of 250 minutes down to nearly zero. It is also easy to see that the reason why this happens is because the early arrivals continue to increase, starting from zero and reaching up to 60 minutes. The implication is that to achieve the objective of minimizing delays (providing reliable service), at least 4 trucks are required and early arrivals of 50 minutes are more are involved.

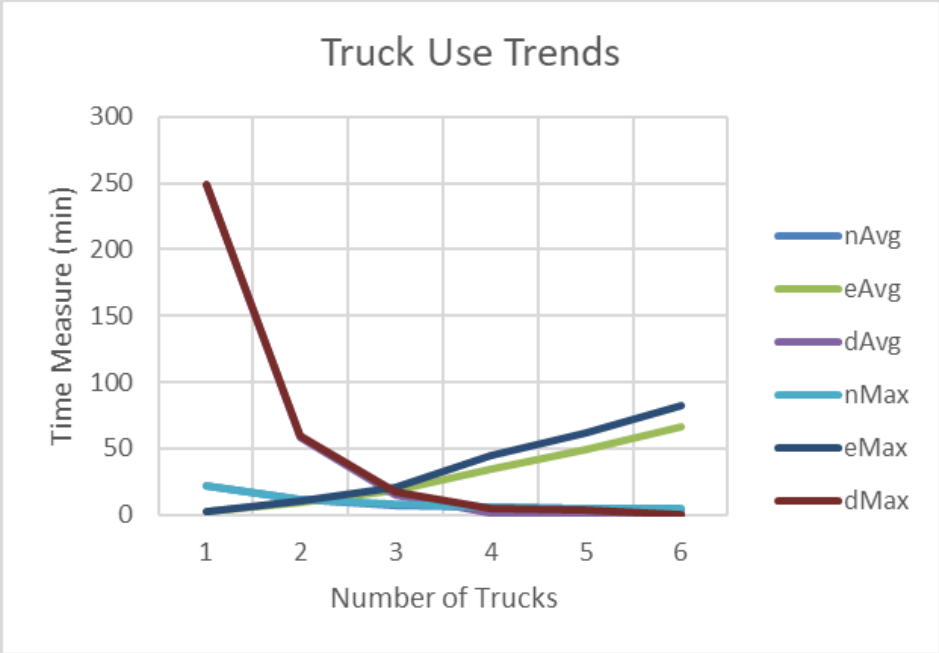

Figure 3.5: Trends in truck performance as affected by fleet size

3.3 SUMMARY

This section has presented a block-building method (BBM) for solving the stochastic truck routing and scheduling problem. The BBM makes multiple passes through the set of customers to be visited based on Monte Carlo simulations of the location-to-location travel times and the loading/unloading times and identifies assignments of the trucks to the customer visits which are both efficient and feasible.

One of the main insights provided by the BBM is the relationship between the reliability of the service provided and the size of the fleet employed. As would be expected, larger fleet sizes result in greater reliability. Most importantly, the analysis quantifies this relationship. It shows the extent to which on-time performance is improved through greater fleet sizes. It also allows the analyst to see if there are consistent patterns in the assignment of customer visits to trucks across the problem realizations examined.

4.0 GENETIC ALGORITHM BASED METHOD (GAM)

This section presents a genetic algorithm based vehicle routing and assignment method (GAM) that creates solutions to the stochastic VRP problem. Much like the BBM, the GAM finds a plan (an assignment of requests to trucks) that is best suited to every scenario examined. The assessment portion of the BBM described in Section 3.0 is used to assess the quality of each plan (in part, for consistency with the previous work). A fitness function is used to identify the best plan, as is common with GA-based procedures. The GAM creates new plans in each iteration both by fusing the genes from previously identified good plans and mutating genes that are part of those prior plans. The new, candidate plans are then evaluated for fitness by the assessment part of the BBM (not the truck assignment part) and the results are passed back to the GAM. The fitness results are compared with the fitness of previous plans. The best plans are kept, the poorest ones are discarded, and the remaining set is used to create new candidate plans for assessment in the next iteration.

4.1 METHODOLOGY

The basic tenets of the problem are the same as those presented in Section 3. However, four objectives are considered instead of just one. They are: i) the average delay for all requests (a mini-avg or mini-sum objective which is an L-1 norm), ii) the average earliness (arrival before the servicing time), iii) the maximum tour duration for all trucks (which is an L-∞ norm), and iv) the maximum load-to-capacity ratio among all the trucks (which is again an L-∞ norm).

Other objectives could have been examined and included. In future work they should be considered. Options include the maximum lateness among all requests and the total cost (truck-miles, truck-hours, or a combination of the two) of the solution. The ones that are employed capture the demand-supply tradeoff between providing high-quality service (the first objective) and the resource intensity of providing that service (the other three including the extent to which the resources are stressed as in the maximum tour duration and the maximum load-to-capacity ratio). Moreover, using the maximum load ratio rather than setting an upper bound at the capacity of the trucks allows potential solutions to be assessed even if the load ratios exceed capacity. The analyst parametrically assesses the value of increasing the fleet size and needs to ascertain how large the fleet needs to be to fit within the capacity of the trucks.

The objectives are combined using a “fitness” function, which is common in GA-based searches. Weights are applied to the objectives to create a summary statistic – arguably the “fitness” of the solution for continued use as a parent for future offspring. The weights employed in the case studies presented here are respectively 0.3, 0.1, 0.1, and 0.1. But these values clearly are based on the importance perceived by the analyst and can be changed.

As the search progresses, the performance of the best plan improves. New ones are created. Old ones are rejected. A minimum number of plans (here, 100) is always included in the population.

A maximum exceedance over the best fitness value is imposed. That is, if a given plan exceeds that limit, it is discarded. If this limit is set high, the set of plans remains rich but poor plans are

included. If it is set low, the number of plans remains relatively small and only the best ones are kept.

The choice variables are: 1) the assignment of requests to trucks and 2) the sequencing of the servicing events. The solution introduces slack to minimize late arrivals and ensure early or OTAs. Doing so lengthens the tour durations and decreases truck productivity (requests accommodated per tour). Larger fleet sizes improve on-time performance but also increase cost.

The GAM uses the best, prior plans to create new plans. If there are three trucks in the fleet, an example plan is shown in Figure 4.1.

L1	L2	L3	L4	L5	L6	L7	L8	L9	L10	L11	L12	L13	L14	L15	L16	L17	L18	L19	L20
2	3	1	3	2	2	1	3	1	1	1	1	2	2	3	3	2	2	3	3

Figure 4.1: Example assignment of service requests to a fleet of three trucks

In this plan, for example, the assignment of servicing requests for truck #2 is 1, 5, 6, 13, 14, 17, and 18. This is *not* necessarily the order in which the requests are serviced. It is *only* the list of request assignments.

The active plan population at all points in time is the same. In the analyses presented here, it is 100 plans. Initially, the plans are created using random truck assignments to the requests. As the search continues, the population becomes a combination of plans identified through the search process.

Each new plan is passed to the assessment portion of the BBM for evaluation (not the truck assignment portion) so that its fitness can be evaluated. Because the truck assignments are pre-determined by the GAM, the BBM treats the service request assignments as being fixed (and not to be determined, as the BBM would normally do). This means the BBM solves traveling salesman problems for each truck rather than a VRP in that the request assignments are already prescribed by the GAM. The BBM places the requests for each truck in chronological order. Then it creates itineraries and assesses their quality. The overall fitness of the plan is then assessed. The results of this effort are then passed back to the GAM.

In each generation *k*, the following steps are followed:

1) Assess the fitness of all the new solutions based on the assessment portion of the BBM.
2) Sort all the current solutions into ascending order based upon their fitness.
3) Find the plan with the lowest fitness value.
4) Reject the plans that have fitness values greater than the maximum exceedance allowed. Archive these plans.
5) Ascertain the size of the remaining population and determine how many new plans must be added to return the set of plans to full population size.
6) Generate new plans. Based on probabilities supplied by the analyst, either a) use two parents to generate an offspring by combining their genes, b) use a single parent to create an offspring through mutation that combines the genes of both parents, or c) create a new plan through random assignment without replacement.

The GAM stops searching when a prescribed number of generations have been assessed. This number is set by the analyst. In the case study presented below, the number of generations is 20.

4.2 CASE STUDY

The case study involves the same hypothetical situation considered in Section 3 for the BBM. Table 4.1 shows the same set of travel times presented in Section 3.0. The unit of travel time is one minute. The travel times in the network are symmetric. That is, t_{ij} is the same as t_{ji}.

