MICHIGAN OHIO UNIVERSITY TRANSPORTATION CENTER

Alternate energy and system mobility to stimulate economic development.

Report No: MIOH UTC SC2p5 2010-Final

ENABLING CONGESTION AVOIDANCE AND REDUCTION IN THE MICHIGAN-OHIO TRANSPORTATION NETWORK TO IMPROVE SUPPLY CHAIN EFFICIENCY: FREIGHT ATIS

FINAL REPORT

PROJECT TEAM

Dr. Ratna Babu Chinnam
Dr. Alper Murat
Industrial & Manufacturing Engineering
Wayne State University
4815 Fourth Street
Detroit, Michigan 48202, USA

Dr. Gregory Ulferts
School of Business Administration
University of Detroit Mercy
4001 W. McNichols Rd
Detroit, Michigan 48221, USA

Report No: MIOH UTC SC2p5 2010-Final
FINAL REPORT

Developed By: **Ratna Babu Chinnam, Assoc. Professor**
Principal Investigator, WSU
r_chinnam@wayne.edu
313-577-4846

Alper Murat, Asst. Professor
Investigator, WSU
amurat@wayne.edu
313-577-4846

With Contributions By: **Gregory Ulferts, Professor**
Co-Principal Investigator, UDM
ulfertgw@udmercy.edu
313-993-1219

SPONSORS

This is a Michigan Ohio University Transportation Center project supported by the U.S. Department of Transportation, Wayne State University and the University of Detroit Mercy.

ACKNOWLEDGEMENT

The Project Team would like to acknowledge support from the Michigan Department of Transportation (MDOT), Ford MP&L, UPS and C.H. Robinson during the course of this research.

DISCLAIMER

ABSTRACT

We consider dynamic vehicle routing under milk-run tours with time windows in congested transportation networks for just-in-time (JIT) production. The arc travel times are considered stochastic and time-dependent. The problem integrates TSP with dynamic routing to find a static yet robust recurring tour of a given set of sites (i.e., DC and suppliers) while dynamically routing the vehicle between site visits. The static tour is motivated by the fact that tours cannot be changed on a regular basis (e.g., daily or even weekly) for milk-run pickup and delivery in routine JIT production. We allow network arcs to experience recurrent congestion, leading to stochastic and time-dependent travel times and requiring dynamic routing decisions. While the tour cannot be changed, we dynamically route the vehicle between pair of sites using real-time traffic information (e.g. speeds) from Intelligent Transportation System (ITS) sources to improve delivery performance. Traffic dynamics for individual arcs are modeled with congestion states and state transitions based on time-dependent Markov chains. Based on vehicle location, time of day, and current and projected network congestion states, we generate dynamic routing policies for every pair of sites using a stochastic dynamic programming formulation. The dynamic routing policies are then simulated to find travel time distributions for each pair of sites. These time-dependent stochastic travel time distributions are used to build the robust recurring tour using an efficient stochastic forward dynamic programming formulation. Results are very promising when the algorithms are tested in a simulated network of Southeast-Michigan freeways using historical traffic data from the Michigan ITS Center and Traffic.com.

TABLE OF CONTENTS

PAGE
ABSTRACT iii
LIST OF FIGURES iv
LIST OF TABLES v
1. EXECUTIVE SUMMARY 1
2. ACTION PLAN FOR RESEARCH 2
3. INTRODUCTION 3
4. OBJECTIVE 4
5. SCOPE 4
6. LITERATURE SURVEY 4
7. METHODOLOGY: STD-TSP WITH DYNAMIC ROUTING 6
7.1. Dynamic Routing with Real-time Traffic Information 7
7.1.1. Congestion Modeling 7
7.1.2. DP Formulation for Dynamic Routing 8
7.1.3. Estimating Travel Time Distributions between Sites 10
7.2. Dynamic Programming for STD-TSP 11
7.2.1. STD-TSP with Time Windows 13
8. DISCUSSION OF RESULTS: EXPERIMENTAL STUDY 15
8.1. Estimating Congestion States 16
8.2. Estimating Travel Time Distributions between Sites 17
8.3. Building STD-TSP Tours 18
8.4. Evaluation of STD-TSP Tours with Time Windows 20
9. CONCLUSIONS 23
10. RECOMMENDATIONS FOR FURTHER RESEARCH 24
11. RECOMMENDATIONS FOR IMPLEMENTATION 24
BIBLIOGRAPHY 26

LIST OF FIGURES

PAGE
Figure 1. Southeast Michigan road network considered for experimental study 16
Figure 2. For arc (7,8) (a) raw traffic speeds for May, June, and July 2009 weekdays (b) mean (mph) and standard deviations (mph) of speeds by time of the day with time interval resolution of 15 minutes 17
Figure 3. For arc (7, 8) (a) congestion state-transition probabilities: α, congested to congested transition; β, uncongested to uncongested transition probability (b) mean travel time(min.) for congested and uncongested congestion states 17
Figure 4. The tour 1's (a) mean tour travel time (trip time - service times), (b) standard deviation for 48 starting times during the day for static and dynamic policies. 19
Figure 5. Recommended framework for data communication and decision support integration. 25

LIST OF TABLES

PAGE

Table 1 : Tours, tours mean travel times and standard deviations at two departure times for the static and dynamic policies. 19
Table 2 : Mean of tour trip times at the beginning of shifts based on *static* and *dynamic* policies (without time windows). 20
Table 3 : Standard deviations of tour trip times at the beginning of shifts based on *static* and *dynamic* policies (without time windows). 20
Table 4 : Mean-variance objectives of tour trip times at the beginning of shifts based on *static* and *dynamic* policies (without time windows). 21
Table 5 : Simulated mean arrival times (with time windows) to the sites in the sequence of *tour 1* based on *static* and *dynamic* policies. 22
Table 6 : On-time delivery performances (in percentages) of the policies with time windows. . 22
Table 7 : On-time delivery performances (in percentages) and average waiting times (in minutes) for dynamic policy when setting time windows of dynamic policy as the time windows of static policy. 23

1. EXECUTIVE SUMMARY

In this research, we addressed the problem of planning milk-run tours for JIT production subject to hard time windows in congested road networks. We modeled the milk-run tours as a Traveling Salesman Problem (TSP) with hard time windows. The road network congestion is represented through random network arc travel times and time-dependent congestion states.

The classical TSP is concerned with finding the least cost tour that visits each site exactly once given the set of sites. The travel between any pair of sites is a *path* which can be a fixed sequence of arcs or be determined through a dynamic policy. The cost of travel between pairs of sites can be measured in time, distance or a function of both, be deterministic or probabilistic, and be time-dependent or independent. We consider a TSP with hard time windows under stochastic time-dependent (STD) arc travel times. All preceding work assumes that the path travel cost between pairs of sites is either deterministic or stochastic with a known probability distribution. In our network setting, the path travel times are both stochastic and time-dependent. We modeled the recurrent congestion by defining congestion states of arcs based on historical ITS traffic data using Gaussian Mixture Model (GMM) based clustering. The changes in arc congestion states represent the traffic dynamics and are modeled as Markov processes. Accordingly, the optimal dynamic routing problem is then cast as a Markov decision process (MDP) where the states space consists of the position of the vehicle, the time of the day, and the current and projected congestion states of arcs with limited look ahead. We identified the paths' optimal dynamic routing policies (DRP) by solving a stochastic dynamic programming formulation for each pair of sites. By simulating the optimal DRPs, we then estimated the travel time distributions between every pair of sites and used these distributions to determine the optimal TSP tour by solving a stochastic dynamic programming formulation for TSP. We obtained the optimal TSP tour as the most robust tour based on a mean-variance objective based on the trip time which accounts for the transportation cost and service level (i.e., on-time performance) trade-offs in JIT production systems.

We evaluated the developed approach on a real case study application using the road network from Southeast Michigan. The case study corresponded to an automotive JIT production system where an OEM's DC is replenished by milk-run pickup and deliveries from multiple suppliers. Accordingly, the study road network covered major freeways and highways in and around the Detroit metropolitan area. We compared the selected robust tours with those of the static routing policy between pair of sites and quantified the benefits of using dynamic policy. Without time windows for both static and dynamic policies, the case study implementation results showed that the dynamic policy saves 8.1% in trip duration on the average and reduces standard deviation of trip duration by 21.6% on the average. With the time windows set according to the expected site arrival times, we showed that the on-time delivery performance can be increased up to 8% for a site and up to 4% for a tour by using dynamic routing policy. In a subsequent experiment, we demonstrated the potential to further increase the on-time performance by setting the time windows of dynamic routing policy according to those of the static policy. Our case study results indicate that using dynamic routing policy between milk run visit not only decreases transportation cost (measured by trip time), but also increases the delivery service level performance (measured by on-time delivery).

2. ACTION PLAN FOR RESEARCH

Milk-run deliveries are critical to some of the logistics companies and our partners. The routing algorithms based on point to point routing are not capable of supporting milk-run based deliveries with and without time-windows. Hence we executed the following steps to develop dynamic routing models and algorithms for milk runs.

We approached the data collection from multiple directions. On the ***network structure*** (network topology, design parameters, arc characteristics) side, we developed a network represents freeway and highways of the South East-Michigan. We test our dynamic routing model on this network. For Southeast-Michigan corridor ***arc velocity data***, we have collaborated with the MITS Center and Traffic.com. We received data representing several months of traffic flow (such as velocity, occupancy) for the southeast Michigan highways from them. We have analyzed this data to improve the quality of the models being developed for dynamic vehicle routing decision support when operating with access to Advanced Traveler Information Systems (ATIS) information. For instance, through our analyses, we identified the need for representing each arc's congestion with a different number of states (and not force all arcs to be modeled with two states – i.e., congested and uncongested). Accordingly, we have refined the recurrent congestion state modeling by employing the Gaussian Mixture Model clustering method for automated detection of number of states and state velocity thresholds. Extensive evaluations of our dynamic routing algorithms on hypothetical networks revealed significant reductions in trip completion times in comparison with deterministic algorithms and static stochastic algorithms that do not account for recurring congestion information.

We developed the procedure to select the best tour that visits every customer in the desired set and return to the depot in minimum expected time. We defined a robust tour objective to select the best tour. This robust tour objective captured the tradeoff between transportation efficiency and on-time delivery service level. We used a sequential method to select the robust tour. First, we determined the travel time distributions between every pair of sites. Second, we found and selected the tour minimizing the mean-variance objective of the trip time. The travel time distributions between sites were estimated through the following steps: (1) Develop a dynamic routing policy between every pair of sites. (2) Estimate the travel time distribution through simulation for every possible departure times. Once the travel time distributions were estimated for every pair of sites at different departure times, we then employed a stochastic time-dependent dynamic programming (STD-DP) to select the robust tour. We then tested our models and algorithms in a set of customer locations from the SE Michigan traffic network, compared the performance differences between typical base-line routing algorithms.

