A Novel Approach to Routing and Dispatching Trucks Based on Partial Information in a Dynamic Environment

Final Report
Metrans Project 03-01

August 2004

Investigators:

Maged Dessouky, Ph.D.

Petros Ioannou, Ph.D.

Hossein Jula, Ph.D.

University of Southern California
Daniel J. Epstein Dept. of Industrial and Systems Engineering
Los Angeles, CA 90089-0193

And

University of Southern California
Electrical Engineering - Systems, EEB 200B
Los Angeles, CA 90089-2562

Disclaimer

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated under the sponsorship of the Department of Transportation, University Transportation Centers Program, and California Department of Transportation in the interest of information exchange. The U.S. Government and California Department of Transportation assume no liability for the contents or use thereof. The contents do not necessarily reflect the official views or policies of the State of California or the Department of Transportation. This report does not constitute a standard, specification, or regulation.

Abstract

Congestion affects the trucking industry on three major service dimensions: travel time, reliability, and cost. Trucking is a commercial activity, and trucking operations are driven by the need to satisfy customer demands and the need to operate at the lowest possible cost. This industry is highly competitive, with easy entry into almost any market, relatively little differentiation between operators and slim profit margins.

However, most of the developed techniques and models for planning, routing and scheduling in the trucking industry assume *'known'* static data as their input. For instance, in the Vehicle Routing Problem (VRP) the customer demands, travel costs, and travel times are known in advance. In this case, the fundamental problem is to determine the optimal *route* that minimizes a certain objective such as fleet size and travel distance. The built-in assumption of these approaches is that there will be small deviations on the realization of the demand and travel times from the plan so that the pre-determined routes form a basis for either the pickup or delivery schedule. In the real world, however, operations in any traffic network contain a fairly high degree of uncertainties including variable waiting and travel times due to traffic congestion, arrival of new orders, and cancellation of existing orders. In a highly dynamic and stochastic environment, the pre-planned optimal routes are no longer of practical use. In this case, most of the research effort has focused on easy to control dispatching rules. The drawback with these techniques is that they do not make use of pre-planned and known information.

There is a gap in the routing literature for systems that operate between the two ends of the spectrum, which is the most realistic condition for trucking operations. Our research on partial route development addresses this gap by developing a new approach within an area that has received little attention. In a simulation study, we demonstrate the benefits of the partial routing approach over the pre-planned and dispatching methodologies.

Table of Contents

List of Figures

List of Tables:

Disclosure

Project was funded in entirety under this contract to California Department of Transportation.

Acknowledgments

We would like to thank Metrans for funding this research.

1 Introduction

The ability of a trucking company to succeed economically rests on its ability to move goods reliably and efficiently, with minimal delay. In many traffic networks especially in major cities, traffic congestion has already reduced mobility and system reliability, and has increased transportation costs. In addition to contributing to truck-drivers' inefficiency, traffic congestion is a major source of air pollution (especially diesel toxins), wasted energy, increased maintenance cost caused by the volume of trucks on roadway, etc. (Barton, 2001).

With the expected substantial increase in the volume of international and national cargos entering and moving through the U.S. highway system, together with the anticipated growth in the number of personal vehicles in use, it is expected that the condition of traffic congestion will only get worse, unless careful planning is initiated. For instance, according to the Federal Highway Administration, currently nearly half of the California's urban highways are congested[1]. It is expected that, between 2000 and 2025 in California, personal vehicle trips will be increased by 38 percent (CalTrans 2004), and the volume of containers moving in and out of the major California ports will be tripled (Mallon and Magaddino, 2001).

There are numerous ways to improve traffic congestion, and therefore reduce transport times associated with goods movements. Options include developing new and expanding current facilities, deploying advanced technologies, and improving operational characteristics and system management practices. It should be noted that the scarcity of land in major cities has made the option of developing new facilities if not infeasible significantly costly.

Goods movements, in nature, are time-dependent and contain uncertainties. For instance, the customer demands, travel costs, and travel times are uncertain, time-dependent variables. In the transportation industry, it is widely expected that the deployment of advanced technologies such as the use of information technologies can reduce the level of uncertainties to a manageable level and makes the use of dynamic formulations and solutions feasible. These technologies include:

---

[1] Federal Highway Administration defines congestion as when an Interstate highway exceeds 13,000 vehicles per-lane-mile daily, or 5,000 vehicles per-lane-mile on principal arteries.

- Vehicle tracking, such as global-positioning-systems (GPS), which allow vehicle locations to be determined within a few meter level accuracy.

- Wireless communication, via satellite, cellular and paging networks, which enable 2-way communication with mobile fleets.

- Navigable map databases, from which point-to-point distances can be calculated.

- Real-time information services, which allow for dynamic calculation of travel speeds.

Whereas in the past, it was difficult for a company to control or route vehicles once they left the terminal, these technologies make accurate dynamic real-time routing a possible reality.

The focus of this report is to investigate dynamic methods to improve the operational characteristics of goods movements by developing techniques that can be easily implemented using new but currently available computer and information technologies. The use of information technologies in goods movements can narrow the gap between highly uncertain systems in reality and the assumed known static systems in theory. It is known that for static systems, where the network parameters are known a priori, the well-established routing and scheduling algorithms lead to optimum solutions. On the other hand, in a highly uncertain and dynamic system, heuristic dispatching techniques work reasonably well. As recently observed by Powell et al. (2000), in the presence of high degree of uncertainties, it is widely expected that optimal solutions for goods movements will be outperformed, over time, by algorithms that are more local in nature.

Thus, on one end of the spectrum are route planning techniques when it is reasonable to assume the system is deterministic, and on the other end are dispatching heuristics when the system is highly dynamic and uncertain. There exists a gap in the literature for situations that are in between the two ends of the spectrum. The purpose of this study is to address this gap. The objective of this study is to identify the level of route planning in the freight transportation techniques. In these situations a technique that builds partial routes and has capabilities for dynamic adjustments in real-time may be the most suitable approach.

In this report, we introduce the concept of partial routing to address the gap between routing in deterministic static networks and dispatching in dynamic stochastic networks. The length of the partial route indicates the suitable level of route planning that is needed to account for the uncertainties. We investigate the benefits as well as methods for generating partial routing techniques for goods movements in uncertain and dynamic environments. Furthermore, we develop a dynamic routing methodology, which has the capability for dynamic route adjustments by incorporating updated information of traffic conditions.

The report is organized as follows. Section 2 is devoted to the literature review. In Section 3,the partial routing methodology for dynamic stochastic networks is developed. In this section, we first investigate methods to estimate traveling times on arcs and arriving times at nodes of the network. Then, these estimations are used to generate partial routes, which guarantee some level of service. Simulation experiments are discussed in Section 4. To evaluate the developed partial routing method and to compare this method with other methods three scenarios are developed and considered: deterministic network, stationary stochastic network, and dynamic stochastic network. Section 5 concludes this report and provides recommendations for future studies. Section 6 considers the wider implementations of this study.

2 Literature Review

Most developed techniques and models for planning, routing and scheduling assume *'known'* static data as their input. For instance, in the Vehicle Routing Problem (VRP) the customer demands, travel costs, and travel times are known in advance. In this case, the fundamental problem is to determine the optimal *route* that minimizes a certain objective such as fleet size and travel distance (Laporte, 1992). The built-in assumption of these approaches is that there will be small deviations on the realization of the demand and travel times from the plan so that the pre-determined routes form a basis for either the pickup or delivery schedule.

In the real world, however, operations in any traffic network contain a fairly high degree of uncertainties including variable waiting and travel times due to traffic congestion, arrival of new orders, and cancellation of existing orders. (e.g., see Powell et al., 1995; Powell, 1996). That is why human operators (dispatchers) still play a major role in route planning and vehicle scheduling in the trucking industry. Dispatchers assign drivers to cargoes and inform the drivers about traffic conditions and changes in customer requests in addition to assisting them in departure/arrival decisions and providing navigational information (Ng et al., 1995). Dispatchers currently obtain information about traffic conditions, mostly through radio traffic reports and through information relayed back by the drivers (Hall and Intihar, 1997).

Since route planning is a dynamic problem, any efficient algorithm should also be *dynamic***.** Therefore, the dynamic route planning has emerged as an active and intense area of research, both due to industry needs, but also due to technological advances, including map databases, location determination technology (e.g., GPS), wireless communication and mobile computing.

Probably the stochastic shortest path problem (SSPP) is the most researched problem among non-deterministic route-planning problems. Pattanamekar et al. (2003) examined the characteristics of the mean and variance of traveling time under uncertainties. They developed mathematical models for predicting the first and second order approximation of the mean and variance of the individual travel time in the future. The approximations were found using Taylor's series expansions around the estimated arrival time.

Fu (2001) studied the shortest path problem in traffic networks in which travel times on links are random variables with known mean and standard deviation. In his paper, the realization of traveling time was assumed to be estimated in advance and made available to vehicle's routing system. Fu proposed a close-loop adaptive routing rule (CAR) with the objective of identifying the immediate node instead of the whole path. The problem was formulated as a dynamic programming problem, and an approximate labeling algorithm was developed to solve the problem.

Hall (1996) showed that for a time-dependent stochastic network, standard path algorithms (e.g., Dijkstra's algorithm) do not find the minimum expected cost path. He showed that the optimal choice is not a simple path but an adaptive decision rule (policy), and proposed using Dynamic Programming to find such a policy.

Kim et al. (2003) studied optimal vehicle routing in a non-stationary stochastic network. They developed decision making procedures based on a Markov decision process model for determining the optimal driver attendance time, optimal departure times, and optimal routing policies. The methodology was used to develop routing strategies in the stochastic shortest path problem. The authors concluded that the real-time traffic information combined with historical data can significantly reduce expected total costs and vehicle usage. They also noted that the implementation of historical data is easier than incorporating real-time information.

Bander and White (2002) also considered the non-stationary shortest path problem in which travel time along each arc is modeled as a random variable. The distribution of the random variable is dependent on the time that travel is begun. The paper presented an algorithm based on AO*. It demonstrated that AO* is significantly more computationally efficient than dynamic programming when lower bounds on the cost function is available.

Miller-Hooks and Mahmassani (2003) explored three measures for comparing time-varying, random path travel times: deterministic dominance, first-order stochastic dominance, and comparison via expected value. The measures were used for comparing paths at a single departure time and then it was generalized over multiple time intervals.

In contrast to the dynamic and stochastic shortest path problem (DSSPP), the research efforts on dynamic and stochastic vehicle routing problem (DSVRP) has been scant. Ichoua et al. (2003) investigated the time-dependent vehicle routing problem (TDVRP) with soft time-windows.

They presented a time-dependent speed model to calculate the traveling times between each two nodes. In this model the horizon time was divided into time periods. In each time period the speeds of vehicles are assumed to be constant. A heuristic method known as Tabu Search was developed to find good routes for the TDVRP problem.

Bertsimas and Van Ryzin (1991) examined a model of the VRP problem, referred to as the Dynamic Traveling Repairman Problem (DTRP). In DTRP demands for service arrive in time; these demands are independent and uniformly distributed in a Euclidean service region. The problem is to find a policy for routing the service vehicle that minimizes the average time demands spent in the system (wait plus service). Hence, the problem is stochastic and dynamic in the sense that demands appear randomly in time and space.

Bertsimas and Van Ryzin (1993) extended their work of 1991 by considering m identical vehicles to serve customers in the DTRP. They investigated two cases: 1) vehicles with unlimited capacity, 2) vehicles that can serve at most q customers.

3 Partial Routing in Dynamic Stochastic Networks

In this section, we will develop the partial routing methodology in dynamic and stochastic transportation networks.

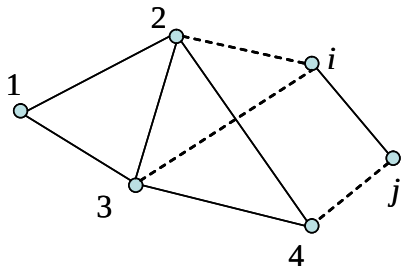

Figure 1: A typical transportation network.

Shown in Figure 1 is a typical transportation network. Let $G:(N,A)$ be a transportation network (which is called a graph hereafter) with node set $N=\{1,..,i,j,..,n\}$ and arc set $A=\{(i,j)|i,j\in N\}$. Solid lines in Figure 1 represent the direct connections (arcs) between two adjacent nodes and dashed lines are indirect connections, which consist of two or more direct connections.

Given graph *G*, our proposed partial routing methodology consists of the following stages.

1) *Predicting traveling times on arcs*: Given the time of the day as well as the historical and real-time data of the traveling time on arcs of graph *G*, we predict the future traveling times on the arcs, recursively.

2) *Estimating arrival times at nodes*: Given the departure time from one node together with the predicted traveling times on arcs of graph *G* (from stage 1), the arrival times at other nodes of the network are estimated.

3) *Generating partial routes*: Given the estimated arrival time at nodes of graph *G* (from stage 2), partial routes are generated which guarantee some service levels.

4) *Selecting the best route*: Among all partial routes (from stage 3), we select the one that covers the most nodes at least cost. Nodes on the generated partial route will be visited next.

In the following, each stage will be investigated and discussed in detail.

3.1 Predicting the Traveling Times on Arcs

Here we assume that the historical data of traveling time on arcs of the network is available. These data can be obtained through observing and averaging the traveling time on each arc for a long period of time. In addition, we assume that the real-time information regarding the traveling time on arcs is also available.