					Times					
	1	2	3	4	5	6	7	8	9	10
1	1	20	10	13	15	21	16	15	28	19
2	20	1	21	22	18	7	17	17	22	21
3	10	21	1	19	17	14	17	20	5	30
4	13	22	19	1	22	28	30	29	30	7
5	15	18	17	22	1	12	16	27	16	22
6	21	7	14	28	12	1	28	14	20	10
7	16	17	17	30	16	28	1	6	28	23
8	15	17	20	29	27	14	6	1	27	15
9	28	22	5	30	16	20	28	27	1	12
10	19	21	30	7	22	10	23	15	12	1
10	= #Loc			1 Time unit = 1 minute						

Table 4.1: Network travel times

Twenty requests are specified, as shown in Table 4.2. These are the same as the ones shown in Section 3. Each has a location (2 .. 10), a beginning time for servicing (*bWin*), an ending time (*eWin*), and a service duration (*tSvc*). The unit of time is one minute. (As was the case in Section 3, there are two more requests, #0 which is departure from the depot at the beginning of the tour and #21 which is the return to the depot.)

Trucks that arrive before bWin must wait until bWin to start their servicing. They are considered late if they start servicing after bWin + OTW where OTW is the duration of the on-time window. OTW is set to 10 minutes. Trucks cannot leave until eWin. They are considered delayed if they depart later than eWin + OTW. The same value of 10 minutes is employed.

A business day of 10 hours is assumed, which is equivalent to 600 minutes. For each realization of the problem setting, a single day is examined.

The fleet size can vary from 1 to 10 trucks. It is set before each model run and its impact on the objectives is explored parametrically. All truck tours originate and terminate at the depot (node #1). An upper bound can be imposed on the capacity of each truck.

The problem scenarios (realizations) are created by pre-multiplying the nominal times t_{nom} by a random variable r_k. Moreover, r_k follows a four-point discrete distribution. Specifically, the possible values of r_k are 0.8, 1.0, 1.5, and 2.0 and they have probabilities of 10%, 50%, 20% and 10%. To illustrate, this means there is a 20% chance that r_k will be 1.5 and therefore t_k will be 1.5 times the nominal value. The travel times and load/unload times for a given scenario (realization) are both created this way. The number of realizations studied here is 200.

Stop	Loc	bWin	eWin	tSvc
1	8	135	165	18
2	10	240	270	20
3	2	195	255	23
4	7	15	45	20
5	4	180	225	23
6	3	240	255	16
7	3	105	150	23
8	6	195	225	20
9	8	285	315	23
10	9	435	480	21
11	8	195	240	16
12	6	240	255	21
13	10	285	345	20
14	3	360	390	23
15	5	405	450	18
16	9	300	360	15
17	4	255	300	16
18	8	345	375	15
19	6	255	285	24
20	2	300	315	24

Table 4.2: Visits and their characteristics

Figure 4.2 shows that for one of the scenarios examined (the last one), the GAM starts with a fitness value of about 120 and progresses very quickly toward a value in the 50-60 range.

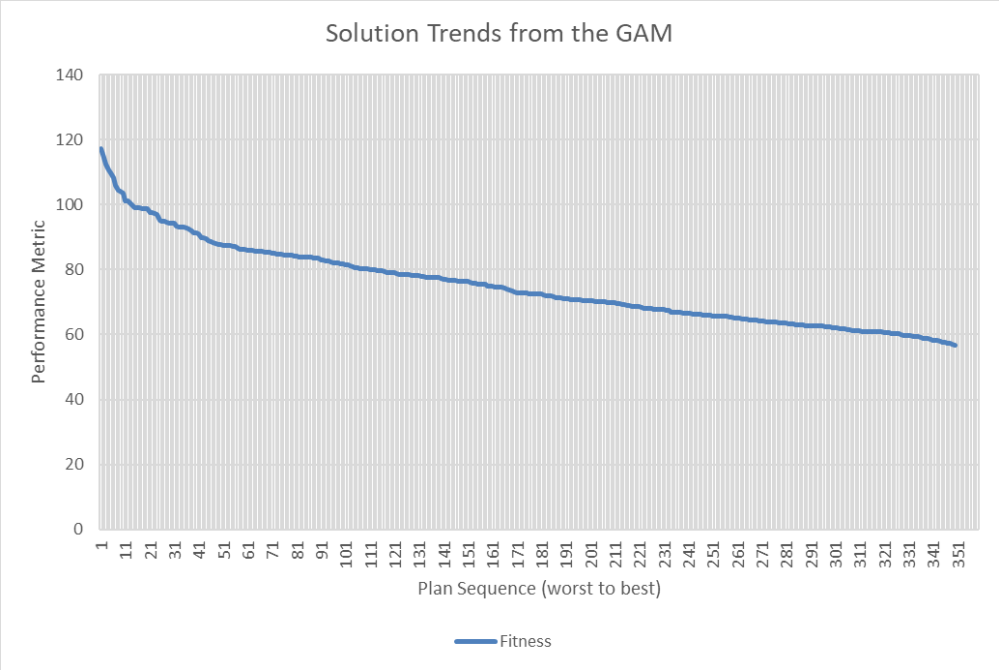

Figure 4.2: Trends in the fitness metric

The final fitness value is 56.8. The last 100 plans (from 250-350) are part of the active population at the time the search ceases. The rest (from 1-250) are part of the archive.

The fitness value trends in Figure 4.2 can be compared with 48 which is the value obtained by the BBM. This is disappointing, but the GAM starts from no idea about what the plan should be. Also, the GAM can be made sensitive to multiple objectives and identify solutions that achieve tradeoffs among those metrics depending upon their relative importance.

The trends in the average delay and average wait time show trends like those for the fitness metric. As Figure 4.3 shows, the average delay drops precipitously from about 120 to 11. Since the solutions are sorted in descending order based on fitness, there are some upward trends from one plan to the next, but the general trend is diminishing.

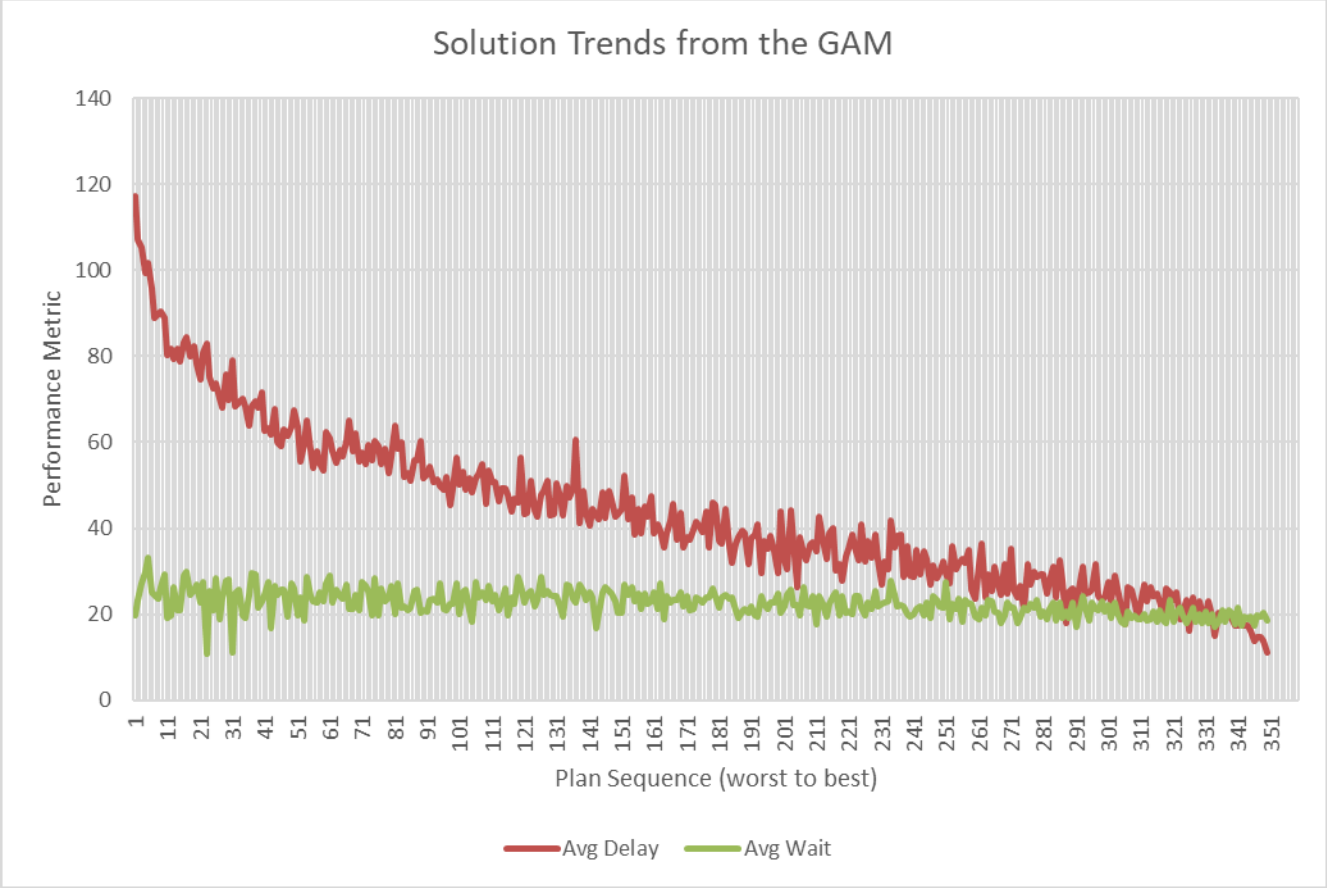

Figure 4.3: Trends in average delay and average wait time

From the standpoint of the trucks, the trends in the average and maximum tour durations show similar downward decreases, significant at first, and then less change, as shown in Figure 4.4. At about the 60th plan, the maximum tour duration fits within the 600-minute daily limit. Eventually, it falls to about 515, which is about 86% of the day's duration.