3. INTRODUCTION

Just-in-time (JIT) production requires frequent small-batch pickups and deliveries subject to fixed time windows. Since the shipments are usually less than truck load, the freight carrier planners develop milk-run tours (e.g., a visiting sequence of pickup and delivery sites). In a milk-run tour, for example, the vehicle departs from a distribution center (DC), picks up goods from several supplier sites, and returns to the DC for delivery. In planning milk-run tours, managers also consider *heijunka* (production smoothing or workload leveling) and *muda* (waste) philosophies of JIT production. Whereas the former can be achieved by equally spacing the delivery time windows over the suppliers' operating hours, the latter can be achieved by visiting the supplier sites at an optimal frequency, balancing transportation and inventory costs. The recurrent and non-recurrent congestion on road networks increase the travel time variability thus rendering it difficult to make delivery and pickup visits within the established time windows, which can be as narrow as 15 to 30 minutes (Chen et al. 2003, Groenevelt, 1993). For carriers, as congestion worsens the costs related to travel time (e.g. labor and overtime costs) may outweigh other operating costs (e.g. vehicle miles traveled) (Figliozzi, 2010).

For example, a survey in California found that 85% of trucking companies miss their time window schedules due to road network congestion. Furthermore, 78% of the managers surveyed stated that the time-window schedules for pickup and deliveries force their drivers to operate under congested road network conditions (Golob and Regan, 2003). Some industries allow early or tardy delivery and/or pickups with a penalty (*soft* time windows). However, there are many practical settings (e.g., JIT production) with *hard* time windows where vehicles may pickup or deliver only during fixed times without exception (Cordeau et al., 2000).

The randomness of travel times on arcs may be because of several reasons. Recurrent and/or non-recurrent congestion are the two prime reasons hence we develop delay estimation models for both of these congestion types. We assume the traffic dynamics follows a Markov process. Namely, the state of the next time period depends on only the state of the previous time period. This allows us to model our problem based on Markov decision process (MDP). The state set of the MDP is based on the position of the vehicle, the time of the day and the (recurrent and non-recurrent) congestion states of the arcs.

We define recurrent (peak-time) congestion states based on the average speed of the vehicles, time of the day, and a cut-off speed. The congestion state classes (i.e.: congested, uncongested, etc.) of the roads are determined with historic traffic data from ITS center based on Gaussian Mixture Model (GMM). Since, not all of the network information affects an optimal decision, we assume the arc set of a state such that only the arcs those are close the vehicle affects the decision. We also assume that the traffic data for some of the arcs may not be available.
The contribution of this study is three-fold. First, we developed an integrated methodology for identifying the TSP tours of sites in STD networks where the stochastic path travel times between pairs of pickup and delivery sites are estimated through optimal dynamic routing. Second, we proposed an approach for dynamic routing between pairs of sites in STD networks using the real-time congestion information available from ITS sensor networks. Third, using a real network and data, we simulated the results of the proposed integrated approach and demonstrate the transportation cost and delivery service level improvement based on optimal dynamic routing between sites.

4. OBJECTIVE

The objective of our study is to develop methods for routing vehicles in stochastic road network environments representative of real-world conditions.

In the literature some aspects of this problem have been studied at some level but there does not exist any study that takes into account all aspects of our dynamic routing problem. To the best of our knowledge, there is no earlier study on the dynamic routing for the stochastic time-dependent TSP problem. The objectives of this study are: (1) Developing an integrated methodology for identifying the TSP tours of sites in STD networks. (2) Finding dynamic routing policies between pairs of sites in STD networks that use real-time congestion information.

5. SCOPE

Given an origin and customer set, the traveling salesman problem is to decide which arc to choose at each decision node (customer locations and/or intersections) such that the expected total travel time (or another performance criteria) is minimized while visiting all customers in their specified delivery time windows.

Our most general model is *a non-stationary stochastic time dependent traveling salesman problem with time windows* (STD-TSP-TW). The Traveling Salesman Problem (TSP) [2] is concerned with finding optimal trip (e.g. with the least travel time, distance, or other performance measure) in which the vehicle starts from the depot, visits every customer in a given set, and returns to the depot. If the travel time between two customers or between a customer and the depot depends on not only the distance/travel time between the customers, but also the time of day of departure then it is called *time-dependent TSP* (TD-TSP). The service time at each customer may also depend on the time of day. If the travel times and/or service times are also random values then this lead to another variant of TSP namely, *stochastic TD-TSP* (STD-TSP). Finally, each of the customers may also have imposed time window constraints on delivery time. In literature this is called *STD-TSP with time windows* (STD-TSP-TW). Hence, in the STD-TSP-TW, a vehicle is initially located at the depot, and must serve a number of geographically dispersed customers in a network where travel times are stochastic and time dependent and each customer must be served within a specified time window. The objective is to find the optimum route with minimum total cost of travel and service time.

6. LITERATURE SURVEY

In JIT production systems, the pickup and delivery tours are constructed while accounting for logistics drivers such as leveling the workload and decreasing inventory levels. One approach for determining pickup and delivery tours in JIT systems is the common frequency routing (CFR) method, where the suppliers are grouped into subsets and each subset of suppliers is served in a single tour (Chuah and Yingling, 2005). The CFR method considers scheduling and routing decisions jointly while accounting for transportation and inventory costs. For computational tractability, the CFR method assumes fixed routes and identical visit frequency for suppliers in the same subset. Another approach is the generalized frequency routing (GFR) where a supplier's visit frequency is not required to be the same as other suppliers in the subset (Ohlmann et al., 2010).

One of the goals in scheduling and routing decisions is to achieve production smoothing through uniformly spaced pickup and delivery visits. These “lean” routing studies consider a more general problem (e.g., VRP) than the TSP studied in this paper but assume that the travel times on the transportation network are deterministic and time-independent. Accordingly, our focus was on selecting robust tours for a given subset of suppliers with uniformly spaced hard time windows.

The body of literature to which this study is related is the stochastic time-dependent traveling salesman problem (TSP) with time windows. In the classical TSP, given a set of sites and the cost matrix relating pairs of sites, the goal was to find the shortest tour starting from the origin site, visiting each site exactly once, and returning to the origin site. TSP and its generalization VRP have been studied for more than five decades and a wide variety of exact and heuristic algorithms have been developed (Johnson and McGeoch 1997, Junger et al. 1995, Laporte 2009, Laporte 2010). There are many variants of the classical TSP but we restricted our review to those studies with time-dependent and stochastic travel times. Malandraki and Dial (1996) presented a dynamic programming (DP) procedure and a “restricted” DP procedure that uses the nearest-neighbor heuristic approach to solve the time-dependent TSP (TD-TSP). They modeled the time dependency by discrete step functions such that the planning horizon had a number of different time zones and the travel times differed only at different time zones. Ichoua et al. (2003) recognized the limitation of using such step functions which violates the first-in-first-out (FIFO) principle by causing a later departure time leading to an earlier arrival time if steep speed increases occur. Accordingly, they emphasized the need to explicitly model time-dependent travel times and proposed a model to determine TSP tours in compliance with the FIFO principle.

Another variant of the classical TSP is the TSP with stochastic travel times between sites. This variant is most studied in the more general form of the vehicle routing problem (Laporte et al. 1992, Lambert et al. 1993). Jula et al. (2006) and Chang et al. (2010) studied the stochastic time-dependent TSP with time windows (STD-TSP-TW). Jula et al. (2006) solved the TSP through a dynamic programming approach applied to a reduced state space. They employed two-state space reduction strategies to reduce the computational complexity. Initially they estimated the mean and variance of the arrival time of the vehicle at each site based on the first (or second) order Taylor approximation. In the first strategy, they defined a service level based on the arrival times to sites and eliminated routes that did not satisfy those service levels. The other strategy eliminates states based on expected travel times. Chang et al. (2010) developed a convolution–propagation approach (CPA) to estimate the mean and variance of arrival times at sites assuming the arc travel times are normally distributed. They proposed a heuristic algorithm that uses the n-path relaxation of deterministic TSP in Houck et al., (1980) to solve the problem. Although the TSP problem we considered is similar to those in Jula et al. (2006) and Chang et al. (2010), the travel time distributions between pairs of sites were endogenous in our study. In particular, we integrated the construction of a TSP tour among sites with the road network routing between pairs of sites in the TSP tour. The dynamic routing between sites accounts for the time-dependent stochastic congestion states by using real-time traffic information and by anticipating congestion states with limited look ahead. To the best of our knowledge, there is no prior study proposing and integrating dynamic routing between sites for the stochastic time-dependent TSP problem.

In addition, whereas Jula et al. (2006) and Chang et al. (2010) identified tour(s) with least expected tour times, we selected tour(s) with minimum mean-variance objective of the trip times. Dynamic routing and modeling real time information has mostly been studied in shortest path problem literature. Polychronopoulos and Tsitsiklis (1993) conducted the first study to consider the stochastic temporal dependence of arc costs and suggested using online information en route. They defined the environmental state of nodes that is learned only when the vehicle arrives at the source node. They considered the state changes according to a Markovian process and employed a dynamic programming procedure to determine the optimal DRP. Kim et al. (2005a) studied a similar problem as did Psaraftis and Tsitsiklis (1993) except that the information of all of the arcs was available in real-time. They proposed a dynamic programming (DP) formulation where the state space included the states of all arcs, time, and the current node. They noted that the state space of the proposed formulation became quite large making the problem intractable. They reported substantial cost savings in a computational study based on a Southeast-Michigan road network. To address the intractable state-space issue, Kim et al. (2005b) proposed state space reduction methods. A limitation of Kim et al. (2005a) is the modeling and partitioning of travel speeds for the determination of arc congestion states. They assumed that the joint distribution of velocities from any two consecutive periods followed a single unimodal Gaussian distribution, which did not adequately represent arc travel velocities for arcs that routinely experience multiple congestion states. Moreover, they also employed a fixed velocity threshold *(50 mph)* for all arcs and for all times in partitioning the Gaussian distribution to estimate state-transition probabilities (i.e., transitions between congested and uncongested states). As a result, the value of real-time information was compromised rendering the loss of performance of the DRP. Our dynamic routing approach addressed all of these limitations. The detailed steps of our model are described in the Section 3.1.

7. METHODOLOGY: STD-TSP WITH DYNAMIC ROUTING

The *STD-TSP with dynamic routing problem* is to find a tour of a given set of sites (i.e., DC and supplier) while dynamically routing between sites' visits on a STD network to meet the time windows requirements. It differs from the TSP with stochastic travel times in that the travel time distributions are obtained through dynamic routing on the road network and thus are dependent on the site departure times. We selected the tours based on a robust tour objective. This robust tour objective captured the tradeoff between transportation efficiency and on-time delivery service level.

We used a sequential method to select the robust tour. First, we first determined the travel time distributions between every pair of sites. Second, we found and selected the tour minimizing the mean-variance objective of the trip time. The travel time distributions between sites were estimated through the following steps (See Section 3.1.):

- Develop a dynamic routing policy between every pair of sites.
- Estimate the travel time distribution through simulation for every possible departure times.

Once the travel time distributions were estimated for every pair of sites at different departure times, we then employed a stochastic time-dependent dynamic programming (STD-DP) to select the robust tour (Section 3.2.).