Let

T be the length of the planning horizon,

k be an index of time in this horizon T,

$x_{ij}(k)$ be the traveling time between nodes i and j at time k,

$x_{ij}^{h}(k)$ be the historical traveling time between nodes i and j at time k obtained from previous measurements on arc (i,j),

$u_{ij}(k)$ be the historical change in the traveling time between nodes i and j from time k to k+1, in other words $u_{ij}(k) = x_{ij}^{h}(k+1) - x_{ij}^{h}(k)$, and

$y_{ij}(k)$ be the measured traveling time between nodes i and j at time k.

We model the dynamic behavior of the traveling time on arc (i,j) as follows:

$$\begin{aligned} x_{ij}(k+1) &= x_{ij}(k) + u_{ij}(k) + w_{ij}(k) \\ y_{ij}(k) &= x_{ij}(k) + v_{ij}(k) \end{aligned} \qquad (1)$$

where $w_{ij}(k)$ is the traveling time disturbance on arc (i,j) representing real-time changes in the traveling time at time k, which are not included in the historical data $u_{ij}(k)$, and $v_{ij}(k)$ is the

error in the measurements of traveling time on arc (*i*,*j*) which is introduced by sensing devices or human errors.

We assume that $u_{ij}(k)$, $w_{ij}(k)$, $v_{ij}(k)$, and $x_{ij}(0)$ are all mutually uncorrelated Gaussian random variables with the following specifications:

$$E\{u_{ij}(k)\} = \eta_{ij}(k); \quad E\{u_{ij}(k)u_{ij}(l)\} = \begin{cases} \sigma_{ij}^2(k) + \eta_{ij}^2(k) & k = l \\ 0 & k \neq l \end{cases} \tag{2}$$

$$E\{w_{ij}(k)\} = 0; \qquad E\{w_{ij}(k)w_{ij}(l)\} = \begin{cases} q_{ij}^2(k) & k = l \\ 0 & k \neq l \end{cases} \tag{3}$$

$$E\{v_{ij}(k)\} = 0; \qquad E\{v_{ij}(k)v_{ij}(l)\} = \begin{cases} r_{ij}^2(k) & k = l \\ 0 & k \neq l \end{cases} \tag{4}$$

and

$$x_{ij}(0) \sim \left(\lambda_{ij}(0), p_{ij}(0)\right) \tag{5}$$

In the following, we use a Kalman-Filtering corrector-predictor technique to estimate the future traveling times on the arcs of graph *G*.

3.1.1 Single-Stage Predictor

In the single stage predictor case, given the measured traveling time $y_{ij}(k)$ on arc (*i*,*j*) at time *k*, we would like to estimate the traveling time $x_{ij}(k+1)$ on the arc at time *k*+1. From estimation theory results, we know that the estimator that minimizes the mean-squared error of the estimation error is given by

$$\hat{x}_{ij}(k+1|k) = E\{x_{ij}(k+1)|y_{ij}(k)\} \tag{6}$$

where $\hat{x}_{ij}\left(k+1|k\right)$ denotes the estimate of traveling time $x_{ij}\left(k+1\right)$ given measurement $y_{ij}\left(k\right)$, and is called the mean-squared *predicted estimator* of $x_{ij}\left(k+1\right)$. Using our dynamical model in (1), the predicted estimator $\hat{x}_{ij}\left(k+1|k\right)$ can be generated as

$$\begin{aligned}\hat{x}_{ij}\left(k+1|k\right) &= E\left\{x_{ij}\left(k\right)+u_{ij}\left(k\right)+w_{ij}\left(k\right)\middle|y_{ij}\left(k\right)\right\} \\ &= \hat{x}_{ij}\left(k|k\right)+\eta_{ij}\left(k\right)\end{aligned} \tag{7}$$

where $\hat{x}_{ij}\left(k|k\right)$ is the mean-squared *filtered estimator* of $x_{ij}\left(k\right)$ given $y_{ij}\left(k\right)$. Equation (7) indicates that to calculate the predicted estimator $\hat{x}_{ij}\left(k+1|k\right)$ the value of the filtered estimator $\hat{x}_{ij}\left(k|k\right)$ should be first obtained.

Let's denote by $\tilde{x}_{ij}\left(k+1|k\right)$ the single-stage *predictor error*, which is defined as

$$\tilde{x}_{ij}\left(k+1|k\right) = x_{ij}\left(k+1\right)-\hat{x}_{ij}\left(k+1|k\right) \tag{8}$$

Note that since $x_{ij}\left(k+1\right)$ and $y_{ij}\left(k\right)$ are joint Gaussian random variables, the estimator in (6) is unbiased [Mendel 1995], and we have,

$$E\left\{\tilde{x}_{ij}\left(k+1|k\right)\right\} = 0 \tag{9}$$

Using (9), the error variance of the estimator is given by

$$\begin{aligned}\mathrm{var}\left(\tilde{x}_{ij}\left(k+1|k\right)\right) &= E\left\{\tilde{x}_{ij}\left(k+1|k\right)^2\right\} \\ &= E\left\{\left(x_{ij}\left(k+1\right)-\hat{x}_{ij}\left(k+1|k\right)\right)^2\right\} \\ &= E\left\{\left(x_{ij}\left(k\right)+u_{ij}\left(k\right)+w_{ij}\left(k\right)-\hat{x}_{ij}\left(k|k\right)-\eta_{ij}\left(k\right)\right)^2\right\} \\ &= E\left\{\left(\tilde{x}_{ij}\left(k|k\right)+u_{ij}\left(k\right)-\eta_{ij}\left(k\right)+w_{ij}\left(k\right)\right)^2\right\}\end{aligned} \tag{10}$$

where $\tilde{x}_{ij}(k|k) = x_{ij}(k) - \hat{x}_{ij}(k|k)$ is the filter error which is the error of estimating $x_{ij}(k)$ given $y_{ij}(k)$. Since $\tilde{x}_{ij}(k|k)$, $\left(u_{ij}(k) - \eta_{ij}(k)\right)$, and $w_{ij}(k)$ are statistically uncorrelated, (10) can be simplified as,

$$\begin{aligned} \operatorname{var}\left(\tilde{x}_{ij}(k+1|k)\right) &= E\left\{\tilde{x}_{ij}(k|k)^2\right\} + \sigma_{ij}^2(k) + q_{ij}^2(k) \\ &= \operatorname{var}\left(\tilde{x}_{ij}(k|k)\right) + \sigma_{ij}^2(k) + q_{ij}^2(k) \end{aligned} \tag{11}$$

Equations (9) and (11) provide the mean and variance of the single-stage *predictor error* of the mean-squared *predicted estimator* of $x_{ij}(k+1)$ given in (7).

3.1.2 *m*th-Stage Predictor

In this subsection, we extend the single-stage predictor to the *m*th-stage predictor, such that given the measured traveling time $y_{ij}(k)$ at time *k* on arc (*i*,*j*), we determine an unbiased estimate of traveling time on the arc at time *k*+*m*, where *m*≥1. Similar to the single-stage predictor, we know that the estimator that minimizes the mean-squared estimation error is given by

$$\hat{x}_{ij}(k+m|k) = E\left\{x_{ij}(k+m)\middle|y_{ij}(k)\right\} \tag{12}$$

where $\hat{x}_{ij}(k+m|k)$ is the estimate of $x_{ij}(k+m)$ given the measurement $y_{ij}(k)$. Using (1) recursively, we have

$$\begin{aligned} \hat{x}_{ij}(k+m|k) &= E\left\{x_{ij}(k) + \sum_{l=0}^{m-1} u_{ij}(k+l) + \sum_{l=0}^{m-1} w_{ij}(k+l)\middle|y_{ij}(k)\right\} \\ &= \hat{x}_{ij}(k|k) + \sum_{l=0}^{m-1} \eta_{ij}(k+l) \end{aligned} \tag{13}$$

The *m*th-stage predictor error is defined as $\tilde{x}_{ij}(k+m|k) = x_{ij}(k+m) - \hat{x}_{ij}(k+m|k)$. Similar to the single stage predictor case, since $x_{ij}(k+m)$ and $y_{ij}(k)$ are joint Gaussian random variables, the estimator in (13) is unbiased, i.e.,

$$E\left\{\tilde{x}_{ij}\left(k+m\middle|k\right)\right\}=0 \tag{14}$$

The error variance of the *m*th-stage predictor is

$$\begin{aligned}\mathrm{var}\left(\tilde{x}_{ij}\left(k+m\middle|k\right)\right)&=E\left\{\tilde{x}_{ij}\left(k+m\middle|k\right)^2\right\}=E\left\{\left(x_{ij}\left(k+m\right)-\hat{x}_{ij}\left(k+m\middle|k\right)\right)^2\right\}\\&=E\left\{\left(x_{ij}\left(k\right)+\sum_{l=0}^{m-1}u_{ij}\left(k+l\right)+\sum_{l=0}^{m-1}w_{ij}\left(k+l\right)-\hat{x}_{ij}\left(k\middle|k\right)-\sum_{l=0}^{m-1}\eta_{ij}\left(k+l\right)\right)^2\right\}\\&=E\left\{\left(\tilde{x}_{ij}\left(k\middle|k\right)+\sum_{l=0}^{m-1}\left(u_{ij}\left(k+l\right)-\eta_{ij}\left(k+l\right)\right)+\sum_{l=0}^{m-1}w_{ij}\left(k+l\right)\right)^2\right\}\end{aligned} \tag{15}$$

Using (2) and (3), we know that $\tilde{x}_{ij}\left(k\middle|k\right)$, $\left(u_{ij}\left(k+l\right)-\eta_{ij}\left(k+l\right)\right)$, and $w_{ij}\left(k+l\right)$ are statistically uncorrelated for $\forall l=0,...,m-1$, therefore

$$\begin{aligned}\mathrm{var}\left(\tilde{x}_{ij}\left(k+j\middle|k\right)\right)&=E\left\{\tilde{x}_{ij}\left(k\middle|k\right)^2\right\}+\sum_{l=0}^{m-1}\sigma_{ij}^2\left(k+l\right)+\sum_{l=0}^{m-1}q_{ij}^2\left(k+l\right)\\&=\mathrm{var}\left(\tilde{x}_{ij}\left(k\middle|k\right)\right)+\sum_{l=0}^{m-1}\sigma_{ij}^2\left(k+l\right)+\sum_{l=0}^{m-1}q_{ij}^2\left(k+l\right)\end{aligned} \tag{16}$$

Equations (14) and (16) present the mean and variance of the *m*th-stage *predictor error* of the mean-squared *predicted estimator* of $x_{ij}\left(k+m\right)$ given in (13). Equation (16) indicates that as *m*, the number of stages of prediction, increases, the variance of the prediction error increases too. In other words, as *m* becomes larger, we face more uncertainties in estimating the traveling time on arc (*i*,*j*) using the current information at time *k*.

3.1.3 State Filter

As seen from (7) and (13), the predicted estimate of traveling time $x_{ij}\left(k+m\right)$ on arc (*i*,*j*) at time *k*+*m*, $m\geq 1$, depends on the value of the filtered estimator $\hat{x}_{ij}\left(k\middle|k\right)$, i.e. the estimate of traveling time $x_{ij}\left(k\right)$ at time *k*, given the measured traveling time $y_{ij}\left(k\right)$ at time *k*. The Kalman filter allows us to calculate the filtered estimator $\hat{x}_{ij}\left(k\middle|k\right)$. The predictor-corrector form of the Kalman filter is as follows [Mendel 1995]:

$$\hat{x}_{ij}\left(k\middle|k\right)=\hat{x}_{ij}\left(k\middle|k-1\right)+\mathrm{K}_{ij}\left(k\right)\tilde{y}_{ij}\left(k\middle|k-1\right) \tag{17}$$

where for the dynamic model in (1) $\tilde{y}_{ij}\left(k\middle|k-1\right)$ is the *measurement residual process* of $y_{ij}\left(k\right)$, and is defined as

$$\tilde{y}_{ij}\left(k\middle|k-1\right)=y_{ij}\left(k\right)-\hat{x}_{ij}\left(k\middle|k-1\right), \tag{18}$$

and $\mathrm{K}_{ij}\left(k\right)$ is the *Kalman gain* of arc (*i*,*j*) at time *k* which is specified by the following equations:

$$\mathrm{K}_{ij}\left(k\right)=\frac{\mathrm{cov}\left(x_{ij}\left(k\right),\tilde{y}_{ij}\left(k\middle|k-1\right)\right)}{\mathrm{var}\left(\tilde{y}_{ij}\left(k\middle|k-1\right)\right)}=\frac{\mathrm{var}\left(\tilde{x}_{ij}\left(k\middle|k-1\right)\right)}{\mathrm{var}\left(\tilde{x}_{ij}\left(k\middle|k-1\right)\right)+r_{ij}^{2}(k)}, \tag{19}$$

and from (11)

$$\mathrm{var}\left(\tilde{x}_{ij}\left(k\middle|k-1\right)\right)=\mathrm{var}\left(\tilde{x}_{ij}\left(k-1\middle|k-1\right)\right)+\sigma_{ij}^{2}\left(k-1\right)+q_{ij}^{2}\left(k-1\right), \tag{20}$$

and

$$\mathrm{var}\left(\tilde{x}_{ij}\left(k\middle|k\right)\right)=\left(1-\mathrm{K}_{ij}\left(k\right)\right)\mathrm{var}\left(\tilde{x}_{ij}\left(k\middle|k-1\right)\right). \tag{21}$$

It is always possible that the value of $\mathrm{var}\left(\tilde{x}_{ij}\left(k\middle|k-1\right)\right)$ is equal to zero for some *k*, which makes equation (19) undefined if $r_{ij}\left(k\right)=0$. Therefore, we require that $r_{ij}\left(k\right)>0$ for all *k*.

