# An Information Based Routing Model for Hazardous Material Route Selection Problem

## Sumeet S. Desai[1] and Gino J. Lim[2]

[1]Laureate Education Inc, Baltimore, MD

[2]Department of Industrial Engineering, University of Houston

Corresponding author's Email: ginolim@uh.edu

**Abstract:** In this paper, we address some key research questions concerning the alternative routing policy of hazardous materials in real time using stochastic dynamic networks based on real life situations. The scenario that we address in this paper involves the use of sophisticated communication tools to provide information on the current condition of the optimal path and incorporate them in our optimization model to generate alternative routes for hazmat vehicles. We address the issues of designing a framework and requirements for an adaptive routing system. To overcome system instability and information overloading, a feeback based routing policy within the framework has been developed. We show the implementation of the framework and disucss the potential benefits of our approach with the help of numerical experiments based on a real hazmat transportation network.

*Keywords:* Hazardous Material, Emergency Management, Stochastic Dynamic Networks, Information Based Routing

## 1. Problem Background

### 1.1  Motivation

Suppose we have a truck carrying hazardous materials from an origin location to a destination and we want to determine the best routing policy for the truck with the intent of minimizing the risk of an accident. We assume that we have a model that determines an optimal routing policy in the form of the next best destination to visit when the truck arrives at a particular location at a certain time of the day (Desai & Lim, 2012). This routing policy is handed to the truck driver in the form of instructions and he follows them to reach the destination. This process of generating the optimal routes beforehand is often termed as "*a priori* route planning" where we assume that the information on the routes is up-to-date. A major disadvantage of the *a priori* route planning process is that it does not have the ability to regenerate the routes after they are planned.

However, it can happen that the conditions on the planned route suddenly become unfavorable to drive while the truck is en route. These changes in the optimal route could have occurred due to severe weather conditions, an accident, traffic congestion, or construction work. The conditions of the routes may have changed due to several reasons such as severe thunderstorm, hailstorm, snowstorm, explosion on route, heavy traffic congestion, on-going road maintenance, conventions, concerts or closure of a link due to construction. In such a situation, we should be able to provide the truck driver with the renewed set of instructions for reaching the destination. In this paper, we will address the problem of re-routing the trucks based on the most recent information available to us. We will also address the components required to establish such a framework and discuss their interaction to provide the re-routing instructions in an efficient manner.

The sudden changes in routes are referred to as Real-Time Events (RTE) (Beroggi & Wallace, 1995). Such events can be broadly classified into two categories: temporary delay and permanent delay. A temporary delay on a link may be defined as a closure of the link for a duration of a few hours and usually results from bad weather in a city, accident clearance or road maintenance. A permanent delay can be defined as a delay that results in closure of a link for more than a day and is usually a consequence of construction work or damaged link by some accident or natural calamity. It is quite clear that the occurrence of these events will affect the trip in several ways: a delay in shipment, increased chances of an accident (in case of hazmats) and poor performance. If a system that provides complete information of all these events at the time of route planning exists, this problem would have been re-solved to a large extent. However, the dynamic nature of the problem makes it very difficult to capture the occurrence of all such events on a network beforehand. In most routing problems, the driving directions generated from models is an open-loop system, i.e., routing policies do not take into account any real-time events. Therefore, we need a closed-loop system that incorporates these events into our optimization models and provides feedback as driving instructions to

drivers in real-time. We define the process of using real-time information to generate adaptive routing policies as *Information Based Routing*.
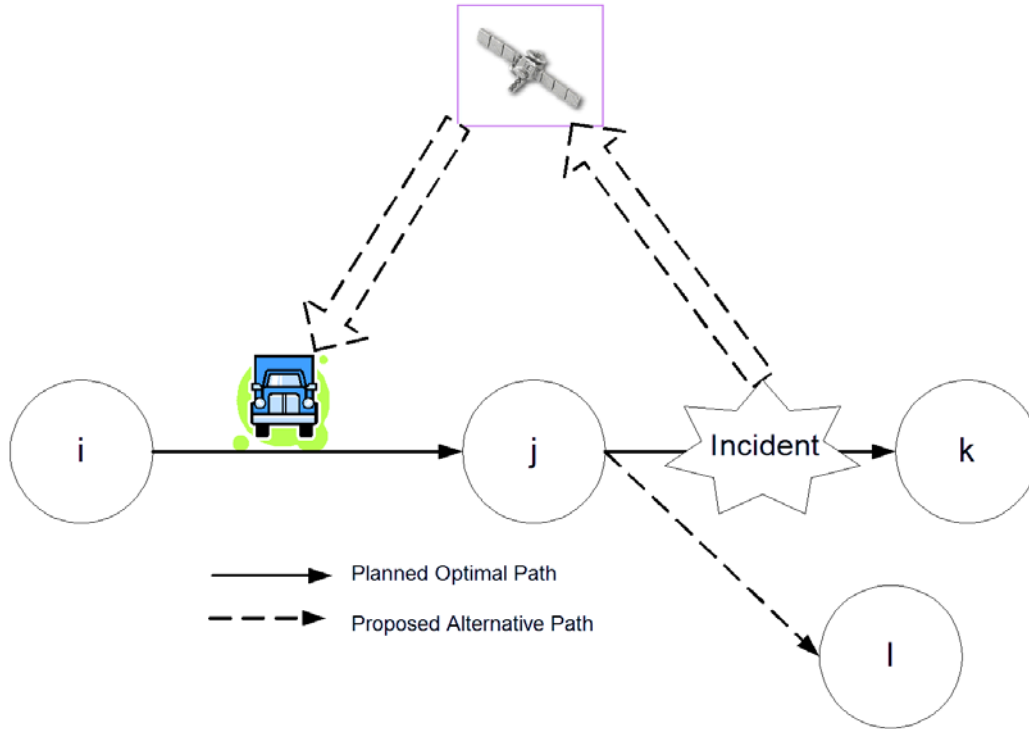