7.1. Dynamic Routing with Real-time Traffic Information

Let $G=(N,A)$ be a directed graph in which N is the set of nodes and $A\subseteq N\times N$ is the set of directed arcs. The (decision) node $n\in N$ represents an intersection where the driver can decide which arc to select next. A directed arc is represented by an ordered pair of nodes $(n,n')\in A$ in which n is called the origin and n' is called the destination of the arc. Given an origin-destination (OD) node pair of sites (DC, supplier), the dynamic routing problem is to decide which arc to choose at each decision node such that the expected total OD travel time is minimized. We denote the origin and destination nodes with n_0 and n_d, respectively. We formulate this problem as a finite horizon Markov decision process (MDP), where the travel time on each arc follows a non-stationary stochastic process. We first describe the modeling of recurrent congestion and then present the stochastic dynamic programming formulation and solution approach.

7.1.1. Congestion Modeling

A directed arc $(n,n')\in A$ is labeled as observed if its real-time traffic data (e.g., velocity) is available through the ITS. An observed arc can be in $r+1\in \mathrm{Z}^{+}$ different states that represent the arc's traffic congestion level at a given time. Let $s_a(t)$ be the congestion state of arc a at time period t, i.e. $s_a(t)=\{\text{Congested at level } i\}=\{i\}$ for $i=1,2,...,r+1$ and be determined as follows:

$$s_a(t)=\left\{i,\text{if } c_a^{i-1}(t)\le v_a(t)<c_a^{i}(t)\right\} \tag{1}$$

where $c_a^i(t)$ denote the cut-off velocity at level *i*. For instance, if there are two congestion levels (e.g., $r+1=2$), then the states will be i.e., $s_a(t)=\{\text{Uncongested}\}=\{0\}$ and $s_a(t)=\{\text{Congested}\}=\{1\}$.

We assume that the state of an arc evolves according to a non-stationary Markov chain. In a network with all arcs observed, $S(t)$ denotes the traffic congestion state vector for the entire network, i.e., $S(t)=\left\{s_1(t),s_2(t),...,s_{|A|}(t)\right\}$ at time t. For presentation clarity, we will suppress (t) in the notation whenever time reference is obvious from the expression. Let the state realization of $S(t)$ be denoted by $s(t)$. We assume that arc states are independent from each other and have the single-stage Markovian property. To estimate the state transitions for each arc, we jointly model the velocities of two consecutive periods Accordingly, the time-dependent single-period state transition probability from state $s_a(t)=i$ to state $s_a(t+1)=j$ is denoted by $P\left\{s_a(t+1)=j \mid s_a(t)=i\right\}=\alpha_a^{ij}(t)$. We estimate the transition probability for arc *a*, $\alpha_a^{ij}(t)$ from the joint velocity distribution as follows:

$$\alpha_a^{ij}(t)=\frac{\left|c_a^{i-1}(t)\le V_a(t)<c_a^{i}(t)\cap c_a^{j-1}(t+1)<V_a(t+1)<c_a^{j}(t+1)\right|}{\left|c_a^{i-1}(t)\le V_a(t)<c_a^{i}(t)\right|} \tag{2}$$

where the $|e|$ operator corresponds to the frequency count of event *e*. Let $TP_a(t,t+1)$ denote the matrix of state transition probabilities from time t to time $t+1$, then, we have $TP_a(t,t+1)=\left[\alpha_a^{ij}(t)\right]_{ij}$. Note that the single-stage Markovian assumption is not restrictive in our approach as we could extend our methods to the multi-stage case by expanding the state space (Bertsekas, 2001). Let the network be in state $S(t)$ at time t, and we want to find the probability of the network state $S(t+\delta)$, where δ is a positive integer number. Given the independence assumption of the arcs' congestion states, this can be formulated as follows:

$$P\left(S(t+\delta) \mid S(t)\right) = \prod_{a=1}^{|A|} P\left(s_a(t+\delta) \mid s_a(t)\right) \tag{3}$$

Then the congestion state transition probability matrix for each arc in δ periods can be found by the Kolmogorov's equation:

$$TP_a(t,t+\delta) = \left[\alpha_a^{ij}(t)\right]_{ij} \times \left[\alpha_a^{ij}(t+1)\right]_{ij} \times \ldots \times \left[\alpha_a^{ij}(t+\delta)\right]_{ij} \tag{4}$$

We assume that the distribution of an arc travel time is Gaussian. We further assume that the arc travel time depends on the congestion state of the arc at the time of departure (equivalent to the arrival time whenever there is no waiting). It can be determined according to the corresponding normal distribution:

$$\delta(t,a,s_a) \sim N\left(\mu(t,a,s_a), \sigma^2(t,a,s_a)\right) \tag{5}$$

where $\delta(t,a,s_a)$ is the travel time; $\mu(t,a,s_a)$ and $\sigma(t,a,s_a)$ are the mean and the standard deviation of the travel time on arc a at time t with congestion state $s_a(t)$. For clarity of notation, we hereafter suppress the arc label from the parameter space wherever it is obvious, i.e. $\delta(t,a,s_a)$ will be referred as $\delta_a(t,s)$.

7.1.2. DP Formulation for Dynamic Routing

The objective of the dynamic routing algorithm is to minimize the expected travel time based on real-time information such as the path originates at node n_0 and ends at node n_d. Let us assume that there is a feasible path between (n_0,n_d) where a path $p=(n_0,..,n_k,..,n_{K-1})$ is defined as the sequence of (decision) nodes such that $a_k \equiv (n_k,n_{k+1}) \in A$, $k=0,..,K-1$ and K is the number of nodes on the path.

We define set $a_k \equiv (n_k,n_{k+1}) \in A$ as the current arcs set of node n_k, denoted with $CrAS(n_k)$. That is, $CrAS(n_k) \equiv \{a_k : a_k \equiv (n_k,n_{k+1}) \in A\}$ is the set of arcs emanating from node n_k. Each node on a path is a decision stage (or epoch) at which a routing decision (which node to select next) is to be made. Let $n_k \in N$ be the location of k^{th} decision stage, t_k is the time at k^{th} decision stage where $t_k \in \{1,...,T\}$ $T > t_{K-1}$. T is an arbitrarily large number and is used to limit the planning horizon for modeling purposes. Note that we are discretizing the planning horizon.

While the optimal dynamic routing policy requires real-time consideration and projection of the traffic states of the complete network, this approach renders the state space prohibitively large. In fact, there is little value in projecting the congestion states well ahead of the current location. This is because the projected information is not different from the long run average steady state probabilities of the arc congestion states.

Hence, an efficient but practical approach would trade off the degree of look-ahead (e.g., the number of arcs to monitor) with the resulting projection accuracy and routing performance. This has been very well illustrated in Kim et al. (2005b). Thus, we limit our look-ahead to a finite number of arcs that can vary by the vehicle location on the network. The selection of the arcs to monitor would depend on factors such as arc lengths, the value of real-time information, and the congestion state transition characteristics of the arcs. For ease of presentation and without loss of generality, we choose to monitor only two arcs ahead of the vehicle location and model the rest of the arcs' congestion states through their steady state probabilities. Accordingly, we define the following two sets for all of the arcs in the network. $ScAS(a_k)$, the successor arc set of arcs a_k, $ScAS(a_k) \equiv \{a_{k+1} : a_{k+1} \equiv (n_{k+1}, n_{k+2}) \in A\}$, i.e., the set of outgoing arcs from the destination node (n_{k+1}) of arc a_k. $PScAS(a_k)$, the post-successor arc set of arc a_k, $PScAS(a_k) \equiv \{a_{k+2} : a_{k+2} \equiv (n_{k+2}, n_{k+3}) \in A\}$ i.e., the set of outgoing arcs from the destination nodes (n_{k+2}) of arcs a_{k+1}.

Since the total path travel time is an additive function of the individual arc travel times on the path plus a penalty function measuring earliness/tardiness of arrival time to the destination node, the dynamic route selection problem can be modeled as a dynamic programming model. The state$\left(n_k, t_k, s_{a_{k+1} \cup a_{k+2}, k}\right)$ of the system at the k th decision stage is denoted by Ω_k. This state vector is composed of the state of the vehicle and network and thus is characterized by the current node (n_k), the current node arrival time (t_k), and $s_{a_{k+1} \cup a_{k+2}, k}$, the congestion state of arcs $a_{k+1} \cup a_{k+2}$ where $\{a_{k+1} : a_{k+1} \in ScAS(a_k)\}$ and $\{a_{k+2} : a_{k+2} \in PScAS(a_k)\}$ at k th decision stage.

The action space for the state Ω_k is the set of current arcs of node n_k, $CrAS(n_k)$. At every decision stage, the trip planner evaluates the alternative arcs based on the remaining expected travel time. The expected travel time at a given node with the selection of an outgoing arc is the summation of expected arc travel time on the arc chosen and the expected travel time of the next node. Let $\boldsymbol{\pi}_{n_0 n_d} = \{\pi_0, \pi_1, ..., \pi_{K-1}\}$ be the dynamic routing policy (DRP) of the trip that is composed of policies for each of the K-1 decision stages. For a given state $\Omega_k = \left(n_k, t_k, s_{a_{k+1} \cup a_{k+2}, k}\right)$, the policy $\pi_k(\Omega_k)$ is a deterministic Markov policy which chooses the outgoing arc from node n_k, i.e., $\pi_k(\Omega_k) = a \in CrAS(n_k)$. Therefore, the expected travel cost for a given policy vector π is as follows:

$$F^{\pi}(\Omega_0) = \underset{\delta_k}{E}\left\{\sum_{k=0}^{K-2} g\left(\Omega_k, \pi_k(\Omega_k), \delta_k\right) + \bar{g}(\Omega_{K-1})\right\} \tag{6}$$

where $\Omega_0 = (n_0, t_0, S_0)$ is the starting state of the system. δ_k is the random travel time at decision stage k, i.e., $\delta_k \equiv \delta\left(t_k, \pi_k(\Omega_k), s_a(t_k)\right)$. $g(\Omega_k, \pi_k(\Omega_k), \delta_k)$ is cost of travel on arc $\pi_k(\Omega_k) = a \in CrAS(n_k)$ at stage k, i.e., if travel cost is a function (ϕ) of the travel time, then $g(\Omega_k, \pi_k(\Omega_k), \delta_k) \equiv \phi(\delta_k)$ and $\bar{g}(\Omega_{K-1})$ is terminal cost of earliness/tardiness of arrival time to the destination node under state Ω_{K-1}. Then, the minimum expected travel time can be found by minimizing $F(\Omega_0)$ over the policy vector π as follows:

$$F^*(\Omega_0) = \min_{\boldsymbol{\pi}_{n_0 n_d} = \{\pi_0, \pi_1, \ldots, \pi_{K-1}\}} F(\Omega_0) \tag{7}$$

The corresponding optimal policy is then:

$$\boldsymbol{\pi}^*_{n_0 n_d} = \arg\min_{\boldsymbol{\pi}_{n_0 n_d} = \{\pi_0, \pi_1, \ldots, \pi_{K-1}\}} F(\Omega_0) \tag{8}$$

Hence, the Bellman's cost-to-go equation for the dynamic programming model can be expressed as follows (Bertsekas, 2001):

$$F^*(\Omega_k) = \min_{\pi_k} \underset{\delta_k}{E} \left\{ g(\Omega_k, \pi_k(\Omega_k), \delta_k) + F^*(\Omega_{k+1}) \right\} \tag{9}$$

For a given policy $\pi_k(\Omega_k)$, we can re-express the cost-to-go function by writing the expectation in the following explicit form:

$$\begin{aligned} F(\Omega_k \mid a_k) = \sum_{\delta_k} P(\delta_k \mid \Omega_k, a_k) \Big[& g(\Omega_k, a_k, \delta_k) \\ & + \sum_{s_{a_{k+1},k+1}} P\left(s_{a_{k+1},k+1}(t_{k+1}) \mid s_{a_{k+1},k}(t_k)\right) \sum_{s_{a_{k+2},k+1}} P\left(s_{a_{k+2},k+1}(t_{k+1})\right) F(\Omega_{k+1}) \Big] \end{aligned} \tag{10}$$

where $P(\delta_k \mid \Omega_k, a_k)$ is the probability of travelling arc a_k in δ_k periods. $P\left(s_{a_{k+2},k+1}(t_{k+1})\right)$ is the long run probability of arc $a_{k+2} : a_{k+2} \in PScAS(a_k)$ being in state $s_{a_{k+2},k+1}$ in stage $k+1$. This probability can be calculated from the historical frequency of a state for a given arc and time.