Note that the error variance of traveling time on arc (*i*,*j*) at time *k*=0 is given by (5), i.e., $\mathrm{var}\left(\tilde{x}_{ij}\left(0\middle|0\right)\right)=p_{ij}\left(0\right)$, which initializes equations (19) to (21). In other words, given $\mathrm{var}\left(\tilde{x}_{ij}\left(0\middle|0\right)\right)$, $\sigma_{ij}^{2}\left(k\right)$, $q_{ij}^{2}\left(k\right)$, and $r_{ij}^{2}\left(k\right)$, $\forall k=0,1,2,\cdots$ by (5), (2), (3), and (4), respectively, all variables in (19) to (21) can be determined recursively in the following order:

$$\mathrm{var}\left(\tilde{x}_{ij}\left(0\middle|0\right)\right)\xrightarrow{\text{using (20)}}\mathrm{var}\left(\tilde{x}_{ij}\left(1\middle|0\right)\right)\xrightarrow{\text{using (19)}}\mathrm{K}_{ij}\left(1\right)\xrightarrow{\text{using (21)}}\mathrm{var}\left(\tilde{x}_{ij}\left(1\middle|1\right)\right)\rightarrow\cdots. \tag{22}$$

In particular, the Kalman gain $\mathrm{K}_{ij}\left(k\right)$ of arc (*i*,*j*) at each time *k*, is determined by (22).

Note also that Equation (17) is the predictor-corrector form of the Kalman filter, in which the predictor step $\hat{x}_{ij}\left(k\middle|k-1\right)$ uses information from (7) to predict the next step, and the corrector step $\mathrm{K}_{ij}\left(k\right)\tilde{y}_{ij}\left(k\middle|k-1\right)$ uses the new measurement to update or correct the predictor estimation $\hat{x}_{ij}\left(k\middle|k-1\right)$. It can be shown that the filter estimator given by (17) is unbiased [Mendel 95].

The filtered estimate of traveling time on arc (*i*,*j*) at time *k*=0 is also given by (5), i.e., $\hat{x}_{ij}\left(0\middle|0\right)=\lambda_{ij}\left(0\right)$, which initializes equations (7) and (17). In other words, given $\hat{x}_{ij}\left(0\middle|0\right)$, $\eta_{ij}\left(k\right)$, and $\mathrm{K}_{ij}\left(k+1\right)$, $\forall k=0,1,2,\cdots$ by (5), (2), and (22), respectively, together with the measured traveling time $y_{ij}\left(k\right)$ at each arc (*i*,*j*) at each time *k*, the mean-squared *predicted estimate* and mean-squared *filtered estimate* of $x_{ij}\left(k\right)$ can be calculated recursively as follows:

$$\hat{x}_{ij}\left(0\middle|0\right)\xrightarrow{\text{using (7)}}\hat{x}_{ij}\left(1\middle|0\right)\xrightarrow{\text{using (17)}}\hat{x}_{ij}\left(1\middle|1\right)\rightarrow\cdots. \tag{23}$$

Therefore, at each time *k* and by using the *m*th-stage predictor in (13), the traveling time $\hat{x}_{ij}\left(k+m\middle|k\right)$ on each arc (*i*,*j*) at time *k*+*m*, can be predicted.

3.2 Estimating Arrival Times at Nodes

In the previous section, we developed a methodology to predict the traveling times on arcs of a given transportation network. Available historical (offline) data are used for predicting the traveling times, and new measurements (online) were used to correct and update our prediction at each instant of time. In this section, we use the predicted traveling times on arcs to estimate the arrival times at each node of the network.

Let's consider graph *G* as shown in Figure 1, again. We define route *r* in graph *G* as a set of nodes visited in the specified order. Figure 2 shows a typical route $r=\left\{1,2,..,i,j,..,d\right\}$. Let also A^r be the arc set associated with route *r* which is defined as

$$A^r=\left\{\left(i,j\right)\middle|i,j\in r,\text{and } j \text{ is visited immidiately after } i\right\} \tag{24}$$

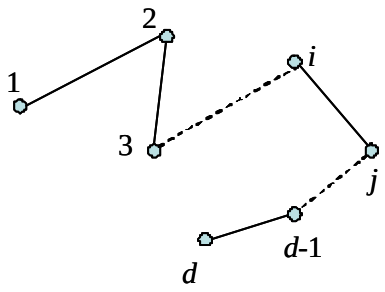

Figure 2: A typical route *r* in graph *G*.

We assume that the departure time from node {1} on route *r*, z_1^r, is given. We also assume that predicted traveling times on arcs of graph *G* are available using the developed methodology in Section 3.1. Given z_1^r, the departure time from node {1}, the arrival time at all other nodes on route *r* in Figure 2 can be determined using the following set of equations,

$$\begin{aligned} z_2^r &= z_1^r + x_{12}\left(z_1^r\right) \\ z_3^r &= z_2^r + x_{23}\left(z_2^r\right) \\ &\vdots \\ z_j^r &= z_i^r + x_{ij}\left(z_i^r\right) \\ &\vdots \\ z_d^r &= z_{d-1}^r + x_{d-1,d}\left(z_{d-1}^r\right) \end{aligned} \tag{25}$$

where z_i^r is the arrival time at node *i* taking route *r*, and $x_{ij}\left(z_i^r\right)$ is the traveling time on arc $(i,j)\in A^r$ at time z_i^r. It should be noted that, in general z_i^r takes a real value, while the methodology developed in Section 3.1 was based on integer numbers of discrete-time indices, *k*'s.

Generally speaking, the dynamics of many transportation networks is slow. Therefore, by selecting an appropriate sampling time, the error of approximating x_{ij} at time z_i^r with its value at the nearest sampling time would be negligible. More precisely, let T_S be the sampling period, and $k\cdot T_S \le z_i^r < (k+1)\cdot T_S$, where *k* is an integer number representing the time index. The argument is that with an appropriate selection of sampling period T_S, x_{ij} at time z_i^r can be well

approximated with $x_{ij}(k \cdot T_s)$, which for simplicity in this report has been denoted by $x_{ij}(k)$. Besides, from a practical point of view, the traveling time on each arc (*i*,*j*) cannot be measured continuously in time, i.e., in practice, u_{ij} and y_{ij} in (1) are discrete-time variables evaluated at each $k \cdot T_S$, $\forall k = 0,1,2,\cdots$.

Given the sampling period T_S and the arrival time at node *i*, z_i^r, we define the *discretizing* function $\Theta: \square \to \square$, which returns the discrete-time index, as follows:

$$\Theta(z_i) = \left\lfloor \frac{z_i}{T_S} \right\rfloor \tag{26}$$

where the *floor function* $\lfloor \cdot \rfloor$ returns the largest integer equal to or less than its argument. In the sequel, we use traveling times x_{ij} approximated at each time instant $k \cdot T_S$.

3.2.1 Estimating the arrival time at the second node

In this subsection, and for the purpose of route planning, we will estimate the arrival time at each node of route *r* in (25). Given the departure time from node {1}, z_1^r, the mean-squared estimate of the arrival time at node {2}, the next node on route *r*, is

$$\begin{aligned} \hat{z}_2^r &= E\left\{ z_2^r \middle| z_1^r \right\} = E\left\{ z_1^r + x_{12} \middle| z_1^r \right\} = z_1^r + E\left\{ x_{12} \middle| z_1^r \right\} \\ &\cong \hat{z}_1^r + \hat{x}_{12}\left(\Theta\left(z_1^r\right) \middle| \Theta\left(z_1^r\right) \right) \end{aligned} \tag{27}$$

where Θ is the discretizing function defined in (26) and $\hat{x}_{12}\left(\Theta\left(z_1^r\right) \middle| \Theta\left(z_1^r\right) \right)$ is the mean-squared *filtered estimate* of traveling time on arc (1,2), x_{12}, at time $\Theta\left(z_1^r\right)$ given the measured traveling time $y_{12}\left(\Theta\left(z_1^r\right)\right)$ at time $\Theta\left(z_1^r\right)$.

We denote by $\tilde{z}_2^r$ the error of the estimator in (27), which is defined as $\tilde{z}_2^r = z_2^r - \hat{z}_2^r$. Recall from Section 3.1 that $\hat{x}_{12}(k|k)$, $\forall k = 0,1,2,\cdots$ has a normal distribution. Besides, $\hat{x}_{12}(k|k)$ and z_1^r are

joint Gaussian random variables, hence, the estimator in (27) is unbiased, and the error variance of the estimator is

$$\begin{aligned} \mathrm{var}\left(\tilde{z}_2^r\right) &= E\left\{\left(z_2^r - \hat{z}_2^r\right)^2\right\} \\ &\cong E\left\{\left(x_{12} - \hat{x}_{12}\left(\Theta\left(z_1^r\right)\middle|\Theta\left(z_1^r\right)\right)\right)^2\right\} = \mathrm{var}\left(\tilde{x}_{12}\left(\Theta\left(z_1^r\right)\middle|\Theta\left(z_1^r\right)\right)\right) \end{aligned} \tag{28}$$

3.2.2 Estimating the arrival time at other nodes

Likewise, given z_1^r, the mean-squared estimate of the arrival time at node {3} is

$$\begin{aligned} \hat{z}_3^r &= E\left\{z_3^r \middle| z_1^r\right\} = E\left\{z_2^r + x_{23}\middle| z_1^r\right\} = E\left\{z_2^r \middle| z_1^r\right\} + E\left\{x_{23}\middle| z_1^r\right\} \\ &\cong \hat{z}_2^r + \hat{x}_{23}\left(\Theta\left(z_2^r\right)\middle|\Theta\left(z_1^r\right)\right) \end{aligned} \tag{29}$$

where $\hat{x}_{23}\left(\Theta\left(z_2^r\right)\middle|\Theta\left(z_1^r\right)\right)$ is the mean-squared *predicted estimate* of x_{23} at time $\Theta\left(z_2^r\right)$ given the measured traveling time $y_{23}\left(\Theta\left(z_1^r\right)\right)$ at time $\Theta\left(z_1^r\right)$.

It should be noted that the estimator in (29) cannot be realized since the value of z_2^r is not available at the time of route preplanning. To overcome the problem, we use the estimate $\hat{z}_2^r$, computed in (27), instead of z_2^r in (29). Therefore, the estimator in (29) can be approximated by

$$\hat{z}_3^r \cong \hat{z}_2^r + \hat{x}_{23}\left(\Theta\left(\hat{z}_2^r\right)\middle|\Theta\left(z_1^r\right)\right) \tag{30}$$

Later in this report, we will show that the error of estimation in (30) is negligible for appropriate selections of the sampling period T_S. Moreover, we will find a bound on the error variance of the estimator in (30).

Now, let's assume that node *i* precedes node *j* on route *r*. Let's also assume that given z_1^r the arrival time at node *i* on route *r* is estimated by $\hat{z}_i^r$. Given $\hat{z}_i^r$, the arrival time at node *j*, the immediate node after node *i* on route *r* can be estimated by

$$\begin{aligned}\hat{z}_j^r &= E\left\{z_j^r \middle| z_1^r\right\} = E\left\{z_i^r + x_{ij} \middle| z_1^r\right\} = E\left\{z_i^r \middle| z_1^r\right\} + E\left\{x_{ij} \middle| z_1^r\right\} \\ &\cong \hat{z}_i^r + \hat{x}_{ij}\left(\Theta\left(\hat{z}_i^r\right)\middle|\Theta\left(z_1^r\right)\right)\end{aligned} \tag{31}$$

We denote by $\tilde{z}_j^r$ the error of estimation in (31) which is defined as $\tilde{z}_j^r = z_j^r - \hat{z}_j^r$. In the following, we will show that if $\hat{z}_i^r$ is an unbiased estimate of the arrival time at node *i*, the error of the arrival time estimator of $\hat{z}_j^r$ in (31) will be negligible as the sampling period T_S becomes smaller.

Definition 1 (Bounded function): The discrete-time function $\mu: \mathbb{Z} \to \mathbb{Z}$ is said to be bounded in time if there exists a $\bar{\mu} < \infty$ such that $\left|\mu(k)\right| \le \bar{\mu}$, $\forall k = 0,1,2,\cdots$.

Proposition 1: Let route *r* start from node {1} and pass through nodes *i* and *j*, in that order. Let z_1^r be the departure time from node {1}, and $\hat{z}_i^r$ be an unbiased estimate of the arrival time at node *i* given z_1^r. Let also $h(k)$ be a discrete-time random process with $E\{h(k)\} = \mu(k)$, $\forall k = 0,1,2,\cdots$, where $\mu(k)$ is a bounded function for all $k \ge z_1^r$, then

$$E\left\{\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(z_i^r\right)-1} h(k)\right\} \cong \sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} \mu(k) \tag{32}$$

where the error of the approximation goes to zero as the sampling period T_S becomes smaller.