Figure   1: An Example of using sophisticated communication devices for hazmat routing

## 1.2   Description

We describe the real-time routing problem with the help of   Figure 1 . A truck carrying hazmats is traversing on an intermediate link $(i, j) \in P^*$ where $P^*$ is the optimal path determined *a priori*. The truck driver receives information via sophisticated communication devices that further link(s) $(j, k) \in P^*$ per se is permanently closed or may incur severe hours of travel delay. The truck is then forced to choose link $(j, l) \notin P^*$. Note that, we assume that a model from Desai & Lim, 2012 is able to determine $P^*$ and the delayed or closed links in the network will henceforth be referred to as *broken links*. Such a closure of any link or travel delays can be attributed to several real life situations discussed in earlier section.

Currently, the hazmat vehicles (or truck carriers generally) make use of Automatic Vehicle Locators (AVL) to monitor the progress of the trip. AVL has the ability to provide the location of a truck, but it does not have the ability to provide rerouting instructions (Beroggi & Wallace, 1995). In circumstances where links start operating in a few hours, the truck driver will have to wait or choose any other feasible path towards the destination. However, in case there is permanent damage to the link, any route towards the destination will be chosen. Any feasible path chosen to complete the journey will lead to the selection of a sub-optimal path which can be avoided by generating feedback based routes. Moreover, the AVL in this case would only help the carrier company track its progress on the sub-optimal path. Therefore, there exists a need for a feedback based routing system that is able to provide instructions on the new routes to approach destinations. In   Section 2 , we propose a framework that can be used as a test-bed to solve information based routing problems.

## 1.3   Literature Review

Over three decades, researchers have presented interesting perspectives on the problem of routing hazmat trucks with minimum risk. The transport risk associated with hazmats in literature can be categorized as: (1) expected population at risk, (2) incident probability, (3) emergency response capabilities of a route, and (4) the travel time. In many articles, the routing of hazmat vehicles was viewed as a problem of minimizing the risk with multiple objectives ( (Abkowitz & Cheng, 1988), (Glickman, 1991), (Jin & Batta, 1997), and (Kara, Erkut, & Verter, 2003)). The early years of this study maintained a steady focus on how to model the transport risk. Risk and cost are often used as the two main objectives in modeling the routing problem ( (Batta & Chiu, 1988), (Erkut & Verter, 1998), (Glickman, 1991), (Glickman & Rosenfield, 1984), (Jin, Batta, & Karwan, 1996), and (List & Mirchandani, 1991)). Another research work studied specific hazmat routing problems to show that the safest routes differ with the type of hazmat being transported (Ashtakala & Eno, 1996). Some of the investigations consider minimizing the expected risk of the first incident (Sivakumar, Batta, & Karwan, 1995) and suspension of shipments after a threshold number of accidents occur (Jin & Batta, 1997). Further, heuristic approaches were proposed for solving the route selection problem with a time dependent population nature ( (List, Mirchandani, Turnquist, & Zografos, 1991) and (Nozick, List, & Turnquist, 1997)). A discrete fractional programming model and a branch and bound approach was used to solve the hazmat routing problem (HRP) (Sherali, Brizendine, Glickman, & Subramanian, 1997). They focused on minimizing the risk of low probability-high consequence incidents given that an accident has occurred. Their investigation did not include the varying population at different times of the day. Under the non-order preserving criteria, the problem of transporting hazardous materials as a multiple objective (transport cost and risk to population) formulation (Nembhard & White, 1997). A dynamic programming approach was used to generate a solution. The heuristic methods were integrated with the geographical information systems (GIS) to solve the hazmat routing plan ( (Patel & Horowitz, 1994), (Weigkritch & Fedra, 1995), and (Zografos & Androustopoulos, 2004)). The hazmat vehicle routing problem was also considered under special criteria such as insurance costs, in presence of weather systems, and social equity concerns ( (Akgun, Parekh, Batta, & Rump, 2007), (Gopalan, Kolluri, Batta, & Karwan, 1990), and (Verter & Erkut, 1997)). A Markov Decision Process approach for solving the multiple objective hazmat route selection problem (Lim & Desai, 2010). In a more recent study, they compared the effectiveness of multi-variate statistical methods to predict the expected consequence of low probability-high consequence hazmat events (Desai & Lim, 2010).

In relation to the *a priori* path planning, the vehicle routing problem was solved considering a stochastic non-stationary network with terminal costs (Bander & White, 2002). They show that the performance of their algorithm AO* with lower bounds is significantly improved when compared with the dynamic programming approach. Various methods were proposed for reducing the state space of a non-stationary stochastic shortest path problems with real-time traffic information (Kim, Lewis, & White, 2005). The problem of routing hazmats in a stochastic dynamic environment was solved using stochastic dynamic prorgramming (Desai & Lim, 2012). This work used three conditioning techniques (a mixed use of pre-processing and bounds) to the stochastic dynamic programming to gain a significant computational advantage over stochastic dynamic programming.