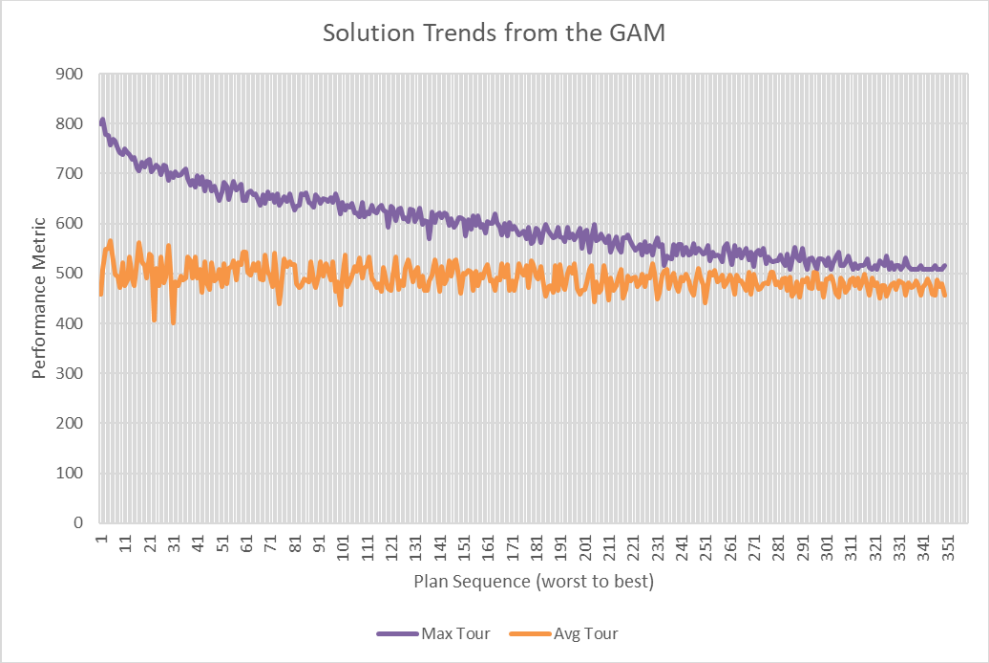

Figure 4.4: Trends in the maximum and average tour durations (in minutes)

Finally, trends in the load ratio are shown in Figure 4.5. It is not until about the last 30 solutions that the maximum value hovers around 1.0. This makes it clear that 3 trucks is about the minimum number required to accommodate all 20 service requests.

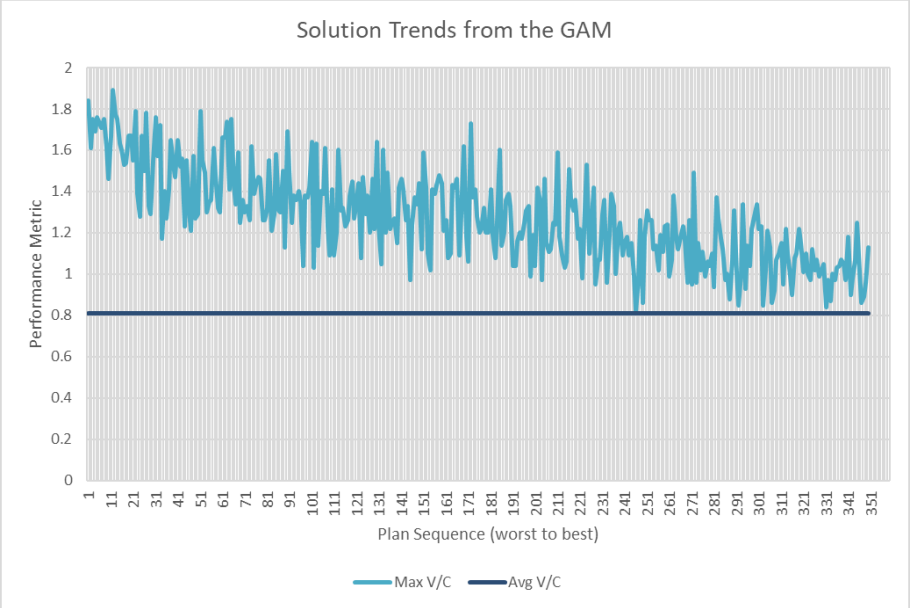

Figure 4.5: Trends in the maximum and average load factors

At a more detailed level, Figure 4.6 shows the distributions in delay for the 20 service requests based on the best solution identified. There are 15 requests with non-zero delays while there were 11 with the BBM solution. Many of the requests with delays are the same, as should be the case since the same problem setting is being examined. The ranges of delays are very similar.

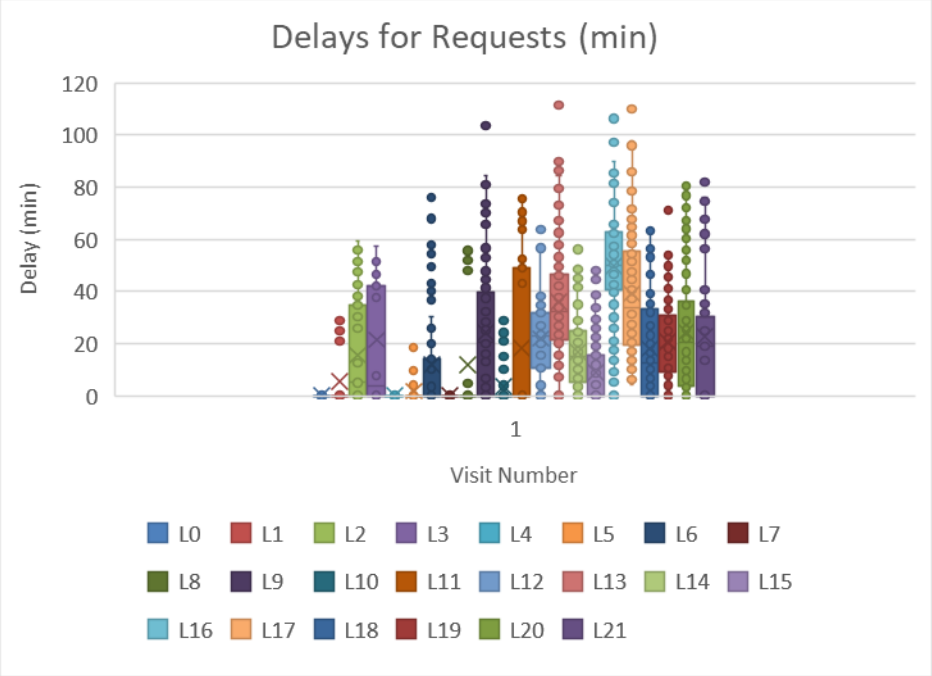

Figure 4.6: Distributions of delays for customer visits – 3 trucks employed

Finally, Figure 4.7 shows the distribution of tour times for the three trucks used out of a possible fleet of 10. The values are all slightly greater than those found with the TA methodology.

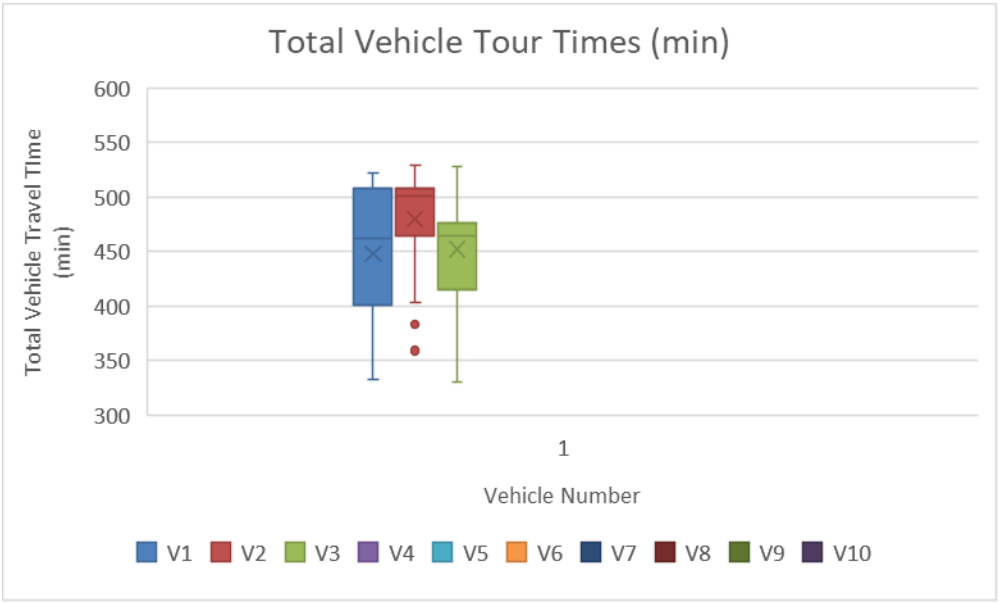

Figure 4.7: Tour Durations for the three trucks in the scenario analyzed

As would be expected, the fleet size has a major impact on the quality of the solution. Figure 4.8 shows the performance that results from fleet sizes ranging from 1 to 6 trucks. The figure presents data for 1) the early arrivals and 2) delays, both average and maximum values. A visit to a customer is considered early if the truck arrives before the beginning of the servicing window. The visit is

"late" if the truck arrives later than 10 minutes after the beginning of the service window. The visit is "delayed" if the truck leaves later than the end of the service window.