Proof:

$$E\left\{\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(z_i^r\right)-1} h(k)\right\} = E\left\{\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} h(k) + \sum_{k=\Theta\left(\hat{z}_i^r\right)}^{\Theta\left(z_i^r\right)-1} h(k)\right\} = E\left\{\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} h(k)\right\} + E\left\{\sum_{k=\Theta\left(\hat{z}_i^r\right)}^{\Theta\left(z_i^r\right)-1} h(k)\right\} \tag{33}$$

where, without loss of generality, in (33) we assumed that $\Theta\left(z_i^r\right) \ge \Theta\left(\hat{z}_i^r\right)$. The first term in (33) can be simplified as follows

$$E\left\{\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} h(k)\right\} = \sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} E\{h(k)\} = \sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} \mu(k) \tag{34}$$

To compute the second term in (33), we use the fact that for any two random variables *x* and *y*, we have (Papoulis [1991])

$$E\{g(x,y)\} = E\{E\{g(x,y)|x\}\} \tag{35}$$

Using (35), the second term in (33) can be written as

$$E\left\{\sum_{k=\Theta(\hat{z}_i^r)}^{\Theta(z_i^r)-1} h(k)\right\} = E\left\{E\left\{\left(\sum_{k=\Theta(\hat{z}_i^r)}^{\Theta(z_i^r)-1} h(k)\right)\middle| z_i^r\right\}\right\} = E\left\{\sum_{k=\Theta(\hat{z}_i^r)}^{\Theta(z_i^r)-1} \mu(k)\right\} \tag{36}$$

Since $\mu(k)$ is a bounded function for all $k \geq z_1^r$, (36) can be simplified as follows

$$E\left\{\sum_{k=\Theta(\hat{z}_i^r)}^{\Theta(z_i^r)-1} \mu(k)\right\} \leq E\left\{\sum_{k=\Theta(\hat{z}_i^r)}^{\Theta(z_i^r)-1} \bar{\mu}\right\} \tag{37}$$

Note that as the sampling period T_S becomes smaller, the last term in (37) can be approximated by

$$\begin{aligned} E\left\{\sum_{k=\Theta(\hat{z}_i^r)}^{\Theta(z_i^r)-1} \bar{\mu}\right\} &\cong E\left\{\int_{\hat{z}_i^r}^{z_i^r} \bar{\mu}\cdot d\left(T_S\cdot\tau\right)\right\} \\ &= \bar{\mu}.T_S\cdot E\left\{\int_{\hat{z}_i^r}^{z_i^r} d\tau\right\} = \bar{\mu}.T_S\cdot E\left\{z_i^r-\hat{z}_i^r\right\} = \bar{\mu}.T_S\cdot E\left\{\tilde{z}_i^r\right\} = 0 \end{aligned} \tag{38}$$

In other words, by appropriately selecting T_S and substituting (37) and (38) in (36) the second term in (33) becomes zero. Hence, using (34) and (33), (32) is obtained.♦

Proposition 2: Let $\hat{z}_i^r$ be an unbiased estimator of z_i^r. Then, with proper selection of the sampling period T_S, the arrival time estimator $\hat{z}_j^r$ in (31) will also be an unbiased estimator of z_j^r.

Proof:

Let $\tilde{z}_j^r$ denote the estimator error, which is defined as $\tilde{z}_j^r = z_j^r - \hat{z}_j^r$, and let T_S be small enough. The mean estimation error in (31) is

$$E\left\{\tilde{z}_j^r\right\} = E\left\{z_j^r - \hat{z}_j^r\right\} = E\left\{z_i^r + x_{ij} - \hat{z}_i^r - \hat{x}_{ij}\left(\Theta\left(\hat{z}_i^r\right)\middle|\Theta\left(z_1^r\right)\right)\right\} \tag{39}$$

Since $\hat{z}_i^r$ is an unbiased estimator of z_i^r, i.e., $E\left\{\tilde{z}_i^r\right\} = 0$, Equation (39) can be written as follows,

$$\begin{aligned} E\left\{\tilde{z}_j^r\right\} &= E\left\{x_{ij} - \hat{x}_{ij}\left(\Theta\left(\hat{z}_i^r\right)\middle|\Theta\left(z_1^r\right)\right)\right\} \\ &\cong E\left\{x_{ij}\left(\Theta\left(z_1^r\right)\right) + \sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(z_i^r\right)-1} u_{ij}(k) + \sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(z_i^r\right)-1} w_{ij}(k) - \hat{x}_{ij}\left(\Theta\left(z_1^r\right)\middle|\Theta\left(z_1^r\right)\right) - \sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} \eta_{ij}(k)\right\} \\ &= E\left\{\tilde{x}_{ij}\left(\Theta\left(z_1^r\right)\middle|\Theta\left(z_1^r\right)\right)\right\} + E\left\{\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(z_i^r\right)-1} u_{ij}(k)\right\} + E\left\{\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(z_i^r\right)-1} w_{ij}(k)\right\} - \sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} \eta_{ij}(k) \end{aligned} \tag{40}$$

where in deriving (40), we use (1) and (13). As discussed, the mean-squared filtered estimate $\hat{x}(k|k)$ is unbiased; i.e.,

$$E\left\{\tilde{x}_{ij}\left(\Theta\left(z_1^r\right)\middle|\Theta\left(z_1^r\right)\right)\right\} = 0 \tag{41}$$

Using Proposition 1 and Equations (2) and (3), Equation (40) can be simplified as follows,

$$\begin{aligned} E\left\{\tilde{z}_j^r\right\} &\cong \sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} E\left\{u_{ij}(k)\right\} + \sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} E\left\{w_{ij}(k)\right\} - \sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} \eta_{ij}(k) \\ &= \sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} \eta_{ij}(l) - \sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} \eta_{ij}(l) = 0 \end{aligned} \tag{42}$$

which indicates that our estimator in (31) is an unbiased estimator if T_S is properly selected.♦

In the following, we calculate the error variance of arrival time estimator $\hat{z}_j^r$ in (31).

Proposition 3: Let route *r* start from node {1} and pass through nodes *i* and *j*, in that order. Let z_1^r be the departure time from node {1}, and $\hat{z}_i^r$ be an unbiased estimate of the arrival time at node *i* given z_1^r. Let also $h(k)$ be a discrete-time random process with $E\{h(k)\} = \mu(k)$, and

$$E\{h(l)h(k)\} = \begin{cases} \delta^2(k) + \mu^2(k) & l = k \\ 0 & l \neq k \end{cases} \qquad \forall k, l = 0,1,2,\cdots \tag{43}$$

where $\delta(k)$ and $\mu(k)$ are bounded functions for all $k \geq z_1^r$, then

$$\mathrm{var}\left(\sum_{k=\Theta(z_1^r)}^{\Theta(z_i^r)-1} h(k) \right) \cong \sum_{k=\Theta(z_1^r)}^{\Theta(\hat{z}_i^r)-1} \delta^2(k) \tag{44}$$

where the error of the approximation goes to zero as the sampling period T_S becomes smaller.

Proof:

Using (43), we have

$$\begin{aligned} \mathrm{var}\left(\sum_{k=\Theta(z_1^r)}^{\Theta(z_i^r)-1} h(k) \right) &= \mathrm{var}\left(\sum_{k=\Theta(z_1^r)}^{\Theta(\hat{z}_1^r)-1} h(k) + \sum_{k=\Theta(\hat{z}_1^r)}^{\Theta(z_1^r)-1} h(k) \right) \\ &= \mathrm{var}\left(\sum_{k=\Theta(z_1^r)}^{\Theta(\hat{z}_1^r)-1} h(k) \right) + \mathrm{var}\left(\sum_{k=\Theta(\hat{z}_1^r)}^{\Theta(z_1^r)-1} h(k) \right) \end{aligned} \tag{45}$$

where in (45) we assumed that $\Theta(z_i^r) \geq \Theta(\hat{z}_i^r)$. The case $\Theta(z_i^r) < \Theta(\hat{z}_i^r)$ will be discussed later. The first term in (45) can be simplified as follows

$$\mathrm{var}\left(\sum_{k=\Theta(z_1^r)}^{\Theta(\hat{z}_i^r)-1} h(k) \right) = \sum_{k=\Theta(z_1^r)}^{\Theta(\hat{z}_i^r)-1} \mathrm{var}(h(k)) = \sum_{k=\Theta(z_1^r)}^{\Theta(\hat{z}_i^r)-1} \delta^2(k) \tag{46}$$

In deriving (46), we used (43). Using (35), the second term in (45) can be calculated as follows

$$\begin{aligned}\operatorname{var}\left(\sum_{k=\Theta\left(\hat{z}_i^r\right)}^{\Theta\left(z_i^r\right)-1} h(k)\right) &= E\left\{\left(\sum_{k=\Theta\left(\hat{z}_i^r\right)}^{\Theta\left(z_i^r\right)-1} h(k) - E\left\{\sum_{k=\Theta\left(\hat{z}_i^r\right)}^{\Theta\left(z_i^r\right)-1} h(k)\right\}\right)^2\right\} \\ &= E\left\{E\left\{\left(\sum_{k=\Theta\left(\hat{z}_i^r\right)}^{\Theta\left(z_i^r\right)-1} h(k) - E\left\{\sum_{k=\Theta\left(\hat{z}_i^r\right)}^{\Theta\left(z_i^r\right)-1} h(k)\right\}\right)^2 \middle| z_i^r\right\}\right\} \\ &= E\left\{\sum_{k=\Theta\left(\hat{z}_i^r\right)}^{\Theta\left(z_i^r\right)-1} \delta^2(k)\right\}\end{aligned} \tag{47}$$

Since $\delta(k)$ is bounded for all $k \geq z_1^r$, (47) can be computed as follows

$$E\left\{\sum_{k=\Theta\left(\hat{z}_i^r\right)}^{\Theta\left(z_i^r\right)-1} \delta(k)\right\} \leq E\left\{\sum_{k=\Theta\left(\hat{z}_i^r\right)}^{\Theta\left(z_i^r\right)-1} \left(\overline{\delta}\right)^2\right\} \cong 0 \tag{48}$$

where in obtaining (48) the same procedure used in (38) was applied here. Substituting (46), (47) and (48), equation (44) is obtained.

Recall that in (45), we assumed $\Theta\left(z_i^r\right) \geq \Theta\left(\hat{z}_i^r\right)$. We will show that (44) is also valid for the case $\Theta\left(z_i^r\right) < \Theta\left(\hat{z}_i^r\right)$. When $\Theta\left(z_i^r\right) < \Theta\left(\hat{z}_i^r\right)$, (45) can be written as follows

$$\begin{aligned}\operatorname{var}\left(\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(z_i^r\right)-1} h(k)\right) &= \operatorname{var}\left(\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} h(k) - \sum_{k=\Theta\left(z_i^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} h(k)\right) \\ &= \operatorname{var}\left(\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} h(k)\right) + \operatorname{var}\left(\sum_{k=\Theta\left(z_i^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} h(k)\right) - 2\operatorname{cov}\left(\sum_{l=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} h(l), \sum_{k=\Theta\left(z_i^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} h(k)\right)\end{aligned} \tag{49}$$

Repeating (47) and (48), the second term in the last row of (49) is approximately zero for small T_S, i.e.,

$$\operatorname{var}\left(\sum_{k=\Theta\left(z_i^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} h(k)\right) \cong 0 \tag{50}$$

The third term in (49) can be calculated as follows

$$\begin{aligned}&\operatorname{cov}\left(\sum_{l=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} h(l), \sum_{k=\Theta\left(z_i^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} h(k)\right)\\&= E\left\{\sum_{l=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} h(l) \cdot \sum_{k=\Theta\left(z_i^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} h(k)\right\} - E\left\{\sum_{l=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} h(l)\right\} \cdot E\left\{\sum_{k=\Theta\left(z_i^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} h(k)\right\}\end{aligned} \tag{51}$$

Using the same procedure in (37) and (38), and knowing that the random variables $h(l)$ and $h(k)$ are statistically uncorrelated for all $k \neq l$, (51) can be written as

$$\begin{aligned}&\operatorname{cov}\left(\sum_{l=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} h(l), \sum_{k=\Theta\left(z_i^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} h(k)\right) = E\left\{\sum_{l=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} h(l) \cdot \sum_{k=\Theta\left(z_i^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} h(k)\right\}\\&= E\left\{\sum_{k=\Theta\left(z_i^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} h(k).h(k)\right\} = E\left\{E\left\{\left(\sum_{k=\Theta\left(z_i^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} h(k).h(k)\right)\middle| z_i^r\right\}\right\}\\&= E\left\{\sum_{k=\Theta\left(z_i^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} \left(\delta^2(k)+\mu^2(k)\right)\right\}\\&\leq E\left\{\sum_{k=\Theta\left(z_i^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} \left(\left(\overline{\delta}\right)^2+\overline{\mu}^2\right)\right\} \cong 0\end{aligned} \tag{52}$$

where in obtaining (52) the same procedure in (37) and (38) is applied. By considering Equations (46) through (52), Equation (44) is obtained. ♦

Proposition 4: Given the assumptions in proposition 3, we have

$$E\left\{\tilde{z}_i^r . \left(\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(z_i^r\right)-1} h(k)\right)\right\} = E\left\{\tilde{z}_i^r . \left(\sum_{k=\Theta\left(\hat{z}_i^r\right)}^{\Theta\left(z_i^r\right)-1} \mu(k)\right)\right\} \leq \overline{\mu} T_S \operatorname{var}\left(\tilde{z}_i^r\right) \tag{53}$$

where $\tilde{z}_i^r = z_i^r - \hat{z}_i^r$ is the arrival time estimation error at node *i* on route *r*.