Overall, the problem of routing hazardous materials has been studied extensively by taking into account various realistic perspectives. Most research articles we encountered have addressed the problem in the perspective of *a priori* path planning. In this paper, we offer a realistic view of information based routing the hazmat vehicles in stochastic dynamic networks and discuss the components and their interaction to establishing a framework that provides a re-routing policy to the drivers in the fastest possible manner. We also observe and address some key design issues that revolve around the computation time and reducing the burden on the framework.

## 2. Framework for Information Based Routing

Several components are required in order to design a routing system that adapts the optimal routes based on route conditions. A system that generates routes in real-time requires synergy of several sub-systems. Figure 2 shows the flow of information for such an information based routing framework. The very first step is to gather and store the current driving condition information on all the links in the current network on a remote server. In the next step, we have a mechanism to decide whether the current driving conditions will have a significant impact on travel time due to traffic, accident, local event, or construction. If the changes are not significantly affecting our current routing policy, we maintain our current optimal route. We trigger our next mechanism when the changes are significant. This mechanism determines whether the current changes affect our optimal routing policy for the driver. If the current changes affect the policy, the most recent information on expected delays on links are shared with the optimization engine. In the last step, we consider the recent information on links and the current position of the truck to trigger the optimizer to re-generate a dynamic optimal routing policy.
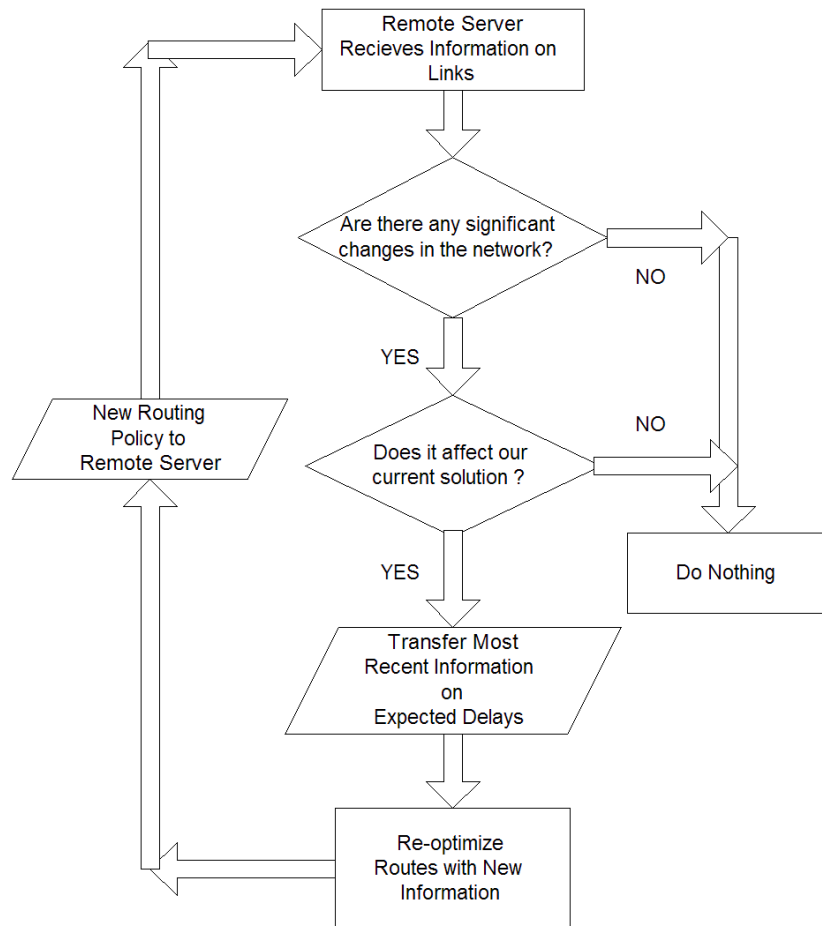
Figure 2: Flowchart showing logic of Information Based Routing