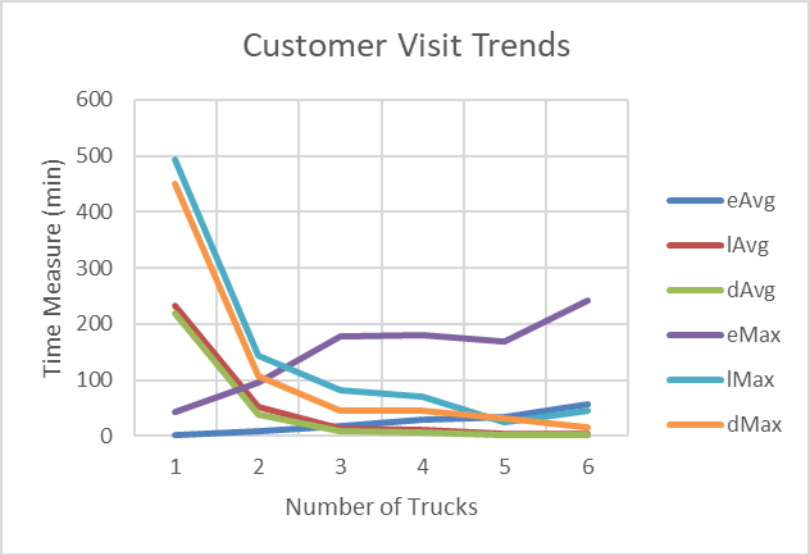

Figure 4.8: Trends in early arrivals and delays

It is easy to see that the delays fall toward zero rapidly as the fleet size increases from 1 to 4 trucks. At 6 trucks, the delays are all zero. Hence, for this setting, a fleet size of 4 or more trucks ensures that reasonably high-quality service will be provided.

It is also important to view these trends in performance from the perspective of the trucks providing the service. Figure 4.9 presents trends, both average and maximum values, in 1) the number of visits assigned per truck, 2) the extent to which the trucks are early and 3) the extent to which they are delayed.

It is easy to see that truck performance improves dramatically as the fleet size grows from 1 to 4 trucks. Beyond that, the improvement is very gradual. The delay performance improves most dramatically, from an initial value of 250 minutes down to effectively zero. It is also easy to see that the reason why this happens is because the early arrivals continue to increase, starting from zero and reaching up to an average of 120 minutes. The implication is that to achieve the objective of minimizing delays (providing reliable service), at least 4 trucks are required and early arrivals of 50 minutes are more are involved.

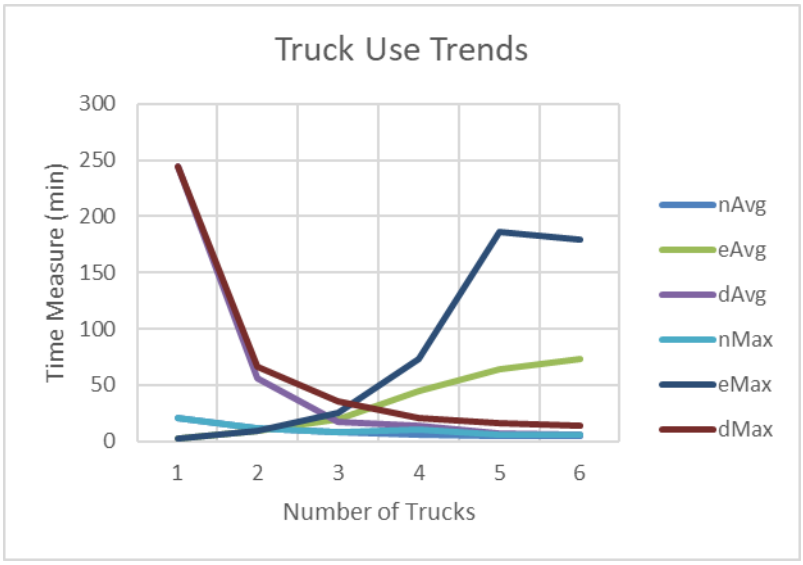

Figure 4.9: Trends in truck performance as affected by fleet size

4.3 SUMMARY

This section has presented a genetic algorithm-based method (GAM) that solves the stochastic truck routing and scheduling problem. Its objective is to find an assignment of loads to trucks and routings for the trucks that optimizes all the performance metrics. Its objectives are to 1) maximize the on-time performance and 2) minimize the cost of the service provided subject to: a) picking up all shipments and/or delivering all shipments, b) not exceeding the capacity of any truck, and c) not exceeding a maximum tour duration for any truck.

One of the main insights provided by the GAM is the relationship between the reliability of the service provided and the size of the fleet employed. As would be expected, it shows that higher reliability is provided by larger fleet sizes. But more importantly, it quantifies the extent of that improvement through the stochastic analysis presented. It also allows the analyst to see if there are consistent patterns in the assignment of customer visits to trucks across the problem realizations examined.

5.0 SEEKING A ROBUST PLAN

Instead of using the GAM to find the best plan for each scenario (realization), as is done in the work described in Section 4, the analyst can focus on finding the best plan (most robust plan) for a given set of scenarios. In this case, the plan creation is the "outside loop" and the examination of the scenarios is the "inner loop". A fitness function is still employed, but the fitness assessment is based on the performance of a given plan across all scenarios. The GAM still creates new plans in each iteration both by fusing the genes from previously identified good plans and mutating genes that are part of those prior plans. The new, candidate plans are then evaluated for fitness by the assessment part of the BBM (not the truck assignment part) and the results are passed back to the GAM. The fitness results are compared with the fitness of previous plans. The best plans are kept, the poorest ones are discarded, and the remaining set is used to create new candidate plans for assessment in the next iteration.

5.1 METHODOLOGY

The method employed is as follows:

1) Create a population of plans
    a) Discard poor performing plans that exceed a ratio based on the best plan
    b) Use the good plans to create offspring
2) Evaluate the performance of the new plans
    a) Consider all of the scenarios for each plan
    b) Working with a composite fitness function that incorporates the performance of the plan for all scenarios.
3) Repeat steps 1) and 2) until a specified number of generations has transpired

The objectives are the same as those described in Section 4. They are combined using the same fitness function and the same weights. The exceedance limit is imposed.

5.2 CASE STUDY

The same case study environment from Sections 3 and 4 is employed to assess the method's performance. But, a given plan is applied to all 200 of the scenarios considered in Sections 3 and 4 and its performance is assessed based on its performance in all scenarios.

For the last scenario considered, as an example, Figure 5.1 shows that the GAM starts with a fitness value of about 320 and progresses very quickly toward a value in the 50-60 range.

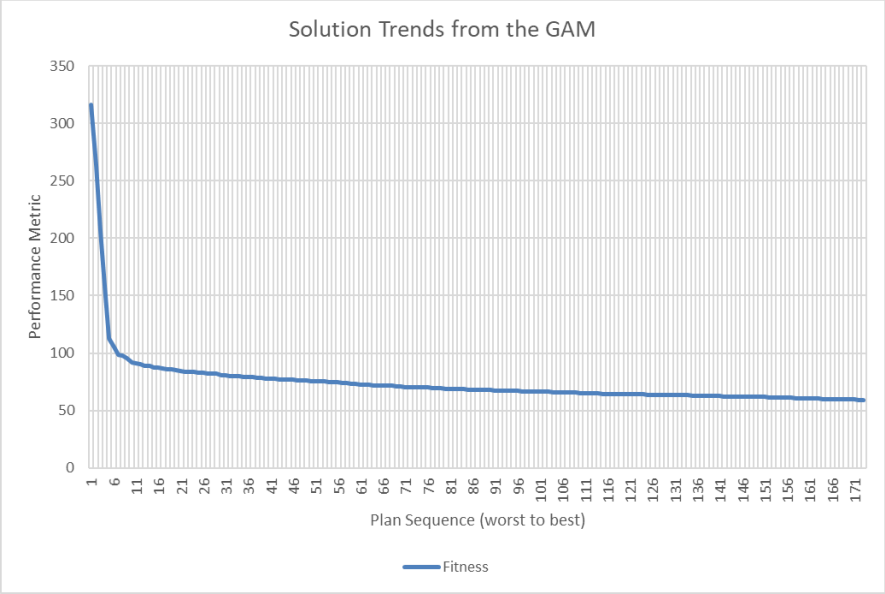

Figure 5.1: Trends in the fitness metric

The final fitness value is 57.6. The last 100 plans (from 75-175) are part of the active population at the time the search ceases. The rest (from 1-75) are part of the archive.