We used the backward dynamic programming algorithm to solve $F^*(\Omega_k)$, $k = K-1, K-2, .., 0$. In the backward induction, we initialize the final decision epoch such that, $\Omega_{K-1} = (n_{K-1}, t_{K-1}, s_{K-1})$, n_{K-1} is the destination node, and $F(\Omega_{K-1}) = 0$ if $t_{K-1} \leq T$. Accordingly, a penalty cost is accrued whenever there is delivery tardiness, e.g., $t_{K-1} > T$. Note that $s_{K-1} = \varnothing$, since the destination node does not have any current and successor arc states, e.g. the travel terminates at the destination node.

7.1.3. Estimating Travel Time Distributions between Sites

Given a pair of sites (DC, supplier), origin $j \in M$ and destination $k \in M$, we solve the dynamic programming formulation in preceding section for all feasible departure times from *j* and obtain the optimal routing policy, $\boldsymbol{\pi}_{jk}$, for each departure time alternative. Next, for each departure time alternative (t_j), we sample a congestion state $s(t_j)$ for current and successor arcs of *j*, and simulate the policy corresponding to the sample state $\Omega = \left(j, t_j, s(t_j)\right)$. Note that the sampling probabilities of the congestion state $s(t_j)$ are based on the steady-state probabilities of the states of current and successor arcs of *j*. Following sufficient sampling for t_j, we estimate the distribution of the mean travel times obtained by simulating corresponding policies for each sampled state Ω. We then calculate the expectation and variance of travel time from *j* to *k* at time t_j and respectively denote them with $E\left[\delta_{jk}\left(t_j\right)\right]$ and $Var\left(\delta_{jk}\left(t_j\right)\right)$. Note that, with slight abuse of notation, $\delta_{jk}\left(t_j\right)$ corresponds to the random travel time between *j* and *k* departing at t_j.

7.2. Dynamic Programming for STD-TSP

In this section, we describe the stochastic time-dependent dynamic programming (STD-DP) approach for selecting a robust tour of a given set of sites (i.e., DC and supplier) while dynamically routing between sites' visits to meet the time windows requirements. The time window requirements are strict (e.g., hard time windows) and each site has a deterministic service time for loading/unloading. This STD-DP approach integrates and builds on the results of earlier studies. Specifically it integrates the stochastic tour search procedure from Malandraki and Dial (1996) and Jula et al. (2006) and the convolution idea from Chang et al. (2010). However, the proposed STD-DP approach uses the travel time distributions obtained in the preceding section by dynamically routing on the road network. Further, the approach selects the most robust tour by trading off the expected duration of the tour with its variability as follows:

$$TC_{00} = E\left[T\left(M,\tau,0\right)\right] + b\sqrt{Var\left(T\left(M,\tau,0\right)\right)}\,, \tag{11}$$

where, τ is the TSP tour, $E\left[T\left(M,\tau,0\right)\right]$ and $Var\left(T\left(M,\tau,0\right)\right)$ are the expected and variance of the round trip duration departing from site *0* (DC) at time t_0, visiting all sites in M once, and returning back to site *0* (DC); b is a user defined risk-parameter for balancing the transportation efficiency with on-time delivery performance.

We first describe the STD-DP approach without the time-windows and present its extension to time window case in Section 3.2.1. There are *m*-1 sites (other than the DC, assuming the vehicle at the DC) to be visited, represented by nodes $1,...,m-1 \in M$. Let $\left(C,k\right) \subseteq M / \{0\}$ be an unordered set of visited sites where $k \in C$ is the last visited site. Define *partial tour* τ as a tour that starts from the DC, visits all sites in $\left(C,k\right)$ only once and ends the tour at site k. Note that there may be more than one partial tour corresponding to set $\left(C,k\right)$ and we denote the set of partial tours with $\tau \in \Gamma\left(C,k\right)$. For brevity, we do not repeat the membership of partial tours in the remainder and assume $\left(C,\tau,k\right)$ implies $\tau \in \Gamma\left(C,k\right)$. Let $T\left(C,\tau,k\right)$ be the random variable of arrival time at site k taking the partial tour τ of set $\left(C,k\right)$ after departing site *0* at time t_0. Let also $E\left[T\left(C,\tau,k\right)\right]$ and $Var\left(T\left(C,\tau,k\right)\right)$ be the mean and variance of arrival time to site *k*, $T\left(C,k\right)$ after taking the partial tour τ, respectively.

Step 1. Initialize: For all $\left|\left(C,k\right)\right| = 1$ where $\left(C,k\right) = \{k\},\, k \in M / \{0\}$, we initialize $E\left[T\left(C,\tau,k\right)\right] = T(0) + s_0 + E\left[\delta_{0k}\left(t_0\right)\right]$ and $Var\left(T\left(C,\tau,k\right)\right) = Var\left(\delta_{0k}\left(t_0\right)\right)$, where $T(0)$ is the arrival time to the site *0* (DC), s_0 is the service (e.g., loading/unloading) time at the site 0, and $E\left[\delta_{0k}\left(t_0\right)\right]$ is the expected travel time from site 0 to site k as a function of the departure time, t_0. Note that the expectation $E\left[\delta_{0k}\left(t_0\right)\right]$ is over the congestion states of current and successor arcs of site *0*.

Step 2. Main: For all $\left|(C,k)\right|>1$, there are partial tours of set (C,k), where we visit k, $k\in M/\{0,j\}$ immediately after j (for all $j\in C/\{k\}$). The mean and variance $T(C,\tau,k)$ for the partial tour τ is calculated through the following convolution propagation approach adapted from Chang et al. (2010):

$$E\left[T(C,\tau,k)\right]=E\left[T(C,\tau,j)\right]+s_j+\sum_{t_j}E\left[\delta_{jk}\left(t_j\right)\right]p_{t_j}, \tag{12}$$

$$\begin{aligned}Var\left(T(C,\tau,k)\right)=&Var\left(T(C,\tau,j)\right)+\sum_{t_j}p_{t_j}\sigma_{t_j}^2+\sum_{t_j}p_{t_j}E\left[\delta_{jk}\left(t_j\right)\right]^2-\left[\sum_{t_j}p_{t_j}E\left[\delta_{jk}\left(t_j\right)\right]\right]^2\\&-2\sum_{t_j}E\left[\delta_{jk}\left(t_j\right)\right]\sqrt{Var\left(T(C,\tau,j)\right)}\left(\varphi_{z_{t_j}}-\varphi_{z_{t_j-1}}\right),\end{aligned} \tag{13}$$

where s_j is the deterministic service time at site j; $\delta_{jk}\left(t_j\right)$ is the travel time from site j to site k at the departure time $t_j=T(C,\tau,j)+s_j$; p_{t_j} is the probability of departing at time t_j from node j. Note that the expectation $E\left[\delta_{jk}\left(t_j\right)\right]$ is over the congestion states of current and successor arcs of site *j*. Let $z_{t_j}=\frac{t_j-E\left[T(C,\tau,j)\right]-s_j}{\sqrt{Var\left(T(C,\tau,j)\right)}}$, we calculate p_{t_j} as $p_{t_j}=\Phi\left(z_{t_j}\right)-\Phi\left(z_{t_j-1}\right)$, where $\varphi(\cdot)$ and $\Phi(\cdot)$ are the density and cumulative distribution functions of the standard normal distribution, respectively. Once $T(C,\tau,k)$ is calculated for all $\left|(C,k)\right|>1$, we decrease the number of partial tours under investigation by performing the following partial tour elimination test adapted from Jula et al. (2006).

Dominancy test: There may be more than one *partial tour* for a set (C,k). Let us assume (C,τ_1,k) and (C,τ_2,k) are two partial tours of set (C,k) that cover same sites. We eliminate the partial tour (C,τ_1,k) if $T(C,\tau_2,k)$ dominates $T(C,\tau_1,k)$, e.g., $E\left[T(C,\tau_2,k)\right]\le E\left[T(C,\tau_1,k)\right]$ and $Var\left(T(C,\tau_2,k)\right)\le Var\left(T(C,\tau_1,k)\right)$.

We note that additional partial tour elimination tests based on time windows are described in the next section. After testing all pairs of partial tours, we repeat the main step until $C=M-\{0\}$.

Step 3. Termination: To complete the tour at the site *0* (DC), we set *k*=0 and perform the main step one last time and obtain the expectation and variance of the total tour time $T(C,\tau,0)$ for all remaining tours τ of $(C,0)$ where *C=M*. We calculate the total tour cost as

$TC_{00} = E\left[T(C,\tau,0)\right] + b\sqrt{Var\left(T(C,\tau,0)\right)}$ for each of the remaining tours. We select the tour with minimum cost as the robust tour solution.

7.2.1. STD-TSP with Time Windows

In the preceding section, we presented STD-DP for solving the STD-TSP without time windows. This section extends it to cases with hard time windows. When there is a time window requirement at a site, there are three possible arrival scenarios to that site with regard to the time window: early, late, and on-time arrival. In our model, we allow early arrivals, if earliness is not greater than a pre-specified value, by requiring the vehicle to wait until the beginning of time window. In comparison, we do not allow late arrivals by eliminating those partial tours with the possibility of tardiness greater than a pre-specified probability.

Let us assume the vehicle arrives at site j with a random arrival time of $T(C,\tau,j)$ with partial tour τ and does not violate any time window requirement. Let (e_j, l_j) be the time window at site j, where e_j is the earliest time and l_j is the latest time to start service at site j.