Based on our logical flowchart in Figure 2 , we propose three sub-systems to design a required framework. The proposed framework for real-time routing system is shown in Figure 3. The first sub-system is a **web server** that is able to capture real-time information on the route conditions throughout the road network under consideration. A salient characteristic of this web server is that it updates and maintains the most recent information on links in the network at a desired periodic interval. The web server has two major databases and include current conditions of the network and driving instructions for the truck. This web server is also referred to as a remote server. The second sub-system of the framework involves **filtering/webposting** information. This filtering/webposting system retrieves updated information on links from the web server via the internet.
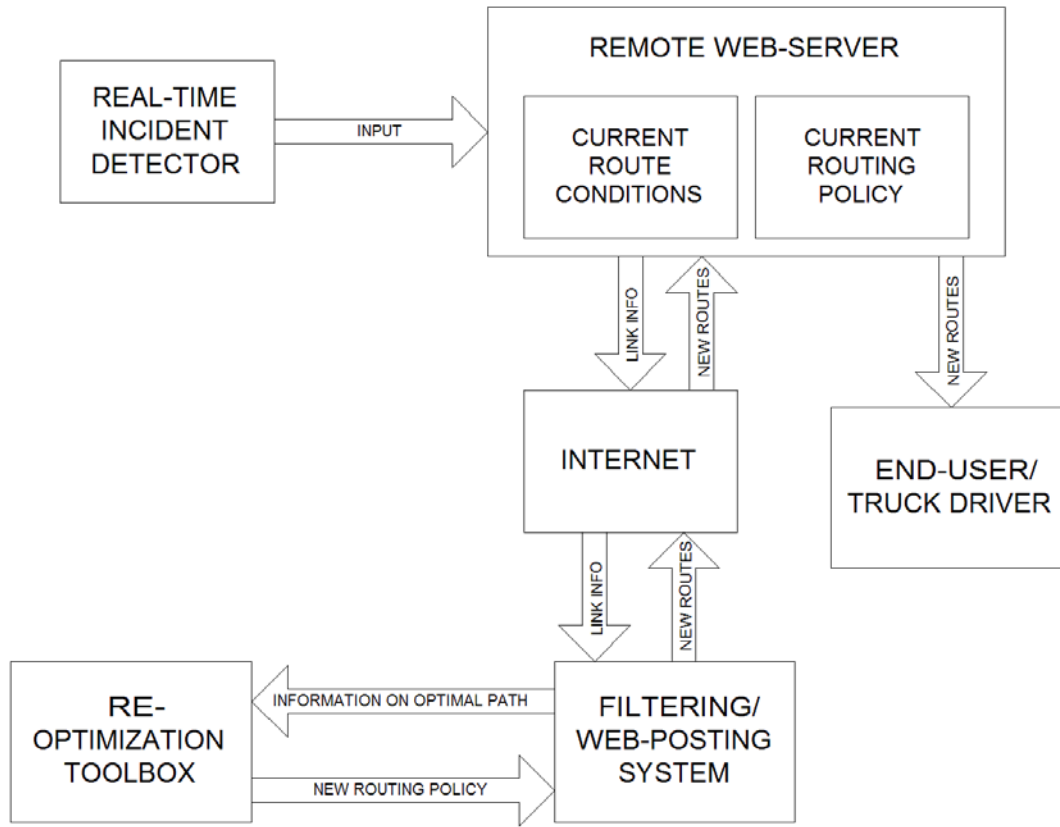
Figure   3: Framework for Information Based Routing using Web Server

The information update would be a list of arcs where driving conditions are unfavorable. The information would also include the expected delay on affected links. This filtering system determines whether the current changes in the links affect our planned optimal path. If there are any changes in the system, this filtering system would send updated information on expected delays on links to the third sub-system. However, if there are no changes on our planned path, the third component is not triggered. The third sub-system is the **re-optimization toolbox** usually triggered by the filtering component. The re-optimization toolbox will consider the most recent information on links and generate a new set of instructions for the driver. The new instructions are sent to the filtering/webposting component. The new instructions are sent to the web server where the driving instructions database is updated. A major advantage of a real-time web server is that it can be accessed by several interested parties at the same time.

## 2.1 Model Re-optimization Tool-box

The model we intend to use in the re-optimization toolbox is used by Desai & Lim (2012) for routing vehicles in a stochastic dynamic network. We also utilize their pre-processing technique to expedite the solution process for re-optimizing the vehicle routes. Consider a directed network $D = (N, A)$ where $N$ is the set of all nodes and $A$ is the set of all arcs in the network. Let $O$ be the origin location of hazardous materials and $\Gamma$ be the set of destination nodes such that $O, \Gamma \in N$ and $O \cap \Gamma = \varnothing$. We define the network as a set of successors, $j$ for each node $i$ such that there exists an arc $(i, j) \in A$. The directed network that we investigate in this problem has a time-dependent nature with several probabilities associated with it. We will refer to this network as a *stochastic dynamic network*. For our problem, travel time has a time dependent consequence denoted by $c(i, t_i, j, t_j)$. It is defined as the travel time required to travel from node $i$ at time $t_i$ to reach node $j$ at time $t_j$, where $t_i < t_j$. The time-dependent consequences are assumed to be non-negative. There are two types of

probabilities associated with the network. First, the probability of safe arrival, $p_s(j \mid i,j)$. It is defined as the probability of safely arriving at node $j$ given that the current location is node $i$ and the driver chooses node $j$ as the next destination. The second probability, $p(t_j \mid i,t_i,j)$, is defined as the probability of arrival at node $j$ at time $t_j$ given that the current time at which the truck departs node $i$ is $t_i$, $t_i < t_j$. The probability $p(t_j \mid i,t_i,j)$ equals 0 if $t_j \leq t_i$ or $t_j > t_i + T$ for a finite $T$. This means that the truck cannot reach the next destination backward in time. We define the expected consequence of traversing from node $i$ at time $t_i$ to node $j$ as $c(i,t_i,j)$. The expected cost of safely arriving at a node is obtained by

$$c(i,t_i,j) = p_s(j \mid i,j)\sum_{t_j} p(t_j \mid i,t_i,j)c(i,t_i,j,t_j).$$

We introduce a term $\pi(i,t_i)$ known as the routing policy for node $i$ at time $t_i$, $\forall\, i \in \mathsf{N}$, and $\forall\, t \in T$. A policy $\pi(i,t_i)$ is a means to instruct the driver about the next best node to be visited once the vehicle arrives at any location at a specified time. It is also termed as Deterministic Markov Policy. Every policy $\pi(i,t_i)$ is usually evaluated by its expected value denoted by $V^\pi(i,t_i)$. For a specified origin $O$ and a departure time $t_o$, the total expected cost of any policy $\pi$ is $v^\pi(O,t_o)$. It can be mathematically expressed as $V^\pi(O,t_0) = E^\pi_{i,t_i}\{\sum_{k=0}^{k-1} c(i_k,t_{i_k},\pi(i,t_i)) + \bar{c}(t_k)\}$, where $E^\pi_{i,t_i}$ is an expectation operator that depends on the use of policy $\pi$ given the initial state $(O,t_0) \in S_t$, $\bar{c}(t_k)$ could be the cost incurred at the end of the trip (depends on time $t_k$, refered as terminal cost (Bander & White, 2002)) and $k$ is the number of edges that construct a route from $O$ to $\Gamma$.