Table 5.1 shows the frequencies of truck assignments to service requests. With some care in thinking, these results can be compared with those in Table 3.3 for the BBM. The difference is that in Table 4.3, the values represent the number of *plans*, out of the 100 best, where a truck was assigned to a service request. In the case of Table 3.3, the values represent the number of scenarios (realizations) in which a given truck was assigned to a specific request. (For the GAM, if the table equivalent to Table 3.3 was presented, one of the three trucks would *always* have been assigned to a specific request because that is the way the plans are developed and specified.

	Vehicles by Load - GA Solution																					
Veh	L0	L1	L2	L3	L4	L5	L6	L7	L8	L9	L10	L11	L12	L13	L14	L15	L16	L17	L18	L19	L20	L21
1	0	25	34	21	19	25	28	23	23	26	29	36	27	28	22	24	20	32	21	31	28	0
2	0	50	40	32	42	52	23	41	29	40	33	29	27	36	51	27	43	43	42	48	30	0
3	0	25	26	47	39	23	49	36	48	34	38	35	46	36	27	49	37	25	37	21	42	0

Table 5.1: Truck assignments to service requests

The trends in the average delay and average wait time are similar to the trend for the fitness metric. As Figure 5.2 shows, the average delay drops precipitously from about 870 to 11.2. Since the solutions are sorted in descending order based on fitness, there are some upward trends from one plan to the next, but the general trend is diminishing.

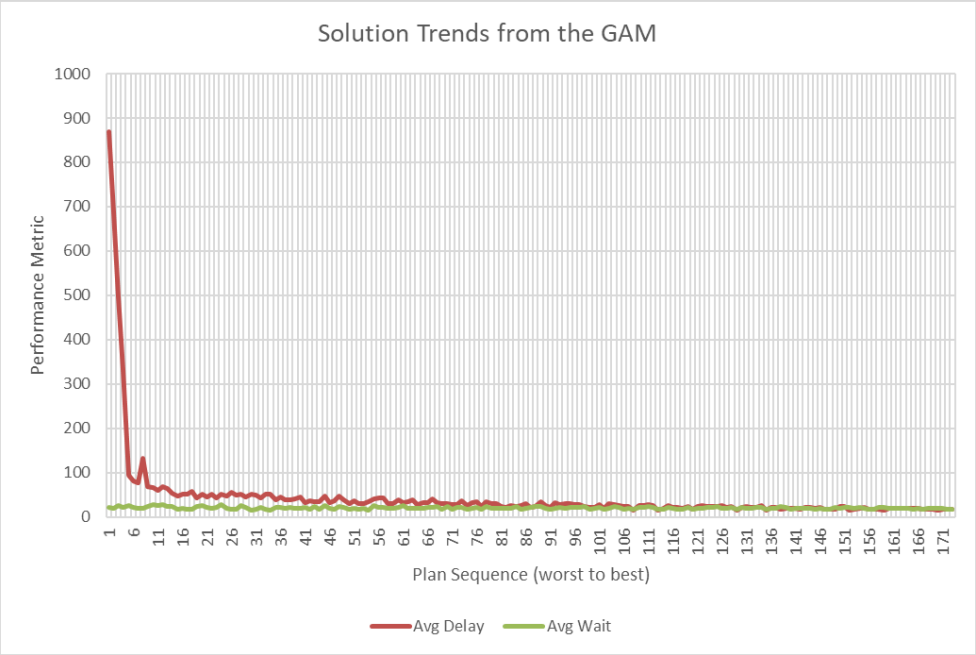

Figure 5.2: Trends in average delay and average wait time

From the standpoint of the trucks, the trends in the average and maximum tour durations show similar downward decreases, significant at first, and then less change, as shown in Figure 5.3. About the 60th plan, the maximum tour duration fits within the 600-minute daily limit. Eventually, it falls to about 520, which is about 87% of the day's duration.

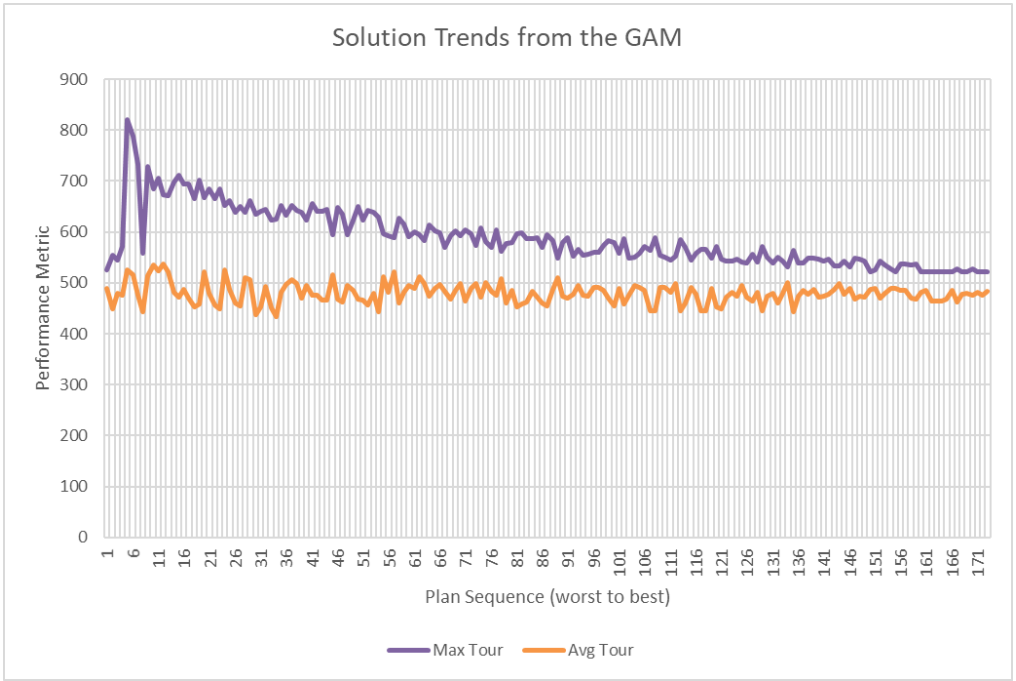

Figure 5.3: Trends in the maximum and average tour durations (in minutes)

Finally, trends in the load ratio are shown in Figure 5.4. It is not until about the last 15 solutions that the maximum value hovers around 1.0. This makes it clear that 3 trucks is about the minimum number required to accommodate all 20 service requests.

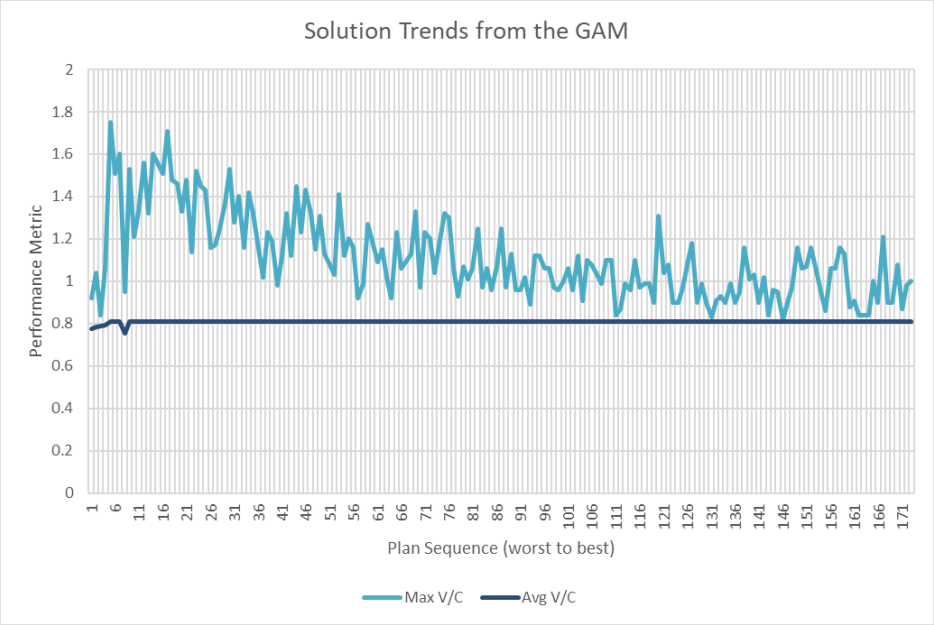

Figure 5.4: Trends in the maximum and average load factors

At a more detailed level, Figure 5.5 shows the distributions in delay for the 20 service requests based on the best solution identified. There are 13 requests with non-zero delays while there were 11 with the BBM solution. Many of the requests with delays are the same, as should be the case since the same problem setting is being examined. The ranges of delays are very similar.