- *Early Arrival:* The vehicle arrival is assumed to be early if probability of arriving later than e_j is less than the early arrival probability $\underline{\gamma}$: $P\left(T(C,\tau,j) \geq e_j\right) \leq \underline{\gamma}$. The vehicle can wait only if $T(C,\tau,j) \geq (e_j - \varepsilon)$, where ε is maximum allowable waiting time at the site; otherwise the vehicle is assumed to be *too early* and the partial tour is then discarded. Note that if a particular vehicle arrival is accepted, then, the start time to service is $\max\left(T(C,\tau,j), e_j\right)$.
- *Late Arrival:* The vehicle arrival is assumed to be late and the partial tour is discarded if probability of arriving later than l_j is greater than the maximum allowable tardiness probability $\overline{\gamma}$: $P\left(T(C,\tau,j) \geq l_j\right) > \overline{\gamma}$.
- *On-time Arrival:* The vehicle arrival is assumed to be on-time and is accepted if both $P\left(T(C,\tau,j) \geq e_j\right) > \underline{\gamma}$ and $P\left(T(C,\tau,j) \geq l_j\right) \leq \overline{\gamma}$.

Given these definitions, $E\left[T(C,\tau,j)\right]$ and $Var\left(T(C,\tau,j)\right)$ in equation (12) and (13) can be calculated with the following formulas:

$$E\left[T(C,\tau,j)\right] = E\left[\max\left(T(C,\tau,j), e_j\right)\right] + s_j \tag{14}$$

$$Var\left(T(C,\tau,j)\right) = E\left[\max\left(T(C,\tau,j), e_j\right)^2\right] - E^2\left[\max\left(T(C,\tau,j), e_j\right)\right] \tag{15}$$

Note that the maximization operator is due to the waiting upon early arrival. For late arrivals, the maximum operator in (14) and (15) does not exist since there is no waiting with late arrivals. In both early and late arrival cases, we eliminate those partial tours according to the corresponding pre-defined parameters ($\underline{\gamma}$, ε, $\overline{\gamma}$).

Note that, different than the stochastic dominance elimination, time window eliminations are used in the initialization step and at the termination step if there are also DC time windows applicable to the tour completion time.

7.2.1.1. Determining Time Windows for a Given Tour

In the preceding section, we described how the STD-DP approach is extended for problems with hard-time windows. In most JIT production systems, the time window requirements affect different parties differently. For instance, the carriers are penalized for late deliveries either by charges associated with contracted service levels or by their reduced ranking as a transportation service supplier. In comparison, early arrivals correspond to lower utilization of assets and drivers. The suppliers (pickup sites), on the other hand, need to stock more safety inventory and allocate more material handling resources if time windows are relaxed (e.g., width of the window is increased). The width of the time windows and their positioning constitute two features of most logistics contracts and are often re-adjusted due to changing production volumes and routes. The time window setting process differs from industry to industry. In JIT environments, it is common that the time windows are set by trucking and/or manufacturer companies according to JIT principles and are usually accepted by the suppliers as part of the sourcing contract. In such a setting, the trucks visit the supplier sites several times per day subject to the tight time windows spaced as much evenly as possible within the supplier's operating hours (even spacing is generally key to supplier efficiency; reduces finished goods inventory levels).

We now describe a procedure for carriers to position the time windows such that the on-time delivery performance is improved. We assume that the width of time windows (w) is determined beforehand by the supplier and manufacturer and they are indifferent to the positioning of the time windows as long as they are uniformly distributed during delivery horizon. The procedure uses the result that the site arrival times follow Gaussian distribution when the arc travel times are also Gaussian (Chang et al., 2010). Therefore, centering the time windows at the expected site arrival times maximizes the on-time delivery performance, if, there is no waiting allowed at the site for early deliveries. This is indeed the case practiced by carriers even if there is some flexibility in early arrival acceptance. Let τ be the selected ordered tour that starts from DC, visits all sites once, and ends at DC. Further let τ_k be the partial tour of τ ending at site *k*. Accordingly, $T(C,\tau_k,k)$ is the random variable of arrival time at site k by following the partial tour τ_k. Let also $E\left[T(C,\tau_k,k)\right]$ and $Var\left(T(C,\tau_k,k)\right)$ be the mean and variance of arrival time $T(C,\tau_k,k)$, respectively.

Procedure for Setting Time Windows:

For $k=1,...,m-1$, Repeat:

- If *k=1,*

  $E\left[T(C,\tau_k,k)\right]=T(0)+s_0+\delta_{0k}(t_0)$ and $Var\left(T(C,\tau_k,k)\right)=Var\left(\delta_{0k}(t_0)\right)$ where $T(0)$ is the arrival time to the site 0 (DC), s_0 is the service time at the site 0, and $\delta_{0k}(t_0)$ is the random travel time from site 0 to site k as a function of the departure time, t_0.

  Else,

  Assume visiting k immediately after j and look up the updated $E\left[T\left(C,\tau_j,j\right)\right]$ from the previous step. Calculate $E\left[T(C,\tau_k,k)\right]$ from (11).

  End.

- Set $e_k=E\left[T(C,\tau_k,k)\right]-w/2$ and $l_k=E\left[T(C,\tau_k,k)\right]+w/2$.

- Update $E\left[T\left(C,\tau_k,k\right)\right]$ and $Var\left(T\left(C,\tau_k,k\right)\right)$ according to equations (14) and (15).

Return.

The above procedure is an iterative procedure where we visit sites according to the tour τ and set time windows for each site one at a time. At each site, we calculate the expected arrival time to that site based on the time windows set at the previously visited sites. We account for the previously set time windows because they affect the site arrival time of the subsequent visited sites through the waiting at early arrivals. Note that the centered placement of time windows is an assumption. It is possible to shift the time windows to the right of the center (expected site arrival time) such that the likelihood of late arrivals decreases. Clearly, this modification is contingent upon the maximum allowable waiting time imposed for early arrivals. In the case of unrestricted waiting, it can be shown that, by shifting the time window to right, one can turn time window constraints into redundant constraints.

8. DISCUSSION OF RESULTS: EXPERIMENTAL STUDY

In this section, we test the proposed methodology on a real case study application using the road network from Southeast Michigan, U.S.A. (Fig. 1). We consider an automotive JIT production system where an OEM's DC is replenished by milk-run pickup and deliveries from multiple suppliers. The case study road network covers major freeways and highways in and around the Detroit metropolitan area. The network has 140 nodes and a total of 492 arcs with 140 observed arcs and 352 unobserved arcs. Real-time traffic data for the observed arcs is collected by the Michigan ITS Center and Traffic.com. In this application, we used data from 66 weekdays of May, June, and July 2009, for the full 24 hours of each day. The raw speed data was aggregated at a resolution of 5 minute intervals. For the experimentation, we increased the resolution of data to one data-point per minute through linear interpolation (see Kim et al., 2005a). Since the collected speed data is averaged across different vehicle classes (i.e., automobile, trucks) and no data was available for individual classes of vehicles, we assumed that the truck being routed could also cruise at the collected average speeds. We implemented all of our algorithms and methods in Matlab 7 and executed them on a Pentium IV machine (with CPU 1.6 GHz and 1024 MB RAM) running Microsoft Windows XP operating system.

Our experimental study is outlined as follows: Section 4.1 describes the estimation and modeling process for recurrent congestion and illustrates through a sample arc of the network. Section 4.2 explains the steps of generating DRPs and estimating travel time distributions between sites. Section 4.3 presents experimental results of identifying and selecting robust STD-TSP tours without time windows and reports savings from employing the dynamic routing policy over the static routing policy between pair of sites. Section 4.4 evaluates the performance of routing policies identified in Section 4.3 after setting the sites' time windows as described in Section 3.2.1.1.

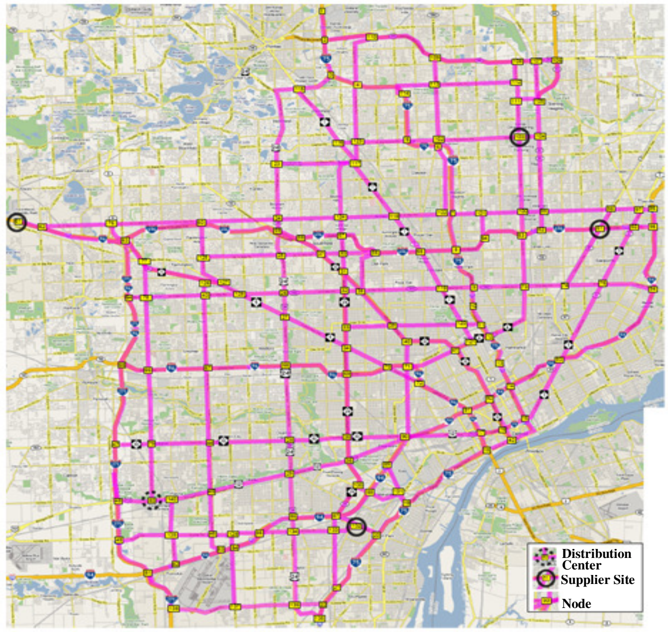

Figure 1. Southeast Michigan Road Network Considered for Experimental Study

8.1. Estimating Congestion States

The proposed dynamic routing algorithm calls for identification of different congestion states, estimation of their state transition rates, and estimation of arc traverse times by time of the day. To better illustrate the modeling of congestion states, we present the data and congestion state identification and separation procedures for an example arc (7, 8). The speed data for arc (7, 8) for the weekdays is illustrated in Figure 2a. The mean and standard deviations of speed for the arc (7, 8) are plotted in (Figure 2b). From Figure 2a and Figure 2b, it can be clearly seen that the traffic speeds follow a non-stationary distribution that vary highly with time of the day.

Given the traffic speed data, we employed the Gaussian Mixture Model (GMM) clustering technique to determine the number of recurrent-congestion states for each arc by time of the day. In particular, we used the greedy learning GMM clustering method of Verbeek et al. (2003) for its computational efficiency and performance. After obtaining the state clusters for each time interval *t*, we first estimate the time-dependent cut-off speeds if GMM yields more than one congestion state at *t*. Next, given cut-off speeds, we then estimate the parameters of the Gaussian distributions for state transitions for congestion state *i* from *t* to *t+1* for all *t,* i.e., ($\boldsymbol{\mu}^i_{t,t+1};\boldsymbol{\Sigma}^i_{t,t+1}$). Applying GMM for arc (7,8), for instance, recommended two clusters of congestion states for almost all time intervals except few. Figure 3a illustrates the transition rates for arc (7, 8) with a 15 minute time interval resolution during the day. Note that, we are using two clusters for arc (7, 8) in all time intervals for presentation purpose (other than increasing computational burden, there are no other consequences). In Figure 3a, the α_t denotes the probability of state transition from congested state to congested state and β_t denotes the probability of state transition from uncongested state to uncongested state. The mean travel time of arc (7, 8) for congested and uncongested traffic states is given in Figure 3b.