Proof:

$$E\left\{\tilde{z}_i^r.\left(\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(z_i^r\right)-1}h(k)\right)\right\}=E\left\{\tilde{z}_i^r.\left(\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1}h(k)+\sum_{k=\Theta\left(\hat{z}_i^r\right)}^{\Theta\left(z_i^r\right)-1}h(k)\right)\right\}$$
$$=E\left\{\tilde{z}_i^r.\left(\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1}h(k)\right)\right\}+E\left\{\tilde{z}_i^r.\left(\sum_{k=\Theta\left(\hat{z}_i^r\right)}^{\Theta\left(z_i^r\right)-1}h(k)\right)\right\} \quad (54)$$

where in (54), and without loss of generality, we assumed $z_i^r \geq \hat{z}_i^r$. Since $\hat{z}_i^r$ is an unbiased estimate of the arrival time z_i^r, the first term in (54) is zero, i.e.,

$$E\left\{\tilde{z}_i^r.\left(\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1}h(k)\right)\right\}=E\left\{\tilde{z}_i^r\right\}E\left\{\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1}h(k)\right\}=0 \quad (55)$$

The second term in (54) can be calculated as follows

$$E\left\{\tilde{z}_i^r.\left(\sum_{k=\Theta\left(\hat{z}_i^r\right)}^{\Theta\left(z_i^r\right)-1}h(k)\right)\right\}=E\left\{E\left\{\tilde{z}_i^r.\left(\sum_{k=\Theta\left(\hat{z}_i^r\right)}^{\Theta\left(z_i^r\right)-1}h(k)\right)\middle|z_i^r\right\}\right\}$$
$$=E\left\{\tilde{z}_i^r.\left(\sum_{k=\Theta\left(\hat{z}_i^r\right)}^{\Theta\left(z_i^r\right)-1}\mu(k)\right)\right\} \quad (56)$$

Since $\mu(k)$ is a bounded function for all $k \geq z_1^r$, we have

$$E\left\{\tilde{z}_i^r.\left(\sum_{k=\Theta\left(\hat{z}_i^r\right)}^{\Theta\left(z_i^r\right)-1}\mu(k)\right)\right\}\leq E\left\{\tilde{z}_i^r.\left(\sum_{k=\Theta\left(\hat{z}_i^r\right)}^{\Theta\left(z_i^r\right)-1}\overline{\mu}\right)\right\} \quad (57)$$

For small sampling period T_S, (57) can be approximated by

$$E\left\{\tilde{z}_i^r\cdot\left(\sum_{k=\Theta\left(\hat{z}_i^r\right)}^{\Theta\left(z_i^r\right)-1}\overline{\mu}\right)\right\}\cong E\left\{\tilde{z}_i^r\cdot\int_{\hat{z}_i^r}^{z_i^r}\overline{\mu}\cdot d\left(T_S\cdot\tau\right)\right\}$$
$$=\overline{\mu}.T_S\cdot E\left\{\tilde{z}_i^r\cdot\int_{\hat{z}_i^r}^{z_i^r}d\tau\right\}=\overline{\mu}.T_S\cdot E\left\{\tilde{z}_i^r\cdot\tilde{z}_i^r\right\}=\overline{\mu}.T_S\cdot\mathrm{var}\left(\tilde{z}_i^r\right) \quad (58)$$

In other words, by appropriately selecting T_S, equations (54) till (58) together result in (53). ♦

Proposition 5: Let $\hat{z}_i^r$ be an unbiased estimator of z_i^r, with error variance equal to $\mathrm{var}\left(\tilde{z}_i^r\right)$. Then, with proper selection of the sampling period T_S, the error variance of estimator $\hat{z}_j^r$ in (31) is bounded by

$$\mathrm{var}\left(\tilde{z}_j^r\right) \le \left(1+2\overline{\eta}_{ij}\right)\mathrm{var}\left(\tilde{z}_i^r\right) + \mathrm{var}\left(\tilde{x}_{ij}\left(\Theta\left(z_1^r\right)\middle|\Theta\left(z_1^r\right)\right)\right) + \sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1}\sigma_{ij}^2(k) + \sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} q_{ij}^2(k) \qquad (59)$$

Proof:

The error variance of the estimator in (31) is

$$\begin{aligned}\mathrm{var}\left(\tilde{z}_j^r\right) &= \mathrm{var}\left(z_j^r - \hat{z}_j^r\right) = \mathrm{var}\left(z_i^r + x_{ij} - \hat{z}_i^r - \hat{x}_{ij}\left(\Theta\left(\hat{z}_i^r\right)\middle|\Theta\left(z_1^r\right)\right)\right)\\ &= \mathrm{var}\left(\tilde{z}_i^r\right) + \mathrm{var}\left(x_{ij} - \hat{x}_{ij}\left(\Theta\left(\hat{z}_i^r\right)\middle|\Theta\left(z_1^r\right)\right)\right) + 2\,\mathrm{cov}\left(\tilde{z}_i^r, \left(x_{ij} - \hat{x}_{ij}\left(\Theta\left(\hat{z}_i^r\right)\middle|\Theta\left(z_1^r\right)\right)\right)\right)\end{aligned} \qquad (60)$$

The second term in (60) can be written as

$$\begin{aligned}&\mathrm{var}\left(x_{ij} - \hat{x}_{ij}\left(\Theta\left(\hat{z}_i^r\right)\middle|\Theta\left(z_1^r\right)\right)\right)\\ &\cong \mathrm{var}\left(x_{ij}\left(\Theta\left(z_1^r\right)\right) + \sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(z_i^r\right)-1} u_{ij}(k) + \sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(z_i^r\right)-1} w_{ij}(k) - \hat{x}_{ij}\left(\Theta\left(\hat{z}_1^r\right)\middle|\Theta\left(z_1^r\right)\right) - \sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1}\eta_{ij}(k)\right) \qquad (61)\\ &= \mathrm{var}\left(\tilde{x}_{ij}\left(\Theta\left(\hat{z}_1^r\right)\middle|\Theta\left(z_1^r\right)\right)\right) + \mathrm{var}\left(\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(z_i^r\right)-1} u_{ij}(k)\right) + \mathrm{var}\left(\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(z_i^r\right)-1} w_{ij}(k)\right)\end{aligned}$$

where in deriving (61) we use (1) and (13). By using Proposition 3 and equations (2) and (3), (61) can be simplified as

$$\mathrm{var}\left(x_{ij} - \hat{x}_{ij}\left(\Theta\left(\hat{z}_i^r\right)\middle|\Theta\left(z_1^r\right)\right)\right) \cong \mathrm{var}\left(\tilde{x}_{ij}\left(\Theta\left(z_1^r\right)\middle|\Theta\left(z_1^r\right)\right)\right) + \sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1}\sigma_{ij}^2(k) + \sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1} q_{ij}^2(k) \qquad (62)$$

The last term in (60) can be calculated as follows

$$\begin{aligned}&\operatorname{cov}\left(\tilde{z}_i^r.\left(x_{ij}-\hat{x}_{ij}\left(\Theta\left(\hat{z}_i^r\right)\middle|\Theta\left(z_1^r\right)\right)\right)\right)\\&\quad=E\left\{\tilde{z}_i^r.\left(x_{ij}-\hat{x}_{ij}\left(\Theta\left(\hat{z}_i^r\right)\middle|\Theta\left(z_1^r\right)\right)\right)\right\}-E\left\{\tilde{z}_i^r\right\}E\left\{x_{ij}-\hat{x}_{ij}\left(\Theta\left(\hat{z}_i^r\right)\middle|\Theta\left(z_1^r\right)\right)\right\}\end{aligned}\tag{63}$$

Knowing that the estimator $\hat{z}_i^r$ is an unbiased estimator of z_i^r, the second term on the right hand side of (63) is zero. Therefore,

$$\operatorname{cov}\left(\tilde{z}_i^r.\left(x_{ij}-\hat{x}_{ij}\left(\Theta\left(\hat{z}_i^r\right)\middle|\Theta\left(z_1^r\right)\right)\right)\right)=E\left\{\tilde{z}_i^r.\left(x_{ij}-\hat{x}_{ij}\left(\Theta\left(\hat{z}_i^r\right)\middle|\Theta\left(z_1^r\right)\right)\right)\right\}\tag{64}$$

where by using (1) and (13) we have

$$\begin{aligned}&E\left\{\tilde{z}_i^r.\left(x_{ij}-\hat{x}_{ij}\left(\Theta\left(\hat{z}_i^r\right)\middle|\Theta\left(z_1^r\right)\right)\right)\right\}\\&\quad\cong E\left\{\tilde{z}_i^r.\left(\tilde{x}_{ij}\left(\Theta\left(z_1^r\right)\middle|\Theta\left(z_1^r\right)\right)+\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(z_i^r\right)-1}u_{ij}(k)+\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(z_i^r\right)-1}w_{ij}(k)-\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1}\eta_{ij}(k)\right)\right\}\\&\quad=E\left\{\tilde{z}_i^r.\tilde{x}_{ij}\left(\Theta\left(z_1^r\right)\middle|\Theta\left(z_1^r\right)\right)\right\}+E\left\{\tilde{z}_i^r.\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(z_i^r\right)-1}u_{ij}(k)\right\}\\&\qquad\qquad+E\left\{\tilde{z}_i^r.\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(z_i^r\right)-1}w_{ij}(k)\right\}-E\left\{\tilde{z}_i^r\right\}\cdot\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(\hat{z}_i^r\right)-1}\eta_{ij}(k)\end{aligned}\tag{65}$$

Since $\tilde{z}_i^r$ and $\tilde{x}_{ij}\left(\Theta\left(z_1^r\right)\middle|\Theta\left(z_1^r\right)\right)$ are statistically uncorrelated and both are unbiased, the first and last terms in (65) are zero. Using Proposition 4 and equations (2) and (3), the second and third terms in (65) can be calculated as follows

$$E\left\{\tilde{z}_i^r.\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(z_i^r\right)-1}u_{ij}(k)\right\}=E\left\{\tilde{z}_i^r.\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(z_i^r\right)-1}\eta_{ij}(k)\right\}\le\overline{\eta}_{ij}\cdot T_S\cdot\operatorname{var}\left(\tilde{z}_i^r\right)\tag{66}$$

$$E\left\{\tilde{z}_i^r.\sum_{k=\Theta\left(z_1^r\right)}^{\Theta\left(z_i^r\right)-1}w_{ij}(k)\right\}=0\tag{67}$$

Therefore by substituting (66) and (67) in (65), and subsequently in (64), we have

$$\operatorname{cov}\left(\tilde{z}_i^r.\left(x_{ij}-\hat{x}_{ij}\left(\hat{z}_i^r\left|z_1^r\right.\right)\right)\right)=E\left\{\tilde{z}_i^r.\sum_{k=\Theta\left(\hat{z}_1^r\right)}^{\Theta\left(z_i^r\right)-1}\eta_{ij}(k)\right\}\le\overline{\eta}_{ij}\cdot T_S\cdot\operatorname{var}\left(\tilde{z}_i^r\right) \tag{68}$$

Using (60), (62) and (68), the bound on error variance of the estimator in (31) can be obtained which is given by (59).♦

In summary, in this section we have developed a methodology to estimate the arrival time at each node of a transportation network. More precisely, given z_1^r, departure time from node {1}, the arrival time at each node *i* on route *r* can be estimated using the developed arrival time estimator in (31) and the predicted traveling times on arcs in (13), recursively, in the following order:

$$z_1^r, x_{12}^r\left(\Theta\left(z_1^r\right)\middle|\Theta\left(z_1^r\right)\right)\xrightarrow{\text{using (31)}}\hat{z}_2^r\xrightarrow{\text{using (13)}}x_{23}^r\left(\Theta\left(\hat{z}_2^r\right)\middle|\Theta\left(z_1^r\right)\right)\to\cdots. \tag{69}$$

We have shown in Propositions 2 and 5 that by properly selecting the sampling period T_S, the arrival time estimator in (31) is an unbiased estimator, and its error variance is bounded by (59). Note that by using Equations (16) and (59), the bound on the error variance of the arrival time at each node *j* on route *r* can be also written as

$$\operatorname{var}\left(\tilde{z}_j^r\right)\le\left(1+2\overline{\eta}_{ij}\right)\operatorname{var}\left(\tilde{z}_i^r\right)+\operatorname{var}\left(\tilde{x}_{ij}\left(\Theta\left(\hat{z}_i^r\right)\middle|\Theta\left(z_1^r\right)\right)\right) \tag{70}$$

Note also that given z_1^r, the error variance of the departure time from node {1} is zero, i.e., $\operatorname{var}\left(\tilde{z}_1^r\right)=0$, which together with the given $\operatorname{var}\left(\tilde{x}_{ij}\left(0\middle|0\right)\right)=p_{ij}(0)$ initializes equations (16), (21) , and (70). Hence, the bound of the error variance of the arrival time at each node *j* on route *r* can be calculated, recursively.