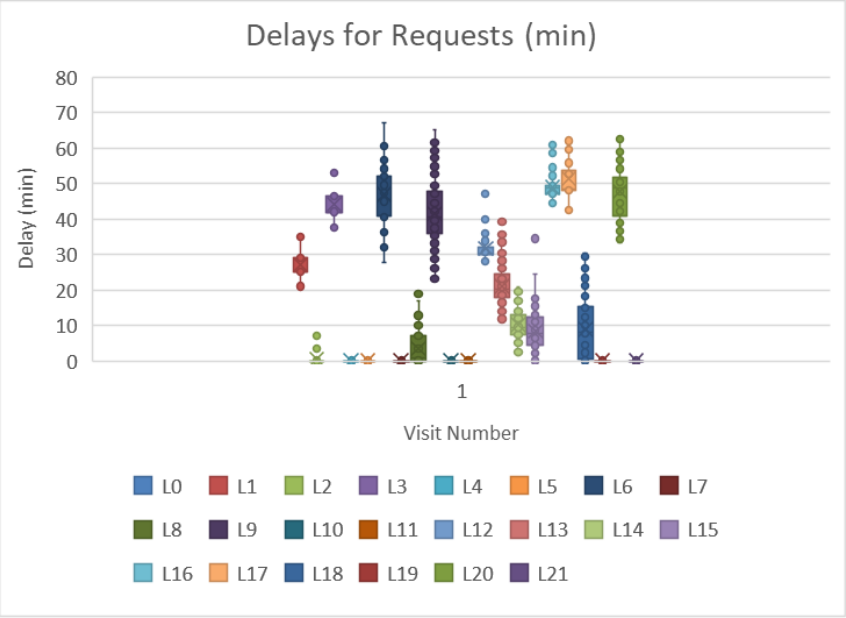

Figure 5.5: Distributions of delays for customer visits – 3 trucks employed

Finally, Figure 5.6 shows the distribution of tour times for the three trucks used of the possible fleet of 10. The values are all slightly greater than those found with the TA methodology.

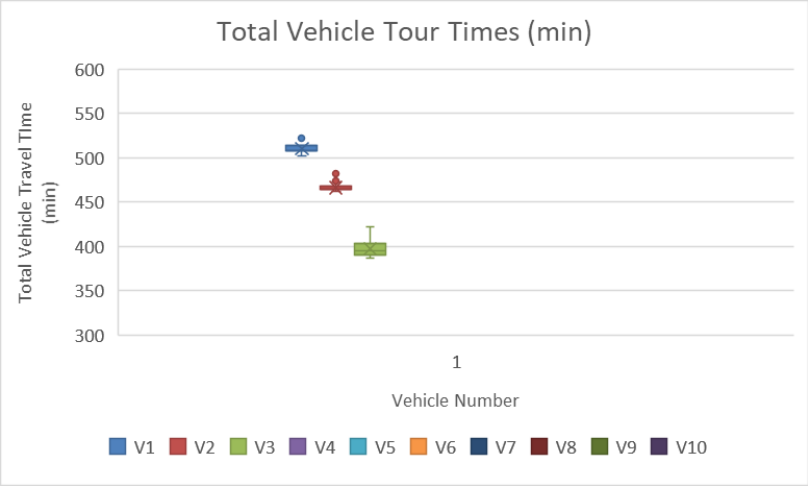

Figure 5.6: Tour Durations for the three trucks in the scenario analyzed

As would be expected, the fleet size has a major impact on the quality of the solution. Figure 5.7 shows the performance that results from fleet sizes ranging from 1 to 6 trucks. The figure presents data for 1) the early arrivals and 2) delays, both average and maximum values. A visit to a customer is considered early if the truck arrives before the beginning of the servicing window. The visit is "late" if the truck arrives later than 10 minutes after the beginning of the service window. The visit is "delayed" if the truck leaves later than the end of the service window.

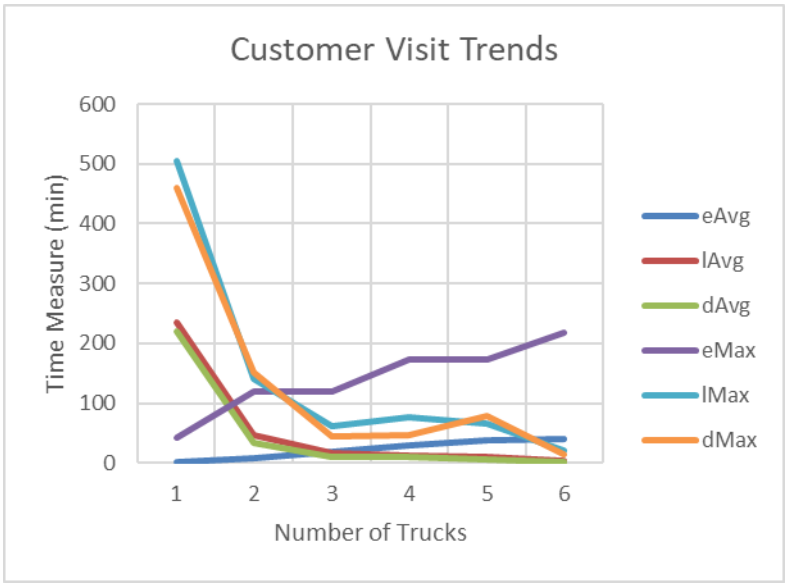

Figure 5.7: Trends in early arrivals and delays

It is easy to see that the delays fall toward zero rapidly as the fleet size increases from 1 to 4 trucks. At 6 trucks, the delays are all zero. Hence, for this setting, a fleet size of 4 or more trucks ensures that reasonably high-quality service will be provided.

It is also important to view these trends in performance from the perspective of the trucks providing the service. Figure 5.8 presents trends in 1) the number of visits assigned per truck, 2) the extent to which the trucks are early and 3) the extent to which they are delayed; both average and maximum values.

It is easy to see that truck performance improves dramatically as the fleet size grows from 1 to 4 trucks. Beyond that, the improvement is very gradual. The delay performance improves most dramatically, from an initial value of 250 minutes down to effectively zero. It is also easy to see that the reason why this happens is because the early arrivals continue to increase, starting from zero and reaching up to an average of 120 minutes. The implication is that to achieve the objective of minimizing delays (providing reliable service), at least 4 trucks are required and early arrivals of 50 minutes are more are involved.

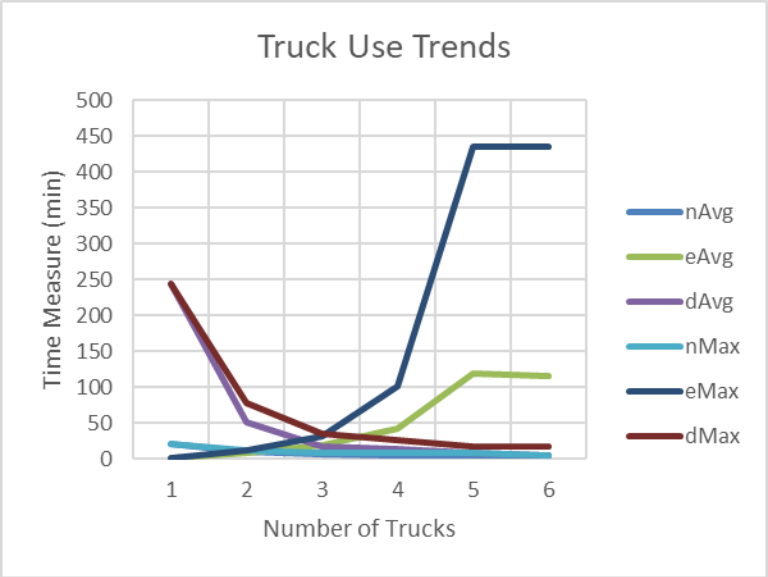

Figure 5.8: Trends in truck performance as affected by fleet size

5.3 SUMMARY

This section has presented an application of the genetic algorithm-based method (GAM) where a given plan is applied to all the scenarios (realizations) simultaneously to create an overall assessment of its performance. From these assessments, a best (most robust) plan is identified. This is different from the assessments presented in Section 4 where the best plan depends on the scenario considered (as is the case in Section 3 as well).

One of the main insights provided by the method is the relationship between the reliability of the service provided and the size of the fleet employed. As would be expected, it shows that higher reliability is provided by larger fleet sizes. But more importantly, it quantifies the extent of that improvement through the stochastic analysis presented. It also allows the analyst to see if there are consistent patterns in the assignment of customer visits to trucks across the problem realizations examined.

6.0 SUMMARY AND FUTURE WORK

6.1 SUMMARY

This research report has described a new genetic algorithm-based method (GAM) for solving the vehicle routing and scheduling problem when the travel times and service times are both stochastic. It deals primarily with truck-related shipments although the tools are applicable to other modes and multi-modal systems. The topic is important because of the economic value that results from minimizing the resource consumption associated with freight activity. Unreliable transport raises costs and diverts scarce factors of production away from other, critically important societal activities. It interferes with the efficiency of the supply chain and increases both monetary and time-related costs and resource requirements (e.g., increased in-process inventory, extra trucks).