## 2.2 Elements of Stochastic Dynamming Programming Model

The problem makes a decision (to choose the next best node) at every time-expanded node $(i,t_i)$ $\forall i \in \mathsf{N}$ and $\forall t \in T$. This stage is often known as **decision epoch**. The overall state (search) space of the problem is denoted as $S_t$. The **state space** for the problem considered is $\{\mathsf{N} \times \{t_0,t_1,\ldots,T\}\}$ and $T$ is the time window. The **action space** for the problem is to find the next best node $j$ given a current state of the problem which is $(i,t_i)$. The **rewards** incurred for the actions we take in each decision epoch at a given state are $V(i,t_i) = \{c(i,t_i,j) + p_s(j \mid i,j)\sum_{t_j} p(t_j \mid i,t_i,j)V(j,t_j)\}$. **Transition probabilities** involved in the model are $p(t_j \mid i,t_i,j)$ and probability of the safe arrival of the vehicle, $p_s(j \mid i,j)$.

The objective is to find a set of actions that result in a routing policy, $\pi^*(i,t_i)$, such that $\pi^*(i,t_i) = argmin\{c(i,t_i,j) + p_s(j \mid i,j)\sum_{t_j} p(t_j \mid i,t_i,j)V^*(j,t_j)\}$, where $V^*(i,t_i)$ is the value of optimal policy $\pi^*(i,t_i)$ that is determined by Equation (1).

$$V^*(i,t_i) = min\{c(i,t_i,j) + p_s(j \mid i,j)\sum_{t_j} p(t_j \mid i,t_i,j)V^*(j,t_j)\}. \tag{1}$$

This equation is known as the *optimality equation* that is an important component for solving the problem using stochastic dynamic programming. Furthermore, we define $V^*(\Gamma,t) = 0$ as the boundary condition. In other words, the trip terminates at the destination $\Gamma$ at any time $t$ and the truck does not accrue any further consequences. Further, we claim that there exists a unique optimal routing policy for this stochastic dynamic programming formulation due to finite search space of the problem (Puterman, 1994). The optimal routing policy implies that the optimality equation is satisfied by every node $i$ at any given time $t_i$.

Proposition 2.1 shows the benefit of using this framework over an open-loop driving policy.

**Proposition 2.1** *In a stochastic dynamic network, the original values of the optimal policy for a re-rerouting point will always be less than or equal to the revised values of an optimal policy. We claim that* $V^*_{original}(i,t_i)\ +\ Expected\ \ Delay$ $\geq V^*_{revised}(i,t_i)$.

*Proof.* It is important in a real time situation for the framwork to be able to decide whether the truck needs to re-route or not. We can define the value of the optimal policy for every node $(i,t_i)$ as $V^*_{original}(i,t_i) \leq V^{\pi}(i,t_i)$ for any other routing policy $\pi$. In the worst case scenario, we can say that $V^*_{original}(i,t_i) = V^{\pi}(i,t_i)$. If we substitute this in our equation of claim we can easily say that $V^*_{revised}(i,t_i) \leq V^{\pi}(i,t_i)$. Also, in the worst case the value of expected delay will be 0. Hence, our claim is true for any case.

Although, this conceptual framework seems very robust we are more likely to encounter several real-time issues regarding its stability. We will discuss these concerns in detail in the next section and try to overcome them to some extent.

## 3. Stability Concerns

From the system flowchart it is easy to see that for the duration of the trip this system will continuously perform two tasks. Its first task is constant exchange information and interaction of several sub-systems. The second task is re-computing the new routes in case of significant changes on the optimal path. Due to the simultaneous occurrence of these tasks, the system will be overloaded with constant data flow between components, continuous re-computation of the routes and will eventually become unstable. This phenomenon is commonly observed in Enterprise Resource Planning (ERP) systems where customer or supplier orders are constantly updated in the system databases. For example, suppose the ERP system has a re-ordering point set such that it places orders to suppliers if its inventory falls below a point. If there are several orders that are received in a day such that this re-ordering point is hit several times, it will constantly re-order the material. This automatic updating of orders will generate several orders in a day which may lead to higher ordering costs and system instability. The strategy that is chosen to re-order the materials is the selection of a desired time frame to accumulate orders and re-order them at a fixed time period.

In our case, a heavy exchange of information will take place if it is decided that the database updates information every $x$ minutes. Therefore, we choose a similar strategy of accumulating desired information until the moment we start re-computing routing strategies. Let us assume that the weather condition in a particular city is very unfavorable and the inbound and outbound links are experiencing delays. The database will send/receive information every $x$ minutes. This means significant changes will be detected every $x$ minutes. These changes will trigger our re-optimization toolbox every $x$ minutes and generate a re-optimization policy. According to our current system, the newly generated policy will be sent to the driver every $x$ minutes. There are two issues that may arise. First, depending on the current location of the truck, network size and time interval chosen, the time required to re-optimize will vary. And due to the stochastic dynamic nature of the problem it may take longer than $x$ minutes to generate an optimal policy ( (Bander & White, 2002) and (Desai & Lim, 2012)). This will result in a lag and will put a heavy burden on computational resources of the optimization toolbox. Second, the driver will find this constantly changing information very confusing and irrelevant. The only instant a truck driver will be waiting for instructions is when he arrives at a node and he has to make a decision among available alternatives. Therefore, we need to accumulate the best information on links until a particular time and then execute re-optimization such that its instructions reach the driver only when he reaches a node or just before he arrives at the node.