As seen from (70), the bound on the error variance of the arrival time at each node *j*, $\operatorname{var}\left(\tilde{z}_j^r\right)$, depends on two elements: 1) the error variance of arrival time estimator at the previous node, $\operatorname{var}\left(\tilde{z}_i^r\right)$, and 2) the error variance of traveling time predictor on arc (i,j) at time $\hat{z}_i^r$, i.e, $\operatorname{var}\left(\tilde{x}_{ij}\left(\Theta\left(\hat{z}_i^r\right)\middle|\Theta\left(z_1^r\right)\right)\right)$. This fact indicates that the error variance increases sharply as we

estimate the arrival time at the nodes further down on the route. It also implies that as we predict the traveling time in the distant future, the error variance in (70) increases abruptly.

3.3 Partial Routing

In the previous section, we developed a methodology to estimate the arrival times at nodes of a transportation network. In this section, we will use the estimated arrival times to plan partial routes. The routes are generated such that the error variance of the arrival time at each node of the route is guaranteed to be less than a desired level which is called the service level, hereafter.

Let's consider graph *G* as shown in Figure 1, and let *r* be a route in graph *G*.

Definition (β efficient route): Let random variable z_i^r denote the arrival time at node *i* taking route *r*. The route *r* is said to be *β efficient* if the error variance of the arrival time estimator $\hat{z}_i^r$ at all nodes $i \in r$ be less than or equal to an arbitrary positive real number β, i.e.,

$$\mathrm{var}\left(\tilde{z}_i^r\right) \le \beta \tag{71}$$

It should be noted that the value of $\mathrm{var}\left(\tilde{z}_i^r\right)$ represents the error in estimating the arrival time at each node *i*. Small values of $\mathrm{var}\left(\tilde{z}_i^r\right)$ implies more confidence in visiting node *i* at time $\hat{z}_i^r$ taking route *r*. Therefore, by choosing β in (71), we determine the desired confidence level in visiting node *i*. In the following, given a network, we use dynamic programming techniques to generate β efficient partial routes.

3.3.1 Dynamic Programming

Let $G:(N,A)$ be a graph with node set $N=\{1,..,i,j,..,n\}$ and arc set $A=\{(i,j)|i,j\in N\}$. Let the state (*S*,*i*) be defined as $S \subseteq N$ is an unordered set of visited nodes, and $i \in S$ is the last visited node. Associated to each state (*S*,*i*) are:

1. the estimated and error variance (denoted by $\hat{z}_i^S$ and $\text{var}\left(\tilde{z}_i^S\right)$, respectively) of the arrival time at node *i* taking the path starting from node {1} passing through every node of *S* exactly once and ending at node $i \in S$, and
2. a cost denoted by C_i^S, defined as the cost of traveling on the path described above.

Definition (acceptable arc): Let the error variance of arrival time at node $i \in S$ meet the required service level, i.e., $\text{var}\left(\tilde{z}_i^S\right) \le \beta$. An arc $(i,j) \in A$ is said to be 'acceptable' if the error variance of arrival time at node *j* traveling on arc (*i*,*j*) also meets the required service level.

Definition (state extendibility): Let node $i \in S$ be the last visited node of the state (*S*,*i*). The state *S* is said to be expandable to node *j*, $j \in N$ and $j \notin \{S\}$, if the arc $(i,j) \in A$ is 'acceptable'.

Having defined a state and state extendibility, we can now use a dynamic programming algorithm to generate partial routes as follows. Let's assume that the state (*S*,*i*) has been generated, i.e., route *S* is β efficient. At each state (*S*,*i*), the algorithm looks for uncovered nodes $j \in N$ and $j \notin \{S\}$. Node *j* will be added to state *S*, if state *S* can be extended to node *j*. Associated with each (*S*,*i*) are $\hat{z}_i^S$ and $\text{var}\left(\tilde{z}_i^S\right)$. Given $\hat{z}_i^S$ and $\text{var}\left(\tilde{z}_i^S\right)$, the estimate and error variance of arrival time at node *j* are calculated using equations (31) and (70). The procedure continues until all β efficient routes are generated. Note that, in general, a β efficient route does not cover all nodes in *N*. That is why we called these routes partial routes. We define max-desired-level as the maximum number of stops that we need to take into consideration in the plan.

In order to reduce the computational time, the following elimination test is performed which is called *state elimination test*. Given generated states (S_1,*i*) and (S_2,*i*), where S_1 and S_2 cover the same set of nodes in *N*, state S_2 is eliminated if $C_i^{S_1} \le C_i^{S_2}$ (and for random variables $E\left\{C_i^{S_1}\right\} \le E\left\{C_i^{S_2}\right\}$).

The following is the summary of the dynamic programming algorithm described above.

Step 1 Level l=1, $\text{var}\left(\tilde{z}_1\right) = 0$ and $\text{var}\left(\tilde{x}_{1i}\left(0\middle|0\right)\right) = p_{1i}\left(0\right)$

(Initialization): For all extendable nodes $i \in N/\{1\}$ from node {1}:

1) Generate (S,i): $S=\{1,i\}$,

2) Compute $\hat{z}_i^S$, $\text{var}\left(\tilde{z}_i^S\right)$, C_i^S according to (31) and (70).

Step 2: Level $l=l+1$,

(During): For all states (S,i) at level $l-1$,

Perform state elimination test

For all extendable nodes $j \in N/\{S\}$ from state (S,i):

1) Generate $(S \cup \{j\},j)$

2) Predict the traveling time $\hat{x}_{ij}\left(\Theta\left(\hat{z}_i^r\right)\middle|\Theta\left(z_1^r\right)\right)$, and compute the error variance of predictor, i.e., $\text{var}\left(\tilde{x}_{ij}\left(\Theta\left(\hat{z}_i^r\right)\middle|\Theta\left(z_1^r\right)\right)\right)$ using (13) and (16).

3) Compute $\hat{z}_j^S$, $\text{var}\left(\tilde{z}_j^S\right)$, C_j^S according to (31) and (70).

Step 3 (Termination): If $l<$max-desired-level and there are still some nodes left to be added to the states go to step 2. Otherwise exit the algorithm, all partial routes have been generated.

3.4 Moving along the Best Route

In the previous section, we used dynamic programming techniques to generate all partial routes in a given transportation network. The objective was to find routes for which the error variances of arrival times at their nodes are guaranteed to be less than or equal to a desired level. We called these routes β efficient partial routes. Recall that to each partial route r, a cost is associated which represents the cost of traveling on the path starting from node {1} passing through every node of r exactly once, in the specified order, and ending at the last node in r. We denote by $|r|$, the cardinality of route r, which indicates the number of nodes in that route.

Among all β efficient partial routes generated in Section 3.3, we select a *good* one to move along. It should be noted that in selecting a partial route, there are multiple, sometimes conflicting, criteria to be considered. For instance, minimum cost partial route could be a criterion. The other criterion could be maximum $|r|$. Note, however, that in a typical network, the minimum cost route would be the one with least number of covered nodes. In other words, the more nodes we visit on a typical partial route, the more cost we endure.

In this report, we select the partial route with the highest cardinality at least cost. More precisely, we sort all partial routes based on their cardinality. Among those with highest cardinality, we select the one with the lowest cost. Hereafter, this route is called the best partial route and is denoted by r^*. The rational for adopting this selection method is to visit more nodes at least cost.

Note that as we move along the best partial route, updated information of traffic conditions are becoming available. Thus, instead of visiting all nodes on the best partial route, r^*, we will visit a predetermined number of nodes on r^* before using the updated information. Later, in this report, we will investigate the sensitivity of the solutions to this predetermined number.

As, we move along the partial routes, we will incorporate the newly available information of the traveling time on arcs and arrival time at nodes into the algorithm given in Section 3.3. That is, the updated information of traveling time is used to correct the filtered estimator in (17) and, consequently, update the predictor in (13). As we reach the last node among the predetermined number of nodes on the best partial route, the updated information of arrival time at that node is used to update the estimate of arrival time and recalculate the error variance of arrival time at other nodes. And finally, a new β efficient partial routes are generated based on the algorithm provided in Section 3.3. This procedure continues until all nodes in the network are visited.

4 Simulation Experiments

In this section, we perform a variety of simulation experiments on selected transportation networks to evaluate the efficiency of the developed partial routing algorithm and to compare the results of the developed algorithm with the solutions obtained by using the existing vehicle dispatching and routing policies. More precisely, we compare the Partial Routing Algorithm (PRA) with the Nearest Dispatching Algorithm (NDA) and Full Routing Algorithm (FRA).

In NDA, the vehicle routing policy is very local. At each node, the vehicle is dispatched to the closest uncovered node. To determine the closest uncovered node, the dispatcher uses the most updated traveling costs between the current node and all remaining (not visited) nodes of the network. This algorithm continues until all nodes are visited.

In contrast to NDA, the FRA is a global routing algorithm. In fact, FRA searches for the global minimum. To find the optimal solution, the algorithm needs to have complete information about the traveling costs between all nodes of the network a priori. For a deterministic network, the FRA basically finds the solution for the Traveling Salesman Problem (TSP). The TSP is NP-hard; therefore, any full routing algorithm will be computationally very slow. The FRA, in this report, is developed based on dynamic programming method similar to that described in Section 3.3.1. It should be noted that, in FRA, once the route is planned any changes in the traveling costs would not alter the preplanned route. Therefore, although FDA leads to global minimum in deterministic static networks, the solution may not be a global minimum in dynamic stochastic networks. For dynamic stochastic networks in this report, we will use the expected costs between the nodes of the network in computing the route in FRA.

In the following, the solutions obtained from NDA, FRA and PRA are evaluated and compared. All algorithms are coded in Matlab 6.5 developed by MathWorks, Inc., and tested on an Intel Pentium 4, 2.2 GHz.

4.1 Simulation scenario 1 (Deterministic routing)

In this simulation scenario, a deterministic network is considered and the costs of the generated routes by NDA and FRA are compared. Note that PRA cannot be evaluated in deterministic network since equation (19) will be undefined.

Let's consider a simple transportation network as shown by graph *G* in Figure 3, which consists of 10 nodes. The solid lines in Figure 3 demonstrate the direct connection between nodes. Numbers adjacent to solid lines indicate the traveling times between each two connected nodes. For instance, the traveling time between nodes {2} and {4} is 24 minutes; and between nodes {3} and {6} is 32 minutes. We assume that graph G is fully connected. Nodes, which are not connected by solid lines, are assumed to be connected via intermediate nodes. Therefore, the traveling time between nodes {2} and {7} in Figure 3 is 42 minutes, which is the shortest traveling time between these two nodes passing through node {4}.

For the sake of simplicity, we assume that graph *G* is symmetric, i.e., the traveling time between nodes *i* and *j* is equal to the traveling time between nodes *j* and *i*. It should be noted that the proposed algorithm is general and it is not limited to symmetric networks.

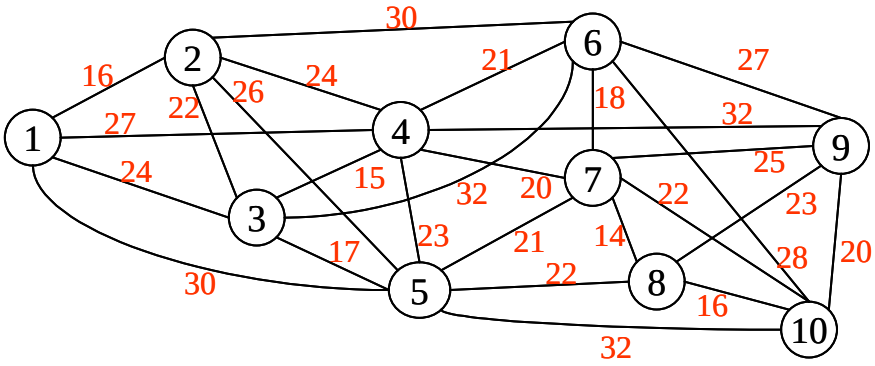

Figure 3: Simple transportation network *G* used in the simulation experiments.

Without loss of generality, we assume that node {1} is the origin node. Table 1 shows the results of using NDA and FRA for finding the best route in the deterministic network illustrated in Figure 3. The cost of traveling on the best route found by each algorithm is given in column 3 in Table 1. We assume that the cost of each route is equal to the traveling time on that route in minutes. Table 1 column 4 presents the CPU time, which is the time in seconds used by each algorithm to obtain the best route.

Table 1: Comparing NDA and FRA for deterministic networks.

Algorithm	Best route found	Cost of the route [min]	CPU time [sec]
NDA	r={1 2 3 4 7 8 10 9 6 5}	180	0.05
FRA	r={1 2 5 3 4 6 7 8 10 9}	163	90.4

Since the network is deterministic and all information is available in advance, the FRA has found the optimal route. As seen, the FRA is computationally slow since it is an exponential algorithm. Table 1 also shows that the NDA provides a local minimum solution in a very short amount of time.