The freight industry continues to be concerned with reliability. To be competitive, companies need to remove inefficiencies in their production functions. Both late and early shipments are included in the set of problematic inefficiencies facing freight companies. The industry's emphasis on just-in-time manufacturing has squeezed buffer stock out of the logistics supply chain. It has also raised the risk of stock-outs. Because storage space is reduced as well, early arrivals are problematic. If reliability suffers, all participants in the supply chain must make extra asset investments to buffer the process and ensure that delivery schedules are met. From a societal perspective, the cost of producing the goods and services increases. Extra scarce resources must be devoted to freight-related activities to make the economic system work.

Several insights have been derived from this effort. They will have an impact on the way in which freight reliability analyses are performed in the future.

On-Time Windows. Shippers and receivers have expectations about when shipments are going to depart and when they are going to arrive. People see similar on-time windows when they travel by commercial carriers in the context of published departure and arrival times. They may not be aware of the details, but the carriers measure their on-time performance based on whether the vehicles (trains, planes, buses) depart and arrive at times that are consistent with the published timetable. The same is true for freight. Shippers perceive that packages have left on time if they are picked up by the carrier within a specific window. The window might be wide (a couple of days) or narrow (less than an hour), but a window exists. The same is true for the receiver. There are expectations that shipments will arrive at specific times, or more precisely, within given windows.

Hence freight reliability is ***not*** about travel times per se or even the variance in those travel times. Rather it is about whether shipments arrive and/or depart during these on-time windows. In other words, reliability is assessed not based on travel time distributions but rather whether the arrival or departure was on time or not. That is, reliability is the probability of arriving (or departing or both) during the on-time window.

However, minimizing the variance in the travel times is still an important thought. But there may be no value in minimizing the variance beyond a certain value once the desired on-time performance is achieved. In effect, the better thought is to control the shape of the travel time

distribution, either viewing it as a PDF, or better yet as a CDF, so that a sufficient percentage of the distribution lies within the AW or DW, or both.

Arrival Times. For freight, it is always the arrival times and frequently the departure times that matter. This is different from most personal trip-based analyses where the focus is on the reliability of travel times based on a departure time. The focus of freight movements is on delivering or picking up packages on time. And thus, the question becomes: when must the truck leave the depot so that the shipment will be delivered on time? It is ***not*** when it will arrive given a departure time, as is often the case.

Vehicle Routing and Scheduling Can Address Reliability. Sections 3 and 4 present two methods for considering reliability in developing solutions to truck routing and scheduling problems.

A block-building method (BBM) is presented in Section 3. It develops for good assignments of customer visits to trucks. A "block" is the terminology used in the transit industry to describe a sequence of bus stops on a route that must be serviced before a break is possible. The BBM makes a single pass through the set of requests, in chronological order, and identifies an assignment of trucks which is both feasible and efficient. It is feasible in that the chronology of events for every truck ensures that the servicing requests are accommodated and each truck has sufficient time to travel from one servicing request to the next. It is efficient in that for every request the truck that is closest at hand for accommodating that request is the one selected. The BBM is based on List *et al.* (2003) and List and Turnquist (1993). Those formulations, in turn, are based on Goeddel (1975) as enhanced by Ball, Bodin, and Greenberg (1985) and described by Wren and Rousseau (1995).

A genetic algorithm-based method (GAM) is presented in Section 4. It considers four objectives: i) the average delay for all requests (a mini-avg or mini-sum objective which is an L-1 norm), ii) the average earliness (arrival before the servicing time) for all requests, iii) the maximum tour duration for all trucks (which is an L-∞ norm), and iv) the maximum load-to-capacity ratio among all the trucks (which is again an L-∞ norm). The fact that the GAM method involves searching for solutions (as opposed to the TA method, which simply puts the requests in chronological order) means that an objective function can be imposed and the best solution can be sought.

The objectives are combined using a "fitness" function, which is common in GA-based searches. Weights are applied to the objectives to create a summary statistic – arguably the "fitness" of the solution for continued use as a parent for future offspring. The weights employed in the case studies presented here are respectively 0.3, 0.1, 0.1, and 0.1. But these values clearly are based on the importance perceived by the analyst and can be changed.

Moreover, a maximum exceedance over the best fitness value is imposed. That is, if a given plan that exceeds this limit, it is discarded. If this limit is set high, the set of plans remains rich but poor plans are included. If it is low, the set of plans remains relatively small and only the best plans are kept. In addition, as the search progresses, the performance of the best plan improves. Hence, the numeric values of both the fitness of the best plan, and the exceedance limit continue to decrease. The second method provided a formulation of a bi-objective problem that involves maximizing the on-time performance and minimizing the cost of the service provided.

This is subject to: picking up all shipments and/or delivering all shipments, not exceeding the capacity of any truck, and not exceeding a maximum tour duration for any truck. It assumes the fleet size is fixed, the trucks are always available (that is, maintenance spares exist), and there is a single depot where trucks originate and terminate their tours

It assumes the total time for each tour is the sum of the random variables that describe the travel times between stops and the amount of time spent in pick-up or delivery. This means the travel times are stochastic as are the pick-up and delivery times. It conducts the evaluation by simulating the system's performance multiple times. It samples random variables to establish each realization; assigns loads to the trucks and develop the tours; sequences the loads based on the beginning of their windows; and for each stop, selects the truck that can arrive earliest and has the least delay. The reliability is improved by adding slack time to the schedule; that is, by arriving early, off-site, near a customer's location, at the cost of adding time to the tour and potentially increasing the fleet size. It is also improved by increasing the fleet size. A larger fleet reduces the number of customer stops per truck and adds more resource availability.

6.2 FUTURE WORK

Much future work can be carried out based on the analyses conducted so far. Some important examples of these efforts are as follows:

Real-World Tests. As is often the case, the methodological advances presented here have been tested using a blend of empirical data and hypothetical situations. One natural extension for future work is to test these methods based on datasets that are more representative and reflective of real-world conditions. This pertains to all of the methods presented, from the assessment of reliability for segments and routes to the selection of plans for truck routing and locations for distribution centers.

Vehicle Routing and Scheduling. Vehicle routing and scheduling will continue to be a critical element of the reliability analysis. This report presented two methods for developing a vehicle routing and scheduling plan that maximizes the reliability of the service provided.

The meta-heuristic method could be further improved in several ways. One of which is to incorporate a Bayesian statistics framework in which times are estimated via prior distributions (not necessarily lognormal as used in this paper). Ostensibly this would allow for any given distribution to be used to set relative time windows and test routes. Notably, a business with repetitive deliveries could benefit from using the same routing for a given period of time to obtain new samples and update the routing accordingly. Additionally, with customer penalty assessment data, customer-specific penalties could be introduced. This would likely result in higher late penalties for larger and/or more demanding clients due to phenomena such as customer loyalty and product obsolescence. Lastly, it would be worth investigating some of the natural extensions to the uncapacitated VRP, such as putting capacities on trucks or implementing precedence constraints as in pickup-and-delivery customers.

The building-block method could be enhanced by adding a search procedure to the current heuristic. Presently, it sorts the customer visits into ascending order by the beginning of the on-

time arrival window and assigns the earliest available truck to each visit in sequence. The production of sub-optimal solutions has been demonstrated preliminarily by taking specific realizations of the problem settings and finding solutions by both the heuristic and explicit optimization using a mixed-integer linear programming (MILP) representation of the problem. The MILP is almost always able to find a better solution than does the heuristic. (However, the MILP formulation can only be applied to small-scale problems.) One way to improve the solution provided by the heuristic is to link it to a search procedure (e.g., a genetic algorithm, tabu search, or simulated annealing).

REFERENCES

Agin, N. I., & Cullen, D. E. (1975). Algorithm For Transportation Routing And Vehicle Loading. *Logistics*, pp. 1-20.

Androutsopoulos, K. N., & Zografos, K. G. (2012). A bi-objective time-dependent vehicle routing and scheduling problem for hazardous materials distribution. *EURO Journal on Transportation and Logistics*, *1*(1-2), pp. 157-183.

Baker, E. K., & Rushinek, S. F. (1982). Large Scale Implementation of a Time Oriented Vehicle Scheduling Model, pp. 115.

Ball, M. O., Bodin, L. D., & Greenberg, J. (1985). Enhancements to the RUCUS-II crew scheduling system. In J. M. Rousseau (Ed.), Proceedings of the Second International Workshop on Computer-Aided Scheduling of Public Transport (pp. 279–293). North-Holland.

Bell, M. G. H. (2004). Games, heuristics, and risk averseness in vehicle routing problems. *Journal of Urban Planning and Development, 130*(1), pp. 37-41.