### 3.1 Approach for Overcoming System Instability

We need a very effective procedure that not only helps overcome this drawback by reducing the burden on system resources but also provides instructions of the new policy with minimum possible wait at re-routing nodes. We demonstrate a step wise procedure that can be used to overcome this defect (See Figure 4 ).
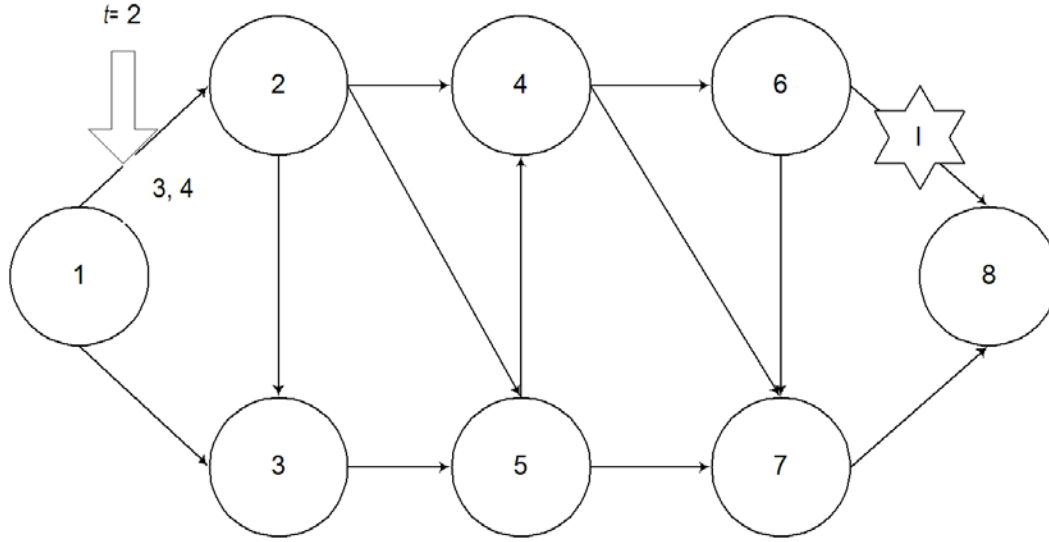
Figure 4: Illustration for Unstable Information based system

At the starting point, the driver will have an initial optimal routing policy about the next best nodes to visit. The time required to obtain this policy is recorded. Let us assume that it took 30 minutes to obtain the optimal policy and the database has the capability to update information every 15 minutes. The model shows that 1-2-4-6-8 is the optimal path to reach destination node 8. Suppose we start the truck at $t = 0$ hours from node 1. At time $t = 2$, we receive information that link (6,8) is having an expected delay of 5 hours. Based on available information, the earliest time the truck can reach node 2 is $t = 3$. According to our current framework, if the time taken to optimize the routes was 0.5 hours, we run re-optimization when we receive updates. In this case, the optimization will keep generating same policy every 30 minutes. The current system would encounter exchanging information and triggering optimization two times before the truck could reach the next node. Suppose the route generation takes 15 minutes, then the driver would receive the same information four times in a period of one hour. However, in order to avoid repetitive calculations, the best strategy to run a re-optimization can be at $t = 3.0 - 0.5 = 2.5$ hours so that the model will receive the most recent updated information and will be able to provide the best policy in real-time when it is most desired. Given that the truck is en route, re-optimizing the routes should take the optimization model run time into consideration. Therefore, a sub-routine is developed in our framework that considers optimization run model time to generate an update route before the truck reaches the next node in the travel path. The steps of this sub-routine for determining the appropriate time to run re-optimization are formalized below:

1. Run Optimization Model to generate initial policy.
2. Record POLICY and time required to optimize $t_{opt}$.
3. Run optimization model at time, $t_{run} = t_{earliest}(j) - t_{opt} - SS$ where $t_{earliest}(j)$ is the earliest time when the truck can reach node $j$ given the starting point was node $i$.
4. Update $t_{opt}$ and POLICY and send it to driver.
5. Verify whether there are any changes in links involving recent policy at time $t_j = t_j - t_{run}$ where $t_j$ is the time a truck reaches node $j$.
    a. If yes, re-run optimization model and ask driver to wait till $t_j + t_{opt}$ and,
    b. wait for the new POLICY and update $t_{opt}$.
    c. If no, send the best POLICY to driver.

In the above sub-routine, we are considering a worst case available time to re-optimize by using the earliest arrival time to reach

next node and $t_{reopt} \leq t_{opt}$. Thus, in most cases we will be able to provide real-time instructions with least possible wait. We will not only reduce major computation burden on system resources but also improve system stability and performance. Our current focus will be developing framework to address real time routing. The next section will provide details on experiment set-up for the problem and provide results on performance of the system.