4.2 Simulation Scenario 2 (Stationary stochastic routing)

In this simulation scenario, we consider a stationary stochastic network. It is stationary in the sense that the characteristics of the network given by Equations (2) to (4) are independent of time *k*. Without loss of generality, we assume that the historical changes in the traveling time between nodes *i* and *j*, u_{ij}, at any time is zero. Given the above assumptions, the network characteristics in Equations (2) to (4) can be rewritten as follows

$$E\{u_{ij}(k)\}=0; \qquad E\{u_{ij}(k)u_{ij}(l)\}=\begin{cases}\sigma_{ij}^2 & k=l\\ 0 & k\neq l\end{cases} \tag{72}$$

$$E\{w_{ij}(k)\}=0; \qquad E\{w_{ij}(k)w_{ij}(l)\}=\begin{cases}q_{ij}^2 & k=l\\ 0 & k\neq l\end{cases} \tag{73}$$

$$E\{v_{ij}(k)\}=0; \qquad E\{v_{ij}(k)v_{ij}(l)\}=\begin{cases}r_{ij}^2 & k=l\\ 0 & k\neq l\end{cases} \tag{74}$$

Note that Equations (72) and (73) together with the dynamic model in (1) necessitate the mean traveling time between each two nodes *i* and *j* be constant in time equal to its initial value $x_{ij}(0)$, i.e.,

$$E\left\{x_{ij}(k)\right\} = x_{ij}(0) \quad \forall k = 1, 2, \cdots \tag{75}$$

Let's consider graph *G* given in Figure 3 again, where the numbers adjacent to solid lines are assumed to be the initial value of traveling times, i.e., at time zero. According to (75), the mean traveling times at any time $k>0$ would be equal to its initial value.

In this simulation experiment, we vary the uncertainties in the transportation network given in Figure 3 to evaluate and compare the impact of increasing uncertainties on the cost of the routes generated by the three routing methodologies: NDA, FRA, and PRA. To measure the uncertainty of the network, we use the concept of *coefficient of variation* (COV) of each arc. The COV of arc (*i*,*j*) is defined by

$$COV_{ij} = \frac{\sigma_{ij} + q_{ij}}{x_{ij}(0)} \times 100\% \tag{76}$$

Let's assume that the ratios $\sigma_{ij}/x_{ij}(0)$, $q_{ij}/x_{ij}(0)$, and $r_{ij}/x_{ij}(0)$ for all arcs (*i*,*j*) of the network *G* are constants. Consequently, the COV of all the arcs of the network given by (76) are equal which, hereafter, referred to as the COV of the network for the sake of simplicity.

Table 2 shows the costs of routes generated by the NDA, FRA and PRA when the COV of the stationary stochastic network *G* in Figure 3 varies. We assume that the cost of each route is equal to the traveling time on that route in minutes. For each simulation experiment, we change the uncertainty (COV) of network *G* by altering the values of $\sigma = \sigma_{ij}/x_{ij}(0)$, $q = q_{ij}/x_{ij}(0)$, and $r = r_{ij}/x_{ij}(0)$, as shown in Table 2 columns 2, 3, and 4, respectively.

For each simulation experiment, the network in Figure 3 was generated 25 times using a Guassian random number generator with parameters given by (72) to (75). Table 2 shows the average (mean) and the standard deviation (STD) of routing costs based on the results of the 25 trials. Table 2 also compares the CPU time needed by each methodology to generate the best route.

As discussed earlier, the FRA is an offline preplanning routing algorithm. In this report, the mean travel times on arcs of the network is used for advance routing. Recall from (75) that we assumed the mean traveling times would be equal to its initial value given in Figure 3. Therefore,

the best route generated by the FRA for the simulation scenario 1 would be the best route for the simulation scenario 2, too. Thus, the route generated in simulation scenario 1 and given in Table 1 is used here. Given by Table 2 columns 9 and 10 are the mean and STD of the costs of the preplanned route evaluated using the data of the 25 generated simulation trials. Note that, the CPU-time needed to generate the best route for the FRA is the same as that in simulation experiment 1 in Table 1 (i.e., 90.4). Hence, it is not repeated in Table 2.

Table 2 columns 11 and 12 present the mean and standard deviation (STD) of the costs of routes generated by the PRA based on the results of the 25 trials. We assumed the following specifications for the PRA. The sampling time is 5 minutes, the maximum level of route planning is 5; and the maximum number of nodes visited on each partial route is 2. The maximum level of route planning represents the number of stops that we need to take into consideration in the plan. The maximum number of nodes visited on each partial route is the maximum number of stops the vehicle visits before the route is updated. The last column of Table 2 shows the value of parameter β in (71) used in each simulation experiment. Note that as the COV of the network increases, the error variance of arrival time at each node of the network increases too. Thus, to find good partial routes, we increase the parameter β as the COV of the network increases.

Table 2: Comparing different routing algorithms in a stationary stochastic network.

				NDA				FRA		PRA				
				cost		CPU time		cost		Cost		CPU time		
COV	σ%	q %	r %	Mean	STD	Mean	STD	Mean	STD	Mean	STD	Mean	STD	β
0.5	0.25	0.25	0.5	180	0.19	0.05	0.01	162.9	0.21	173	0.21	60.7	0.2	2
1	0.5	0.5	1	180	0.37	0.05	0.01	163	0.29	173	0.38	59.1	1.48	3
2	1	1	1	180.1	0.72	0.05	0.01	163	0.95	172.8	0.97	58.9	0.21	6
5	2.5	2.5	1	180.6	2.52	0.05	0.01	163	2.43	172.5	2.44	59.0	0.1	15
10	5	5	2	179.6	5.53	0.05	0.01	162.7	4.95	170.4	4.59	59.1	0.07	30
20	10	10	2	176	9.94	0.05	0.01	161.5	7.2	168.9	6.85	54.4	0.1	50
50	30	20	5	161.6	11.96	0.05	0.01	164	16.9	159.8	14.2	45.2	0.21	100
75	40	3.5	5	137.8	24.4	0.05	0.01	163.6	32.5	132.8	21.0	37.4	0.75	125

Table 2 indicates that when the uncertainty (COV) in network *G* is relatively low (e.g., COV≤20), the FRA generates the best route in average. However, as the COV of the network increases other routing methods outperform the FRA. The results in Table 2 signify that for networks with high level of uncertainties (e.g., COV>20) the routes obtained by PRA is the least-cost routes in average. Interestingly, under high level of uncertainties, routes generated by NDA, which is a very local algorithm, excel the optimal solution of the FRA in average.

Table 2 shows that the NDA provides a route in a very short amount of time, and that the mean CPU time for the PRA to find the best route declines as the COV of the network increases. We have observed that the quality of the solution and the CPU time to find the solution by PRA is very much dependent on the PRA parameters including β. A detailed sensitivity analysis to study the effect of the variation of these parameters on the cost of solution and the CPU time is conducted and presented in the following simulation scenario.

4.3 Simulation scenario 3 (Sensitivity analysis)

In this simulation scenario, we investigate the sensitivity of the cost of solutions and CPU time to generate these solutions by the PRA to the changes of its parameters. We consider the Simulation Scenario 2 with a COV of 50.

Table 3 presents the average cost of routes generated by the PRA when the parameter β varies from 40 to 160. The table also shows the average CPU-time needed to generate these routes. We still assume that the sampling time is 5 minutes; the maximum level of route planning is 5; and the maximum number of nodes visited on each partial route is 2. Table 3 shows the average of the results of the 25 trials.

Table 3: Sensitivity of the PRA to changes of parameter β.

	Cost		**CPU time**	
β	**Mean**	**STD**	**Mean**	**STD**
40	160.5	12.9	11.9	0.55
55	160.4	14.7	17.2	0.68
70	158.7	16.0	27.7	0.60
85	158.4	14.9	37.0	0.45

100	151.4	15.1	43.9	0.35
115	150.1	15.3	50.1	0.28
130	150	15.3	54.5	0.42
145	150	15.3	56.9	0.35
160	150	15.3	58.2	0.26

The results in Table 3 indicate that as parameter β increases, the PRA generates better routes in average, however, at the cost of increasing CPU-time. Note that, for relatively small value of β many good routes may be discarded due to (71). As β increases, more routes satisfy the β efficiency test in (71) and will be considered for being selected as a final solution.

As seen from Table 3, having fixed all other parameters of the PRA, there exits a β^* which increasing the parameter β beyond that level, will not improve the quality of the result found for a specific graph *G*. For the Simulation Scenario 2 with COV=50 and the specified specifications, the β^* is about 115.

Table 4 shows the sensitivity analysis of solutions generated by the PRA versus the changes in the maximum level of route planning. In this simulation experiment the maximum level of routing is varied between 2 to 9. We assume that the sampling time is 5 minutes; the maximum number of nodes visited on each partial route is 2; and the parameter β is 100. Table 4 shows the average of the results of the 25 trials.

Table 4: Sensitivity of the PRA to the changes of the maximum level of routing (MLV).

	Cost		**CPU time**	
MLV	**Mean**	**STD**	**Mean**	**STD**
2	160.6	14.7	1.8	0.05
3	162.1	13.2	7.9	0.04
4	160.2	15.7	13.3	0.07
5	151.4	15.1	44.2	0.34
6	152	14.7	48.9	0.71
7	1152	14.7	48.9	0.72
8	152	14.7	48.9	0.72
9	152	14.7	48.9	0.73

Table 4 indicates that as the maximum level of route planning increases, the PRA generates better routes in average. With an increment in maximum level, the algorithm, in fact, looks further down the road. In other words, the algorithm behaves more globally than locally, which in turn improves the quality of the obtained solutions. However, this benefit is obtained at the price of dramatically increasing the CPU-time.

Table 5 presents the impact of changes in the maximum number of nodes visited on each partial route on the quality of the generated routes by the PRA. Here, the maximum number of nodes visited is changed from 1 to 5. We assume that the sampling time is 5 minutes; the maximum level of route planning is 5; and the parameter β is 100. Table 5 shows the average of the results of the 25 trials.

Table 5: Sensitivity of the PRA to changes in the maximum number of nodes visited on partial route (MN).

	Cost		**CPU time**	
MN	**Mean**	**STD**	**Mean**	**STD**
1	150.9	18.1	67.2	0.5
2	151.4	15.1	44.2	0.33
3	157.4	16.0	37.6	0.11
4	160.8	14.8	35.0	0.08
5	168.5	18.6	34.0	0.07

As indicated by the results in Table 5, as the maximum number of visited nodes decreases, the PRA generates better routes in average. Note that, with a decrement in this number, the algorithm will look for new routes more frequently, thus correcting and updating the generated solutions based on updated information obtained from the transportation network. Therefore, as the maximum number of visited nodes increases, the CPU-time decreases.

4.4 Simulation Scenario 4 (Dynamic stochastic routing)

In this simulation scenario, we consider a dynamic stochastic network in order to evaluate the efficiency of the developed algorithm and to compare the solutions generated by the NDA, FRA and PRA when the characteristics of the network change in time.

Let's consider graph *G* shown in Figure 3. We assume that the numbers adjacent to arc (*i*,*j*) are the initial value of traveling times on that arc, i.e., $x_{ij}(0)$. Let's, for the time being, assume that for any arc (*i*,*j*) in Equations (2) and (3), we have $\sigma_{ij}(k)=0$ and $q_{ij}(k)=0$, $\forall k=0,1,2,\cdots$. Without loss of generality, the value of $\eta_{ij}(k)$ in (2) is selected in such a way that the traveling time on arc (*i*,*j*) has a sinusoidal-like wave form as shown in Figure 4. To measure the dynamism of the network, we use the concept of *degree of dynamism* (DOD) of each arc. For the sinusoidal-like mean travel time given in Figure 4, the DOD of arc (*i*,*j*) when $\sigma_{ij}(k)$ and $q_{ij}(k)$ are zero $\forall k=0,1,2,\cdots$, is defined by

$$DOD_{ij}=\frac{\sup\left|x_{ij}(k)-x_{ij}(0)\right|}{x_{ij}(0)}\times 100\% \qquad (77)$$

We assume that the DOD of all arcs of the network *G* are equal which, hereafter, referred to as the DOD of the network. In Figure 4 the DOD of the network *G* is denoted by μ.

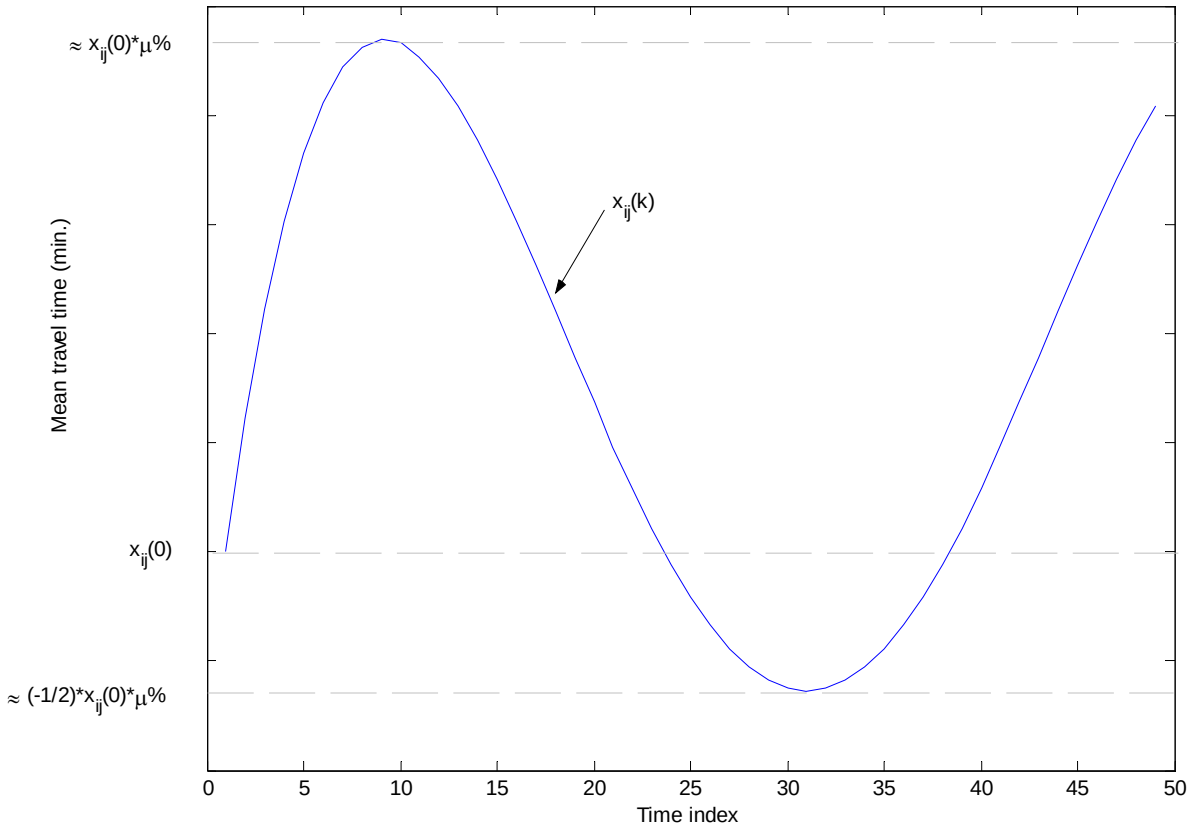

Figure 4: The assumed mean travel time on arc (*i*,*j*) when $\sigma_{ij}(k)$ and $q_{ij}(k)$ are zero.