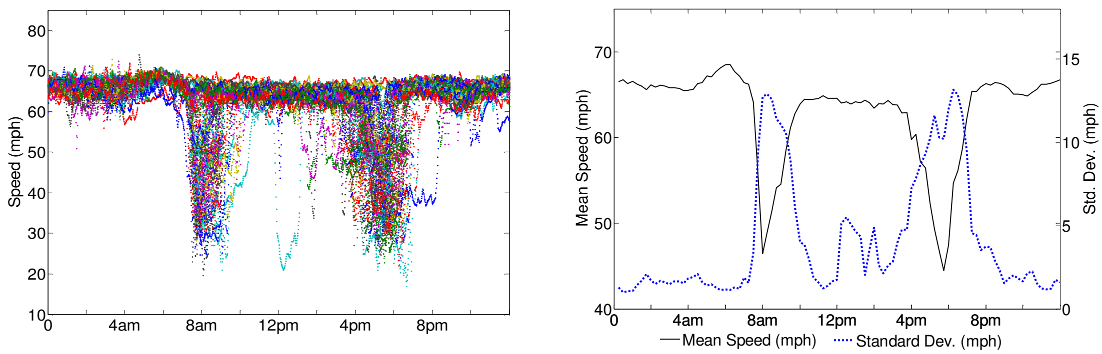

Figure 2. For arc (7,8) (a) raw traffic speeds for May, June, and July 2009 weekdays (b) mean (mph) and standard deviations (mph) of speeds by time of the day with time interval resolution of 15 minutes.

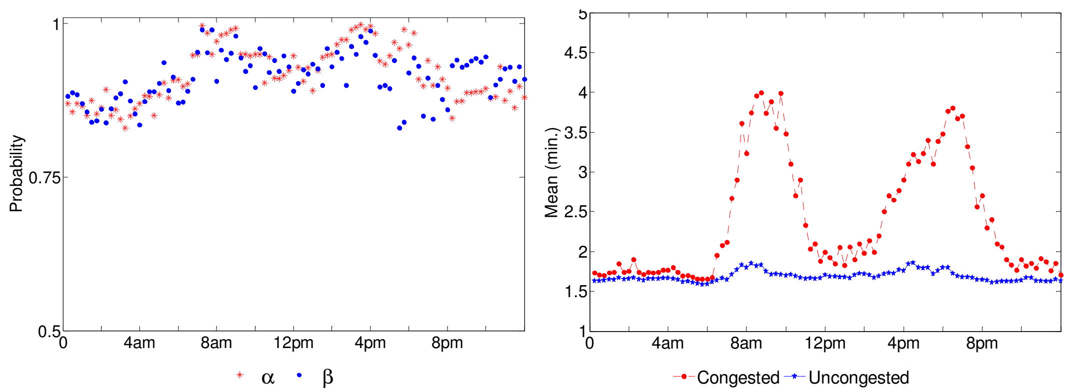

Figure 3. For arc (7, 8) (a) congestion state-transition probabilities: α, congested to congested transition; β, uncongested to uncongested transition probability (b) mean travel time(min.) for congested and uncongested congestion states.

8.2. Estimating Travel Time Distributions between Sites

Using the previous section's results, e.g., time and congestion state dependent distribution of arc travel times and congestion state transition probabilities, we employed the dynamic routing algorithm in Section 3.1.2 to determine the dynamic routing policy $\boldsymbol{\pi}_{jk}$ between every pair of customer sites (j,k) at different departure times. Next, we estimate the travel time distribution between every pair of sites. This can be achieved by simulating the optimal dynamic policies in two different ways: using estimated arc travel time distributions as described in Section 3.1.2. or using the available historical data for 66 weekdays. We choose to use the historical data because of the link interactions and dependencies not captured through the estimation of arc travel time distributions.

In most real transportation networks, the congestion states among the arcs are highly correlated. As a result, independent simulation of each arc's congestion states leads to uncorrelated arc states and might cause incorrect travel time distributions. To avoid such problems, we simulated the network with historical data one day at a time. Specifically, we routed the vehicle from origin site to the destination site; at each decision epoch (e.g. node), the historic arc speed data was used to identify the congestion state and determine which arc to traverse next. We ran the simulations for 66 weekdays of May, June, and July 2009 and obtained 66 samples for all pairs of sites at different departure times.

Although the number of runs was small, we believe it captured the dependency of arc congestion states better and accurately predicts the routing scenario's outcome. In addition, due to weather patterns/seasonality, traffic dynamics do change over extended periods. Hence, it is generally inappropriate to use data from extended periods (e.g., a year) to establish the tours and the dynamic routing policies. For these reasons, it might be best to re-optimize the tour and the dynamic routing policies at regular intervals (e.g., monthly or quarterly).

8.3. Building STD-TSP Tours

In this section, we construct the robust STD-TSP tours using the effective travel time distribution resulting from dynamic routing between every pair of sites (as explained in section 4.2). To quantify the benefits of using a dynamic routing policy, we also identify and select the robust STD-TSP tours with a static routing policy between each pair of sites. In milk-run tours, the number of tour stops in urban areas is generally equal or greater than 5 stops per tour: approximately 5.6 in Denver (Holguin-Veras and Patil, 2005), 6 in Calgary (Hunt and Stefan, 2005), and 6.2 in Amsterdam (Vleugel and Janic, 2004). Our case study application also conforms to these estimates as there are 5 stops (i.e., one DC and four supplier sites). Although there are hundreds of suppliers replenishing the same DC, we only consider the subset of suppliers that were part of the same TSP tour. The determination of such supplier clusters is beyond the scope of this study and is assumed to be performed a priori based various factors (e.g., geographical supplier locations, nature of cargo) as in CFR. There were no pre-established requirements on the sequence of site visits and the truck had enough capacity to visit all sites in a single tour. As in most JIT environments, the time windows in this case study were set by trucking and OEM's logistics division and accepted by the suppliers as part of the sourcing contract. Therefore, we herein consider the case without time windows and then set the time windows for on-time performance in Section 4.4. In the STD-TSP of the case study application, we have node 80 as the DC (origin site) and nodes 61, 103, 51, and 132 as the supplier sites (Fig. 1). Accordingly, there are (5-1)!=24 possible dominated and non-dominated tours. To capture the effect of traffic congestion, we consider 48 trip start times evenly spaced every half an hour and determine tours for each of them separately (Figure 4). We assume all the sites' service times are 15 minutes. Since there are 4 sites other than the DC, the total service time is 60 minutes for each trip. To compare the results we define STD-TSP tours with following two site-to-site routing policies:

1. *STD-TSP tour with static routing policies (**Static policy**):* In practice, almost all commercial logistics software aims to identify TSP tours based on a static path between a pair of sites. First, for a given site pair and departure time, all paths are identified and then their expected path travel times are calculated according to the travel time distributions of paths' arcs. Next, the path with the least expected cost is selected as the static path to be used in the TSP tour. Then, for every trip start time, we select a robust TSP tour by solving STD-TSP using travel time distributions between pairs of sites estimated through the static paths.
2. *STD-TSP tour with dynamic routing policies (**Dynamic policy**):* In this policy, the paths between pairs of customers are dynamic routing policies (DRP). Based on the arc travel time distributions, congestion states and transition probabilities, we first generate DRPs between every pair of sites as described in Section 3.1. Then, these DRPs are simulated to find the site-to-site travel time distributions as described in Section 4.2. Finally, for every trip starting time, the robust TSP tour is selected using the DP algorithm for STD-TSP based on the simulated travel time distributions between pair of sites.

In identifying and selecting the robust tour, we set standard deviation coefficient in the cost function $b = 1.65$ such that the robust tour's trip duration is less than the mean-variance objective 97.5% of the time. We calculated the mean and standard deviations of trip times for all static and dynamic policy tours for evenly spaced 48 trip starting times beginning at 00:00am. The results revealed that 4 out of the 24 possible tours dominate the other tours for all 48 trip starting times for both static and dynamic policies. These *dominant* tours are: *tour 1*: 80→132→103→51→61→80; *tour 2*: 80→132→51→103→61→80; *tour 3*: 80→61→103→51→ 132→80; and *tour 4*: 80→61→51→103→132→80. Among these four tours, *tour 1* is the most selected tour by both static (40 times out of 48) and dynamic (41 times out of 48) policies. We report *tour 1* mean travel time and standard deviations in Figure 4 for every starting time during the day. Note that these results are obtained by simulating the *tour 1* using the historic data (66 weekdays of May, June, and July 2009).

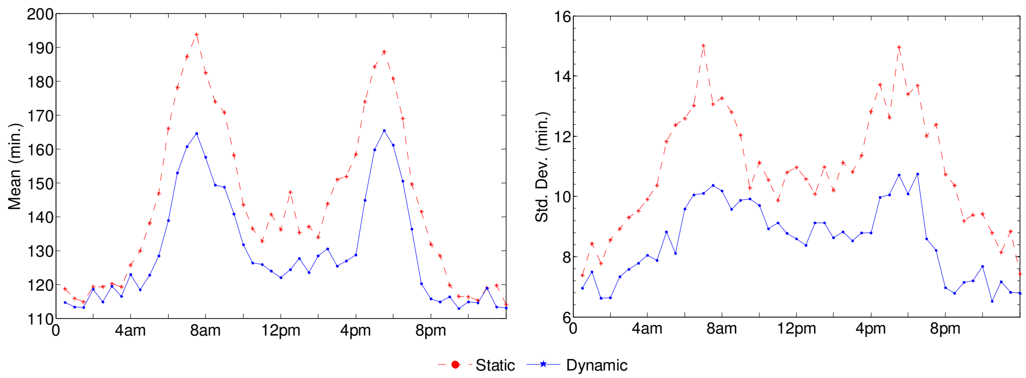

Figure 4. The tour 1's (a) mean tour travel time (trip time - service times), (b) standard deviation for 48 starting times during the day for static and dynamic policies.

As expected, the savings are higher and rather significant during peak traffic times (e.g., around 8:00 and 17:00) and insignificant during uncongested periods. These results clearly illustrate the importance of using dynamic routing between pairs of sites. To further illustrate the savings, we present the selected robust tours and their mean and standard deviation of travel times identified by the two policies for two particular departure times in Table 1.

Table 1. Tours, tours mean travel times and standard deviations at two departure times for the static and dynamic policies

Policy	Robust Tour	Departure Time	Mean Trip Time (min.)	Mean Tour Travel Time (min.)	Std. Dev. of Tour Travel Time (min.)
Static	*tour 1*	7:00	253.8	193.8	13.08
Dynamic	*tour 1*	7:00	224.5	164.5	10.37
Static	*tour 1*	7:30	242.4	182.4	13.27
Dynamic	*tour 2*	7:30	216.1	156.1	10.19

8.4. Evaluation of STD-TSP Tours with Time Windows

In the previous section, we selected the robust tours associated with static and dynamic routing policies across 48 starting times. We originally assumed no time windows. In this case study application, the determination of the TSP tour and the setting of time windows are sequential tasks. Specifically, the carrier first determines the tours for transportation efficiency and then the carrier and OEM's logistics division jointly set the spacing of time windows so as to maximize the on-time delivery performance. Next, we present and compare the trip duration results of using static and dynamic routing policies in a scenario where there are 4 DC replenishment shifts in each day and the shift starting times (ST) are ST= {0:00; 6:00; 12:00; 18:00}. We then present the results after setting time windows.