## 4. Results

Our main objective of this paper is to provide a framework for information based routing and demonstrate its usefulness. The supplementary objective of our paper is to develop a prototype model that can be used as a test-bed to develop a real-time system. The script for accessing the data on a remote web server is developed using JAVA. The other components of the framework such as the filtering system is developed using MATLAB 7.0. This component verifies whether the changes in link conditions are affecting our planned path. We use the pre-processing approach for time bounded models explained in Desai & Lim, 2012 as a re-optimization toolbox. A MATLAB program provides the re-optimized routing policy in the form of a database file. The web-posting module is developed using an interface of JAVA and MATLAB. The web-posting module (JAVA code) retrieves new driving instructions from MATLAB and updates the data on a remote web server. This updated instructions are sent to the truck driver through sophisticated communication devices as Global Positioning System, GPS. In the next subsection, we provide details on the experiment set-up and our ideas on measuring the overall performance.

### 4.1 Experiment Setup

In order to demonstrate the effectiveness of our framework we define and include several parameters for our experiments. We use the 47-node state of Texas major roadways network as a test network for our framework. As mentioned earlier, a delay on a link are categorized broadly into two categories: (1) temporary and (2) permanent delays. In our experiments, we further classify temporary delays into two categories, mild and severe delays. A mild delay can be correlated to a scenario of an accident or severe thunderstorms on links that may last from few minutes to 12 hours. This parameter for experiment is obtained by generating a random number between zero and 12. A link may face severe delays due to construction and maintenance of highways. The expected delay on these links is generated by random number between 12 - 120 hours. For any permanent damages to links a very high penalty is imposed on the link. This penalty will lead the model to choose any other link among available alternatives. Moreover, to generate a scenario of an accident we need to supply few parameters to the model. We generate a random number on how many links are damaged at a time. We assume that 2 - 10% of links are damaged for each run. The next question is which links are damaged at the start of a run. A random number between 0 and $|A|$ ($|A|=$ number of arcs in the network) is generated and assigned to the arc number in the arc list. Further, if the links are damaged we assume that 70% of the times the expected delay is mild; whereas for about 25% of the time expected delay can be severe. Only 5% of the time the links can experience permanent damage. The details obtained from the random number generators are supplied to the filtering component which determines whether optimization toolbox is to be triggered or not.

### 4.2 Numerical Experiments

We run test experiments and for every run there are some induced damaged links to simulate the real-time scenario. We record the numerical experiments when there is a significant change in the optimal path and run optimization models to generate the new policies along with the time taken to re-optimize. We measure the effectiveness of this real-time system by presenting a comparison of expected delays (hour) in presence and absence of the real-time routing system. As a supplementary exercise we keep a track of time required to re-optimize (second) in case a significant change on an optimal path is detected by the filtering component. The setup chosen for running experiments is HML scenario as described in Desai & Lim (2012) with the day divided into 1 - hour intervals. We record the values of revised and original optimal values of $V^*(i,t_i)$ where $(i,t_i)$ is the closest re-rerouting point. Note that $V^*(i,t_i)$ in this case indicates the expected travel time (minute). We ran several experiments by keeping the re-routing points first closer to the origin location and gradually shift the re-routing points closer to the destination. The results are tabulated in Table 1.

Table 1: Effectiveness of Information Based Routing Problems

| No. of Links $\in P^*$ | $V_{original}^*(i,t_i)$ | $V_{revised}^*(i,t_i)$ | $t_{reopt}$ | Expected Delay |
|---|---|---|---|---|
| 2 | 739.34 | 1133.2 | 20.203 | 79 |
| 3 | 488.68 | 97.925 | 9.625 | 69 |
| 2 | 875.45 | 882.94 | 21.391 | 9 |
| 4 | 488.68 | 638.3 | 8.781 | 58 |
| 1 | 488.68 | 746.55 | 10.359 | 7 |
| 1 | 488.68 | 490.66 | 9.438 | 4 |
| 1 | 520.99 | 522.86 | 3.5 | 10 |
| 2 | 160.95 | 164.07 | 3.187 | 15 |
| 2 | 177.69 | 181.01 | 3.656 | 8 |
| 1 | 177.76 | 375.02 | 3.234 | 2 |
| 1 | 115.7 | 120.06 | 2.485 | 175 |
| 2 | 115.7 | 1088.7 | 2.343 | 12 |
| 1 | 498.83 | 509.59 | 1.891 | 12 |
| 2 | 545.03 | 823.76 | 1.609 | 18 |
| 1 | 65.2 | 125.94 | 1.172 | 9 |
| 2 | 355.33 | 756.09 | 1.672 | 4 |

The first inference from Table 1 is that the $V_{original}^*(i,t_i)$ is always less than $V^*(i,t_i)$. This result is consistent with Theorem 2.1 . It shows that the optimal values for the nodes $(i,t_i)$ under consideration have changed. The changes in optimal path have triggered the optimization toolbox to re-generate the values of new policy. The second inference from the results is that the $V_{original}^*(i,t_i) + ExpectedDelay \geq V_{revised}^*(i,t_i)$ . From this result, it re-confirms the fact that the presence of a closed loop routing policy is always beneficial over an open-loop routing policy. A third observation is regarding the time to re-optimize. As the truck moves closer to the destination or farther away from the source node, the time required to re-optimize reduces gradually.

## 5. Discussion

In this paper we identifiedDealing with a real time routing problemconcern for hazmats, . Wwe discussed at length the difference between a open-loop policy and feedback based system along with its advantages and disadvantages. We proposed aA three component framework was proposed to address the problem of real-time routing. We discussed the functionality of each component and its interaction with other sub-systems to provide the result. We identified a real time concern for the system stability and information overloading. To overcome this issue, we incorporate a sub-routine was developed and incorporated into the framework that supports system stability. In our experiments, we show how the presence of a feedback based routing is beneficial over an open-policy. These systems will benefit a hazmat carrier and expedite the delivery of products, will improve on-time delivery performance. As a result, , will reduce the driver fatigue can be reduced, which wil , and will result in improved safety for hazmats.