In this simulation scenario, after determining the values of $\eta_{ij}(k)$, we consider the following characteristics for the network parameters. We assume that the ratios $\sigma_{ij}(k)/\eta_{ij}(k)$, $q_{ij}(k)/x_{ij}(0)$, and $r_{ij}(k)/x_{ij}(0)$ for all arcs (*i*,*j*) of the network *G* and $\forall k = 0,1,2,\cdots$ are constants. These ratios are denoted by σ, *q*, and *r*, respectively, hereafter.

In this simulation scenario, we vary the uncertainties (i.e., σ, *q*, and *r*) and the degree of dynamism (i.e., μ in Figure 4) in the transportation network given in Figure 3 to evaluate and compare the cost of the solutions generated by the three routing methodologies: NDA, FRA, and PRA. Table 6 summarizes and compares the cost of the routes generated by the NDA, FRA and PRA for the dynamic stochastic network *G* in Figure 3. For each simulation experiment, the network in Figure 3 was generated 25 times using a Guassian random number generator. Note that the cost of the routes presented in Table 6 is equal to the traveling time on those routes in

minutes. Table 6 also compares the CPU time needed by each methodology to generate the best route.

To generate the best route using FRA, the initial traveling times and the mean historical traveling times on arcs are used. These parameters are both available offline. Table 6 column 11 shows the CPU-time needed to generate the best route for the FRA when μ is varied. Note that, the FRA is an offline algorithm. Therefore, once the best route was generated it was used for all 25 experimental trials, and, thus, column 12 is left blank.

The "actual" traveling time for a FRA generated route was, however, evaluated at every simulation trial. Table 6 columns 9 and 10 show the mean and standard deviation (STD) of the costs of the preplanned route evaluated using the data of the 25 generated simulation trials.

Table 6 columns 13 and 14 present the mean and STD of the costs of the routes generated by the PRA based on the results of the 25 trials. We assumed that the sampling time is 5 minutes; the maximum level of route planning is 5; and the maximum number of nodes visited on each partial route is 2. The last column in Table 6 shows the value of parameter β in (71) assumed in each simulation experiment.

Table 6: Comparing different routing algorithms in a dynamic stochastic network.

				NDA				FRA				PRA				
				cost		CPU time		cost		CPU time		cost		CPU time		
μ %	σ %	*q* %	*r* %	Mean	STD	Mean	STD	Mean	STD	Mean	STD	Mean	STD	Mean	STD	β
0	0.1	0.1	0.1	180	0.06	0.05	0.01	163	0.07	94.8	–	173	0.06	60.3	0.38	2
0	1	1	1	180.1	0.43	0.05	0.01	163.1	0.51	92.7	–	173.1	0.47	60.1	0.37	6
0	10	10	2	178.9	6.7	0.05	0.01	163.4	5.3	93.1	–	171.4	4.6	60.5	0.11	50
0	20	20	5	171	13	0.05	0.01	163.7	14.2	92.6	–	165.6	7.71	60.3	0.06	100
0	40	40	5	158.6	17.6	0.05	0.01	162.8	16	92.8	–	148.6	14.7	58.9	0.07	125
5	0.1	0.1	0.1	183.1	0.05	0.05	0.01	166.3	0.05	94.1	–	176.5	0.05	65.9	8.7	2
5	1	1	1	183.2	0.5	0.05	0.01	166.2	0.7	93.2	–	176.3	0.6	62.7	0.3	6
5	10	10	2	185.1	6	0.05	0.01	167.9	5.4	93.1	–	174.3	5.3	62.3	0.5	50
5	20	20	5	172.5	10.6	0.05	0.01	167.5	9.6	92.6	–	170.2	8.6	61.8	0.1	100
5	40	40	5	160.2	21	0.05	0.01	162	23.3	92.5	–	149.4	13.3	52.8	0.11	125
10	0.1	0.1	0.1	186.1	0.04	0.05	0.01	169.4	0.04	92.3	–	179.5	0.05	62.5	0.41	2
10	1	1	1	186.2	0.4	0.05	0.01	169.7	0.5	92.6	–	179.6	0.6	62.5	0.7	6
10	10	10	2	186.2	9.03	0.05	0.01	169.3	6.65	92.7	–	177	3.7	59.2	0.05	50
10	20	20	5	176.4	11.3	0.05	0.01	171.5	11.5	92.7	–	169	9.5	59.3	0.5	100
10	40	40	5	166.8	15.7	0.05	0.01	172.2	23.3	92.3	–	161.8	16.3	45.4	0.3	125
25	0.1	0.1	0.1	192.2	0.07	0.05	0.01	176.2	0.07	92.8	–	186.2	0.05	62.1	0.21	5
25	1	1	1	192	0.6	0.05	0.01	176.2	0.5	92.2	–	186.3	0.4	60.4	1.3	15
25	10	10	2	193.2	6.6	0.05	0.01	176.7	6.3	92.3	–	185.4	6.1	55.1	0.1	75
25	20	20	5	187.2	10.2	0.05	0.01	170.8	12.1	92.2	–	174.9	11.03	54.6	0.13	150
25	40	40	5	172.7	22.8	0.05	0.01	175.9	22.8	92.1	–	165.2	18.2	46.3	0.2	250
50	0.1	0.1	0.1	199.1	0.06	0.05	0.01	183.6	0.06	91.7	–	194.2	0.06	61.1	0.3	5
50	1	1	1	199.1	0.6	0.05	0.01	183.7	0.75	91.5	–	194.3	0.6	52.7	0.18	15
50	10	10	2	197.5	6.7	0.05	0.01	181.9	6.1	91.5	–	190.3	8.8	41.2	0.1	75
50	20	20	5	187.3	14.5	0.05	0.01	180.5	9.03	91.3	–	184.4	10.8	40.2	0.15	150
50	40	40	5	177.8	22.6	0.05	0.01	192.1	31.1	91.1	–	170.1	5.8	39.9	0.4	300

Table 6 shows that for dynamic networks when the uncertainty is relatively low, the FRA generates the best route, whereas when the uncertainty is high, the routes obtained by PRA outperform other routes in average. As seen from Table 6, for networks with high dynamism and uncertainties, the variation in the best route (STD) found by the PRA is relatively low compared to routes generated by the NDA and FDA. In other words, not only do the routes generated by the PRA cost less, in average, but also we have more confidence in taking these routes.

Results in Table 6 also indicate that for high dynamic networks, the quality of solutions generated by PRA is significantly sensitive to the parameters β. To obtain better solutions, we had to increase the parameter β for the last two sets of simulation experiments (i.e., μ=25% and μ=50%). Similar to Table 2, we can see that computationally the NDA obtains the solution in a very short amount of time, and that CPU time for the PRA to find the best route declines as the uncertainties and degree of dynamisms of the network increases.

5 Conclusions and Recommendations

The transportation industry, like many others, has undergone significant change in the last few years through the introduction of information technologies. Whereas in the past, it was difficult for a company to control or route vehicles once they left the terminal, these technologies make accurate dynamic real-time routing a very real possibility. Most of the existing routing strategies operate under one of the two modes: (1) pre-planned full routes when the level of uncertainty is low, and (2) local dispatching strategy when the level of uncertainty is high. There is lack of approaches for addressing the routing when the level of uncertainty is in between the two ends of the spectrum. To address this gap, we developed a partial routing strategy that preplans a portion of the demand and makes adjustments in real-time. Experimental results demonstrated the benefits of the partial routing strategy over the other two methods for a wide variety of scenarios.

Two factors that control the effectiveness of the partial routing strategy are (1) the maximum level of route planning which represents the number of stops that we need to take into consideration in the plan, and (2) the maximum number of nodes visited on each partial route which is the number of stops the vehicle visits before the route is updated. Future research can focus on developing models to optimize these two variables.

6 Implementation

Most of the hardware components required for the developed partial routing algorithms are already available in the market place. They include wireless communication devices used for dispatching vehicles, automatic vehicle locator for monitoring the vehicles, traveler information system to provide actual road conditions, a server to continuously run the algorithm, and mobile data terminals in the vehicles so that route data can be automatically sent via a data dispatcher to the vehicles. In terms of the software, the partial routing algorithm needs to be coded as a module in existing vehicle routing software. The algorithm will need to access the data collected from the vehicles regarding location and system status regarding actual road conditions.

References

Barton M. E., "24/7 Operation by Marine Terminals in Southern California: How to Make it Happen," CITT Industry Stakeholder Workshop One, Metrans Report, Nov. 2001.

Bander J.L., White C.C., "A heuristic search approach for a nonstationary stochastic shortest path problem with terminal cost," *Transportation Science*, vol. 36, no. 2, pp. 218-230, May 2002.

Bertsimas J., Van Ryzin G., "A stochastic and dynamic vehicle-routing problem in the Euclidean plane," *Operations Research*, vol. 39, no. 4, pp. 601-615, 1991.

Bertsimas J., Van Ryzin G., "Stochastic and dynamic vehicle-routing in the Euclidean plane with multiple capacitated vehicles," *Operations Research*, vol. 41, no. 1, pp. 60-76, 1993.

CalTrans (California Department of Transportation), "California Transportation Plan 2025", under preparation, 2004.

Fu L., "An adaptive routing algorithm for in-vehicle route guidance systems with real-time information," *Transportation Research Part B*, vol. 35, pp. 749-765, 2001.

Ichoua S, Gendreau M, Potvin JY, "Vehicle dispatching with time-dependent travel times," *European Journal Of Operational Research*, vol. 144, no. 2, pp. 379-396, 2003.

Hall R. P., "The fastest path through a network with random time-dependent travel times," *Transportation Science*, Vol. 20, No. 3, 1986.

Kim S., Lewis M.E., and C.C. White, "Optimal Vehicle Routing with real-time traffic information," submitted to *IEEE Transactions on Intelligent Transportation Systems*, 2003.

Laporte G., "The vehicle routing problem: an overview of exact and approximate algorithms," *European Journal of Operational Research*, vol. 59, pp. 345-358, 1992.

Mallon L.G. and Magaddino J.P., An Integrated Approach to Managing ,ocal Container Traffic Growth in the Long Beach –Los Angeles Port Complex, Phase II. Technical Report, Metrans Report 00-17, CA, 2001.

Mendel J. M., *Lessons in Estimation Theory for Signal Processing, Communications, and Control*, Prentice-Hall, Inc., USA, 1995.

Miller-Hooks E., and Mahmassani H., "Path comparison for a priori and time-adaptive decisions in stochastic time-varying networks," *European Journal of Operational Research*, vol. 146, pp. 67-82, 2003.

Ng L., Wessels, R.L., Do D., Mannering F., and Barfield W., "Statistical Analysis of Commercial Driver and Dispatcher Requirements for Advanced Traveller Information Systems." *Transportation Research Part C-Emerging Technologies*, vol. 3, no.6, 353-369, 1995.

Pattanamekar P, Park D, Rilett LR, Lee J, and Lee C, "Dynamic and stochastic shortest path in transportation networks with two components of travel time uncertainty," *Transportation Research Part C-Emerging Technologies*, vol. 11, no. 5, pp. 331-354, 2003.

Papoulis A., *Probability, random variable, and stochastic processes*, McGraw-Hill, New York, Third Edition, 1991.

Powell W.B., Jaillet P., and Odoni A., "Stochastic and dynamic networks and routing," in: Ball M.O., Magnati T.L., Monma C.L., Nemhauser G.L., (eds.) Network Routing, Handbooks in Operations Research and management Science, Volume 8, INFORMS, Elsevier Science, pp. 141-295, 1995.

Powell W.B., "A stochastic formulation of the dynamic assignment problem, with an application to truckload motor carriers," *Transportation Science*, vol. 30, no. 3, pp. 195-219, 1996.

Powell W.B., Towns M.T., and Marar A., "On the value of optimal myopic solutions for dynamic routing and scheduling problems in the presence of user noncompliance," *Transportation Science*, vol. 34, no. 1, 67-85 , 2000.