According to the results in the preceding section, *tour 1* is the most selected tour by both static and dynamic policies across different trip start times. The other robust tours identified are *tours 2, 3,* and *4* in decreasing order of selection frequency. In Table 2 and Table 3, we provide the mean and standard deviation of *trip times* (tour travel time + service times) of these four dominant tours and their associated standard deviations at shift starting times when following *static* and *dynamic* policies between pair of sites, respectively. These results are obtained by simulating the corresponding tours using the historic data (66 weekdays of May, June, and July 2009).

Table 2. Mean of tour trip times at the beginning of shifts based on *static* and *dynamic* policies (without time windows)

		Mean Tour Trip Times											
ST		**0:00**			**6:00**			**12:00**			**18:00**		
Policy		Sta.	Dyn.	Improv.	Sta.	Dyn.	Improv.	Sta.	Dyn.	Improv.	Sta.	Dyn.	Improv.
Tour	**1**	178.7	174.5	2.4%	238.2	212.9	10.6%	207.2	184.4	11.0%	229.1	210.5	8.1%
	2	177.2	174.0	1.8%	241.6	219.3	9.2%	207.8	185.7	10.6%	233.5	207.8	11.0%
	3	181.2	179.0	1.2%	236.4	220.0	6.9%	209.2	189.6	9.4%	237.9	220.1	7.5%
	4	183.6	181.1	1.4%	248.3	224.9	9.4%	205.1	193.5	5.7%	242.6	222.5	8.3%

Table 3. Standard deviations of tour trip times at the beginning of shifts based on *static* and *dynamic* policies (without time windows)

		Standard Deviation of Tour Trip Times											
ST		**0:00**			**6:00**			**12:00**			**18:00**		
Policy		Sta.	Dyn.	Improv.	Sta.	Dyn.	Improv.	Sta.	Dyn.	Improv.	Sta.	Dyn.	Improv.
Tour	**1**	7.8	7.0	10.3%	13.0	10.1	22.8%	10.9	8.4	23.4%	14.1	9.7	31.1%
	2	8.3	7.5	9.5%	13.6	11.2	17.8%	12.4	9.8	20.6%	14.3	10.8	25.0%
	3	7.8	7.7	1.0%	14.5	11.6	20.3%	11.9	10.5	11.9%	14.3	11.1	22.4%
	4	9.8	8.6	12.0%	15.2	12.2	20.3%	12.4	9.6	23.0%	14.8	13.0	12.6%

Table 2 results indicate that the mean tour trip time savings associated with dynamic routing are most in the two congested start times, namely 6:00 and 18:00, which are close to the urban area peak traffic times.

We further note that the savings with start time at 12:00 is also as high as the congested periods (i.e., 6:00 and 18:00). The results in Table 5 for the standard deviation of tour trip times demonstrate the savings in variability similar to those in mean trip times.

The robust tour for each starting time is selected according to the mean-variance objective using the results in Table 4 and Table 5. These mean-variance objectives for the four dominant tours are presented in Table 6 along with that of the selected robust tour in the last row. The selected robust tours corresponding to static and dynamic policies are highlighted in bold for each start time. The dynamic policy's robust tour achieves the most savings over that of the static policy for trips starting at 12:00 and the mean-variance objective savings range from 2.6% to 12.0% with an average of 9.2%. The mean tour trip time savings based on the robust tours range from 1.5% to 11.0% with an average of 8.1% as can be calculated from Table 4. These tour trip duration savings correspond to the improvement in transportation efficiency. Similarly, the savings in the standard deviation of tour trip times based on the robust tours range from 16.5% to 23.7% with an average of 21.6% as can be calculated from Table 5. These savings correspond to the improvement in tour trip time reliability affecting the on-time delivery performance.

Table 4. Mean-variance objectives of tour trip times at the beginning of shifts based on *static* and *dynamic* policies (without time windows)

		Mean-Variance Tour Trip Time Objectives											
ST		**0:00**			**6:00**			**12:00**			**18:00**		
Policy		Sta.	Dyn.	Improv.	Sta.	Dyn.	Improv.	Sta.	Dyn.	Improv.	Sta.	Dyn.	Improv.
Tour	**1**	191.5	**186.0**	2.9%	**259.7**	**229.5**	11.6%	**225.3**	**198.2**	12.0%	**252.3**	226.5	10.2%
	2	**190.9**	186.4	2.4%	264.1	237.8	10.0%	228.2	201.9	11.5%	257.2	**225.5**	12.3%
	3	194.1	191.8	1.2%	260.3	239.1	8.2%	228.8	206.9	9.6%	261.5	238.4	8.8%
	4	199.7	195.3	2.2%	273.4	244.9	10.4%	225.6	209.3	7.2%	267.1	243.9	8.7%
Robust Tour		190.9	186.0	2.6%	259.7	229.5	11.6%	225.3	198.2	12.0%	252.3	225.5	10.6%

Table 6 results indicate that *tours 1* and *2* are dominant tours for the four start times. In the remainder of section, we assume that *tour 1* is selected for both static and dynamic policies. In fact, *tour 1* is indeed the selected robust tour for start times 6:00 and 12:00 and its performance difference from the selected robust tour is small for starting times of 0:00 and 18:00.

Next, we set the time windows according to the procedure described in Section 3.2.1.1. Here, we assume the width of the time windows is 30 minutes for all supplier sites. Further, we allow unrestricted waiting for early arrivals at all sites. We illustrate the time windows through their centers (mean site arrival times) and deviations around centers (standard deviation of site arrival times) in Table 5 for the selected robust *tour 1*.

Table 5. Simulated mean arrival times (with time windows) to the sites in the sequence of *tour 1* based on *static* and *dynamic* policies

	Mean Site Arrival Times								Std. Dev. of Site Arrival Times							
ST	**0:00**		**6:00**		**12:00**		**18:00**		**0:00**		**6:00**		**12:00**		**18:00**	
Policy	Sta.	Dyn.	Sta.	Dyn.	Sta.	Dyn.	Sta.	Dyn.	Sta.	Dyn.	Sta.	Dyn.	Sta.	Dyn.	Sta.	Dyn.
Site 132	18.7	18.5	26.0	20.3	21.6	19.9	23.8	23.6	1.2	1.0	1.9	1.6	1.7	1.7	1.8	1.8
Site 103	67.3	66.6	87.9	80.7	79.2	74.1	102.7	93.5	3.3	2.8	5.2	4.4	4.7	3.9	6.2	4.8
Site 51	98.7	97.9	131.6	113.9	116.8	108.8	137.9	128.6	4.6	3.8	7.3	6.0	6.4	5.3	9.1	6.3
Site 61	147.0	143.9	197.2	172.5	169.8	154.3	192.2	180.2	6.3	5.5	10.3	8.3	8.7	7.0	11.8	8.0
Site 80	179.2	175.1	240.1	214.2	208.4	185.6	231.8	212.7	7.8	7.0	13.1	10.1	11.0	8.4	14.2	9.8

The mean and standard deviation of return times to DC (node #80) corresponds to the mean and standard deviation of the *tour 1* trip times. Note that the means and standard deviations of DC return times in Table 5 are different than those of tour trip times without time windows reported in Table 2. These differences are due to the waiting at the sites upon early arrival. The waiting due to early arrival increases (decreases) the mean (standard deviation) of the tour trip time.

Table 6 presents the service level performance (on-time deliver) of static and dynamic policies for *tour 1* at different start times. These results are based on simulating *tour 1* using dynamic and static policies between sites subject to the time windows set for each policy in Table 5. Table 6 results show that as congestion increases, the dynamic policy taking real-time traffic information into account becomes increasingly superior to the static policy planning methods. The on-time delivery performance can be increased up to 8% for a site and up to 4% for a tour (starting at 18:00). We conclude that the dynamic policy not only decreases transportation cost (measured by trip time), but also increases the delivery service level performance (measured by on-time delivery).

Table 6. On-time delivery performances (in percentages) of the policies with time windows

	On-time delivery performances (in percentages)							
ST	**0:00**		**6:00**		**12:00**		**18:00**	
Policy	Sta.	Dyn.	Sta.	Dyn.	Sta.	Dyn.	Sta.	Dyn.
Site 132	100	100	100	100	100	100	100	100
Site 103	100	100	100	100	100	100	100	100
Site 51	100	100	96	100	98	100	95	100
Site 61	98	100	91	98	96	98	91	97
Site 80	97	98	88	94	92	96	86	94

The results in Table 6 are obtained with the assumption that there is unrestricted waiting for early arrivals at all sites. Further, the time windows are centered on the mean site arrival times depending on whether static or dynamic routing policy is used between pairs of sites. As explained in Section 3.2.1.1, one could shift the time windows to the right of the center (expected site arrival time) to reduce the late arrival occurrences. However, the effectiveness of this modification relies on the maximum allowable waiting time imposed for early arrivals. To understand the effect of shifting time windows, we adapted time windows of the static policy as the time windows of the dynamic policy. This allows us to retain the assumption of unrestricted waiting for early arrivals and compare the on-time delivery results of dynamic policy with those in Table 6. The results of on-time delivery with dynamic policy using the time windows of the static policy are presented in Table 7. With this setting, the on-time delivery performance of the truck following the dynamic policy is 100 percent for all starting times and for all sites based on historic data (66 weekdays of May, June, and July 2009). Clearly, this improvement in on-time performance is attained with increased waiting at sites. Table 7 also presents the mean waiting times at sites.

Table 7. On-time delivery performances (in percentages) and average waiting times (in minutes) for dynamic policy when setting time windows of dynamic policy as the time windows of static policy

		On-time delivery performances (in percentages)				Waiting times (in minutes)			
ST		0:00	6:00	12:00	18:00	0:00	6:00	12:00	18:00
Site	**132**	100	100	100	100	0	0	0	0
	103	100	100	100	100	0	0.06	0	0
	51	100	100	100	100	0	3.49	0.01	0
	61	100	100	100	100	0	10.12	2.21	0
	80	100	100	100	100	0	11.91	9.55	0

9. CONCLUSIONS

In this work, we studied the STD-TSP with dynamic routing problem. It is an extension of stochastic TSP and aims to find a robust milk-run tour of a given set of sites (i.e., DC and suppliers) while dynamically routing on a stochastic time-dependent road network between sites' visits to meet the time windows requirements. The solution is comprised of static TSP tour of sites that remains fixed for extended periods (e.g., months) and a dynamic routing policy between pairs of sites. The static tour is motivated by the fact that tours cannot be changed on a regular basis (e.g., daily) for milk-run pickup and delivery in routine JIT production. The objective trades off the expected duration of the tour with its variability, capturing the tradeoff between transportation efficiency and on-time delivery service level.