Bertsimas, D., Chervi, P., Peterson, M., Golden, B. L., Laporte, G., Taillard, E. D., . . . Badeau, P. (1995). Computational Approaches to Stochastic Vehicle Routing Problems. Institute for Operations Research and the Management Sciences.

Bodin, L., Golden, B., Assad, A., & Ball, M. (1981). The State of Art in the Routing and Scheduling of Vehicles and Crews, pp. 413.

Bodin, L. D., & Jursh, S. J. (1978). A Computer-Assisted System for Routing and Scheduling of Street Sweepers. *Operations Research, 26*(4), pp. 525-537.

Bruck, H. (1969). CARS (Computer Aided Routing System) A Prototype Dial-A-Bus System.

Buxey, G. M. (1979). The Vehicle Scheduling Problem and Monte Carlo Simulation. *Journal of the Operational Research Society, 30*(6), pp. 563-573.

Caccetta, L., Alameen, M., & Abdul-Niby, M. (2013). An improved clarke and wright algorithm to solve the capacitated vehicle routing problem. Engineering, Technology & Applied Science Research, *3*(2), pp. 413.

Campbell, A. M., & Savelsbergh, M. (2004). Efficient Insertion Heuristics for Vehicle Routing and Scheduling Problems. *Transportation Science, 38*(3), pp. 369-378.

Clarke, G., & Wright, J. W. (1964). Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research, 12*(4), pp. 568-581.

Cordeau, J.-F. (2006). A branch-and-cut algorithm for the dial-a-ride problem. *Operations Research, 54*(3), pp. 573-586.

Cordeau, J.-F., & Laporte, G. (2003). A tabu search heuristic for the static multi-vehicle dial-a-ride problem. *Transportation Research Part B: Methodological, 37*(6), pp. 579-594. doi: 10.1016/s0191-2615(02)00045-0.

Dantzig, G. B., & Ramser, J. H. (1959). The truck dispatching problem. *Management science, 6*(1), pp. 80-91.

Deleuw, Cather, & Company. (1971). Tri-City Transportation Needs Study: Analysis of Flexible Transit Service Conceprs and Plans.

Dessouky, M., Ioannou, P., & Jula, H. (2004). A Novel Approach to Routing and Dispatching Trucks Based on Partial Information in a Dynamic Environment, pp. 49.

Field, J. F. (1976). *Rucus: an overview implementation and operation of a run cutting and scheduling system for buses*.

Fielding, G. J. (1977). Shared-Ride Taxi Computer Control System Requirements Study, pp. 53.

Garcia, B. L., & Arunapuram, S. (1993). Development of Techniques for Tabu Research for the Vehicle Routing Proble with Time Windows.

Ghoseiri, K., Ghannadpour, S. F., & Seifi, A. (2010). *Locomotive Routing and Scheduling Problem with Fuzzy Time Windows*.

Goeddel, D. L. (1975). An examination of the Run Cutting and Scheduling (RUCUS) system - A case analysis. *In Automated Techniques for Scheduling of Vehicle Operaors for Urban Public Transportation Services.* Palmer House, Chicago.

Hejazi, B., & Haghani, A. (2009). *Dynamic Decision Making for Less-Than-Truckload Trucking Operations Using Path-Based Network Partitioning*.

Hinds, D. H. (1979). RUCUS: A Comprehensive Status Report and Assessment. *Transit Journal*, pp. 17-34.

Howson, L. L., & Heathington, K. W. (1970). Algorithms for Routing and Scheduling in Demand-Responsive Transportaiton Systems. *Highway Research Record*.

Kanturska, U., Trozzi, V., & Bell, M. G. H. (2013). Scheduled Hyperpath: A Strategy for Reliable Routing and Scheduling of Deliveries in Time-Dependent Networks with Random Delays. *Transportation Research Record: Journal of the Transportation Research Board, 2378*, pp. 99-109.

Kim, S. (2003). Optimal Vehicle Routing and Scheduling with Real-Time Traffic Information, pp. xi - 111.

Laporte, G. (1992). The vehicle routing problem: An overview of exact and approximate algorithms. *European Journal of Operational Research*, *59*(3), pp. 345–358. https://doi.org/10.1016/0377-2217(92)90192-C.

Laporte, G., Louveaux, F., & Mercure, H. (1992). The vehicle routing problem with stochastic travel times. *Transportation Science*, *26*(3), pp. 161–170.

Lin, S.-W., Yu, V. F., & Chou, S.-Y. (2011). A simulated annealing heuristic for the truck and trailer routing problem with time windows. *Expert Systems with Applications, 38*(12), 15244-15252.

List, G.F., and M.A. Turnquist, Prototype Crew Assignment Module, working paper, including software, prepared for Sandia National Laboratories, December 1993.

List, G. F., Wood, B., Nozick, L. K., Turnquist, M. A., Jones, D. A., Kjeldgaard, E. A., & Lawton, C. R. (2003). Robust optimization for fleet planning under uncertainty. *Transportation Research Part E: Logistics and Transportation Review*, *39*(3), pp. 209–227. https://doi.org/10.1016/S1366-5545(02)00026-1.

List, G., E. Williams, J. Addison, and A. Morsali, 2017. Improving the Efficiency and Reliability of Freight Transportation, prepared for the National Transportation Center at Maryland, Project NTC2014-SU-R-06.

Little Inc, A. D. (1971). Economic Impact of Freight Car Shortages. Executive Summary, pp. 8.

Meng, Q., Lee, D.-H., & Cheu, R. L. (2005). Multiobjective Vehicle Routing and Scheduling Problem with Time Window Constraints in Hazardous Material Transportation. *Journal of Transportation Engineering, 131*(9), pp. 699-707.

Mitrovic-Minic, S., & Laporte, G. (2004). Waiting Strategies for the Dynamic Pickup and Delivery Problem with Time Windows. *Transportation Research Part B: Methodological, 38*(7), pp. 635-655.

Nussbaum, E., Rebibo, K. K., & Wilhelm, E. (1975). RUCUS (Run Cutting and Scheduling Implementation Manual). pp. 198.

Potvin, J.-y. (2007). Evolutionary Algorithms for Vehicle Routing. (November).

Powell, W. B. (1988). A Comparative Review of Alternative Algorithms for the Dynamic Vehicle Allocation Problem. *Vehicle Routing*, pp. 249 - 291.

.

Ronen, D. (2002). Cargo Ships Routing and Scheduling: Survey of Models and Problems. *Maritime Transport*, pp. 3-10.

Roos, D. (1971). *Dial-A-Bus System Feasibility.*

Roos, D., & Porter, E. H. (1971). *Dial-A-Ride: Urban Mass Transportation Demonstration Project*, pp. 35.

Roos, D., & Wilson, D. G. (1971). *Dial-A-Ride: An Overview of a New Demand-Responsive Transportation System.*

Slater, A. (2002). Specification for a Dynamic Vehicle Routing and Scheduling System.. *International Journal of Transport Management. 1*(1), pp. 29-40.

Taniguchi, E., & Nakanishi, M. (2003). ITS Based Dynamic Vehicle Routing and Scheduling with Real Time Traffic Information.. *International Journal of ITS Research, 1*(1), pp. 49-60.

Taniguchi, E., Yamada, T., & Tamagawa, D. (1999, 1999). *Modelling Advanced Routing and Scheudling of Urban Pickup/Delivery Trucks.*

Taniguchi, E., Yamada, T., & Tamaishi, M. (2001). *ITS Based Dynamic Vehicle Routing and Scheduling.*

Vidal, T., Crainic, T. G., Gendreau, M., Lahrichi, N., & Rei, W. (2012). A hybrid genetic algorithm for multidepot and periodic vehicle routing problems. *Operations Research, 60*(3), pp. 611-624.

Weise, T., Podlich, A., & Gorldt, C. (2009). Solving real-world vehicle routing problems with evolutionary algorithms *Natural Intelligence for Scheduling, Planning and Packing Problems*, pp. 29-53.

Wren, A., & Holliday, A. (1972). Computer Scheduling of Vehicles from One or More Depots to a Number of Delivery Points. *Operational Research Quarterly (1970-1977), 23*(3), pp. 333–344. https://doi.org/10.2307/3007888.

Wren, A., & Rousseau, J.-M. (1995). Bus Driver Scheduling — An Overview. In Computer-Aided Transit Scheduling, *430*, Springer, Berlin, Heidelberg, pp. 173–187. https://doi.org/10.1007/978-3-642-57762-8_12.

Xu, W., He, S., Song, R., & Li, J. (2009). Reliability based assignment in stochastic-flow freight network. *Applied Mathematics and Computation, 211*(1), pp. 85-94. doi: 10.1016/j.amc.2009.01.024.

Yamada, T., Yoshimura, Y., & Mori, K. (2004, 2004). *Road Network reliability Analysis Using Vehicle Routing and Scheduling Procedures*.

Zografos, K. G., & Androutsopoulos, K. N. (2002). Heuristic Algorithms for Solving Hazardous Materials Logistical Problems. *Transportation Research Record, 1783*, pp. 158-166.