We proposed a sequential solution approach. We first determined the travel time distributions between each pair of sites by formulating and solving a stochastic dynamic programming formulation for the dynamic routing problem on a stochastic time-dependent road network. The dynamic routing model exploits the real-time traffic information available from ITS. We proposed effective data driven methods for accurate modeling and estimation of recurrent congestion states and their state transitions. Whereas we assumed arcs are independent in generating dynamic routing policies, we simulated dynamic routing policies using historic data to capture the arc dependencies in all our experiments. Using simulation results, we estimated the site-to-site travel time distributions. Once the travel time distributions were estimated for every pair of sites at different departure times, we employed a stochastic time-dependent dynamic programming (STD-DP) to solve the problem and select the robust tour minimizing the mean-variance objective of the trip time. We also provided a time window setting procedure to increase on-time delivery performance and support workload leveling.

We tested the proposed methodology on a real case study application using the road network from Southeast Michigan. This study corresponded to an automotive JIT production system where an OEM's DC is replenished by milk-run pickup and deliveries from multiple suppliers. The study road network covered major freeways and highways in and around the Detroit metropolitan area. To quantify the benefits of using dynamic policy, we compared the selected robust STD-TSP tours with those of the static routing policy between pair of sites. We first experimented without time windows for both static and dynamic policies. The results showed that the dynamic policy saves 8.1% in trip duration on the average and reduces standard deviation of trip duration by 21.6% on the average. After setting the time windows according to the expected site arrival times, we showed that the on-time delivery performance can be increased up to 8% for a site and up to 4% for a tour by using dynamic routing policy. Lastly, we showed that it is possible to further increase the on-time performance by setting the time windows of dynamic routing policy according to those of the static policy. We concluded that the dynamic policy not only decreases transportation cost (measured by trip time), but also increases the delivery service level performance (measured by on-time delivery).

10. Recommendations for Further Research

There are several promising extensions of this research. The dynamic routing policies are generated by assuming arc independence. While we have partly compensated for this by simulating the policies using actual historical data from the ITS network, the policies themselves are not guaranteed to be optimal if there are significant arc interactions. Hence, a future study is to account for the link interactions in modeling congestion and generating dynamic routing policies. Another future study is to integrate the proposed approach within the more general problem of VRP, where the supplier-route assignment decisions are made in addition to the routing of individual vehicles.

11. Recommendations for Implementation

The research identified a number of recommendations for implementation to help leverage the full potential of dynamic routing of freight vehicles using real-time ITS information.

- Implementation mechanisms for ensuring data quality. Recommendations include collecting and sharing up-to-date sensor maintenance and placement information in the implementation network. This allows the users of the models and algorithms developed to revise their estimates of the travel times as well as traffic behavior under incident conditions. While the majority of the arteries do not have sensors, we found that some of the sensors in major highways are inactive. The absence of these sensors on the large segments of highways creates quality problems associated with distribution estimations for travel times as well as state transitions. In addition, the data collected from the some of the sensors are found to be inaccurate, e.g. inconsistent speed data, which may be attributable to weather, sensor's health state, and communication network inefficiencies. Recommendations for the missing sensor information or inaccurate data captures include benchmarking the traffic condition on the network segments devoid of sensors with those having sensors and reconciling and using a linear regression estimation of the speed data at any given time between adjacent sensors on a highway segment.

- The off-line routing policy generation is impractical given the large number of links and incident state possibilities. Recommendation for implementation is to communicate the route actions (which road network link to select next) to the driver through a wireless connection (e.g. satellite) in real time. The identification of the real-time routing decisions is achieved through a centralized dynamic routing decision support system implementing the models and algorithms developed in this research. The decision support system is recommended to extract the real-time traffic congestion information from the ITS server. When the server is down or there are communication problems, the default operating mode for the decision support system is to assume the long-term congestion state probabilities. Figure 16 illustrates the recommended framework for data communication and decision support integration.

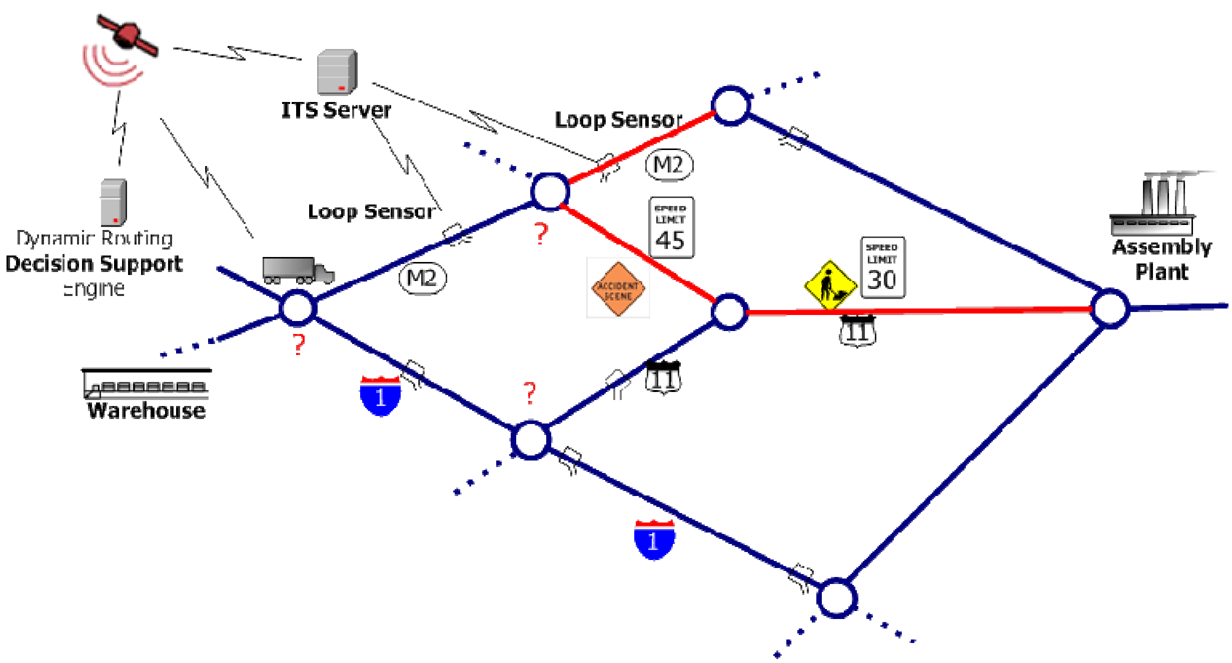

Figure 5. Recommended framework for data communication and decision support integration

Bibliography

Bertsekas, D.P., 2001. Dynamic programming and optimal control. Volume I, Athena Scientific, Belmont, MA.

Chang, T.-S., Wan, Y.-W., OOI, W.T., 2010. A stochastic dynamic traveling salesman problem with hard time windows. European Journal of Operational Research, 198(3): 748-759.

Chen, C., Skabardonis, A., Varaiya, P., 2003. Travel time reliability as a measure of service, 82nd Annual Meeting Transportation Research Board, Washington, D.C.

Chuah, K. H., J. C. Yingling, 2005. Routing for just-in-time supply pickup and delivery system. Transportation Science, 39(3): 328-339.

Cordeau, J.-F., Desaulniers, G., Desrosiers, J., Solomon, M.M., Soumis, F., 2000. The VRP with time windows. In: P. Toth and D. Vigo (Editors), The Vehicle Routing Problem. SIAM Monographs on Discrete Mathematics and Applications. SIAM, Philadelpia, PA.

Figliozzi, M.A., 2010. The impacts of congestion on commercial vehicle tour characteristics and costs. Transportation Research Part E 46: 496-506.

Golob, T.F., Regan, A.C., 2003. Traffic congestion and trucking managers' use of automated routing and scheduling. Transportation Research Part E, 39(1): 61-78.

Groenevelt, H., 1993. The just-in-time system. In: S.C. Graves, A.H.G. Rinnooy Kan and P.H. Zipkin (Editors), Handbooks in Operations Research and Management Science. North-Holland, Amsterdam, pp. 629-670.

Holguin-Veras, J., Patil, G., 2005. Observed trip chain behavior of commercial vehicles. Transportation Research Record, 1906: 74-80.

Houck, D., Picard, J., Queyranne, M., Vegamunti, R., 1980. The travelling salesman as a constrained shortest path problem: Theory and computational experiment. Opsearch, 17(2-3): 93-109.

Hunt, J., Stefan, K., 2005. Tour-based microsimulation of urban commercial movements, 16th International Symposium on Transportation and Traffic Theory, Maryland.

Ichoua, S., Gendreau, M., Potvin, J.-Y., 2003. Vehicle dispatching with time-dependent travel times. European Journal of Operational Research, 144: 379-396.

Johnson, D.S., McGeoch, L.A., 1997. The traveling salesman problem: A case study in local optimization. In: E. Aarts and J.K. Lenstra (Editors), Local Search in Combinatorial Optimization. Wiley, London, pp. 215-310.

Jula, H., Dessouky, M., Ioannou, P.A., 2006. Truck route planning in nonstationary stochastic networks with time windows at customer locations. IEEE Transactions on Intelligent Transportation Systems, 7(1): 51-62.

Junger, M., Reinelt, G., Rinaldi, G., 1995. The traveling salesman problem. In: M. Ball, T. Magnanti, C. Monma and G. Nemhauser (Editors), Network Models. North Holland, Amsterdam, pp. 225-330.

Kim, S., Lewis, M.E., White III, C.C., 2005a. Optimal vehicle routing with real-time traffic information. IEEE Transactions on Intelligent Transportation Systems, 6(2): 178-188.

Kim, S., Lewis, M.E., White III, C.C., 2005b. State space reduction for non-stationary stochastic shortest path problems with real-time traffic congestion information. IEEE Transactions on Intelligent Transportation Systems, 6(3): 273-284.

Lambert, V., Laporte, G., Louveaux, F., 1993. Designing collection routes through bank branches. Computers & Operations Research, 20(7): 783-791.

Laporte, G., Louveaux, F., Mercure, H., 1992. The vehicle routing problem with stochastic travel times. Transportation Science, 26(3): 161-170.
Laporte, G., 2009. Fifty years of vehicle routing. Transportation Science, 43(4): 408-416.
Laporte, G., 2010. A concise guide to the Traveling Salesman Problem. Journal of the Operational Research Society, 61(1): 35-40.
Malandraki, C., Dial, R.B., 1996. A restricted dynamic programming heuristic algorithm for the time-dependent traveling salesman problem. European Journal of Operational Research, 90: 45-55.
Ohlmann, J.W., Fry, M.J., Thomas, B.W., 2010. Route design for lean production systems. Transportation Science, In Press.
Psaraftis, H.N., Tsitsiklis, J.N., 1993. Dynamic shortest paths in acyclic networks with Markovian arc costs. Operations Research, 41: 91-101.
Verbeek, J.J., Vlassis, N., Kröse, B., 2003. Efficient greedy learning of Gaussian Mixture Models. Neural Computation, 5(2): 469-485.
Vleugel , J., Janic, M., 2004. Route choice and the impact of 'logistic routes'. In: E. Taniguchi and R. Thompson (Editors), Logistics systems for sustainable cities. Elsevier.