## 6. Acknowledgements

computational experiments.

## 7. References

Abkowitz, M., & Cheng, P. (1988). Developing Risk-Cost Framework for Routing of Truck Movements of Hazardous Materials. *Accident Analysis and Prevention, 20*(1), 29-51.

Akgun, V., Parekh, A., Batta, R., & Rump, C. (2007). Routing of a Hazmat Truck in Presence of Weather Systems. *Computers and Operations Research, 34*, 1351-1373.

Ashtakala, B., & Eno, L. (1996). Minimum Risk Route Models for Hazardous Materials. *Journal of Transportation Engineering, 122*(3), 350-357.

Bander, J., & White, C. (2002). A Heuristic Approach for Solving Non-stationary Stochastic Shortest Path Problem with Terminal Cost. *Transportation Science, 36*(2), 218-230.

Batta, R., & Chiu, S. (1988). Optimal Obnoxious Paths on a Network: Transportation of Hazardous Materials. *Operations Research, 36*(1), 84-92.

Beroggi, G., & Wallace, W. (1995). Operational Control of the Transportation of Hazardous Materials: An Assessment of Alternative Decision Models. *Management Science, 41*(12), 1962-1977.

Desai, S., & Lim, G. (2010). A Comparison of Multivariate Statistical Methods for Estimating the Expected Consequences for Low Probability High Consequence Events. *Human Factors and Ergonomics in Manufacturing & Service Industries, 20*(3), 233-250.

Desai, S., & Lim, G. (2012). *Solution Rime Reduction Techniques of a Stochastic Dynamic Programming Approach for Hazardous Materials Route Selection Problem.* University of Houston. Retrieved February 5, 2013, from E2Map for Natural Disasters: University of Houston: http://e2map.egr.uh.edu

Erkut, E., & Verter, V. (1998). Modeling of Transport Risk for Hazardous Materials. *Operations Research, 46*(5), 625-642.

Glickman, T. (1991). An Expeditious Risk Assessment of the Highway Transportation of Flammable Liquids. *Transportation Science, 25*(2), 115-123.

Glickman, T., & Rosenfield, D. (1984). Risk of Catastrophic Derailments Involving Hazardous Materials. *Management Science, 30*(4), 503-511.

Gopalan, R., Kolluri, K., Batta, R., & Karwan, M. (1990). Modeling Equity of Risk in the Transportation of Hazardous Materials. *Operations Research, 38*(6), 961-973.

Jin, H., & Batta, R. (1997). Objectives Derived from Viewing Hazmat Shipments as a Sequence of Independent Bernoulli Trials. *Transporation Science, 31*(3), 252-261.

Jin, H., Batta, R., & Karwan, M. (1996). On the Analysis of Two New Models for Transporting Hazardous Materials. *Operations Research, 44*(5), 710-723.

Kara, B., Erkut, E., & Verter, V. (2003). Accurate Calculation of Hazardous Materials Transport Risks. *Operations Research Letters, 31*, 285-292.

Kim, S., Lewis, M., & White, C. (2005). State Space Reduction for Non-stationary Stochastic Shortest Path Problems with Real-time Traffic Information. *IEEE Transactions of Intelligent Transportation Systems, 6*(3), 273-284.

Lim, G., & Desai, S. (2010). Markov Decision Process Approach for Multiple Objective Hazardous Materials Transportation Route Selection Problem. *International Journal of Operational Research, 7*(4), 506-529.

List, G., & Mirchandani, P. (1991). An Integrated Network/Planar Multiobjective Model for Routing and Facility Location of Hazardous Materials and Wastes. *Transportation Science, 25*(2), 146-156.

List, G., Mirchandani, P., Turnquist, M., & Zografos, K. (1991). Modeling and Analysis for Hazardous Material Transportation: Risk Analysis, Routing/Scheduling, and Facility Location. *Transportation Science, 25*(2), 100-114.

Nembhard, D., & White, C. (1997). Application of Non-order Preserving Path Selection to Hazmat Routing. *Transportation Science, 31*(3), 262-271.

Nozick, L., List, G., & Turnquist, M. (1997). Integrated Routing and Scheduling in Hazardous Materials

Transportation. *Transportation Science, 31*(3), 200-215.

Patel, M., & Horowitz, A. (1994). Optimal Routing of Hazardous Material Considering Risk of Spill. *Transportation Research Part A, 28A*(2), 119-132.

Puterman, M. (1994). *Markov Decision Processes: Discrete Stochastic Dynamic Programming.* New York, NY: John Wiley Sons.

Sherali, H., Brizendine, L., Glickman, T., & Subramanian, S. (1997). Low Probability High Consequence Considerations for Routing Hazardous Material Shipments. *Transportation Science, 31*(3), 237-251.

Sivakumar, R., Batta, R., & Karwan, M. (1995). A Multiple Route Conditional Risk Model for Transporting Hazardous Materials. *INFOR, 33*(1), 20-33.

Verter, V., & Erkut, E. (1997). Incorporating Insurance Costs in Hazardous Materials Routing Models. *Transportation Science, 31*(3), 227-236.

Weigkritch, K., & Fedra, E. (1995). Decision Support Systems for Dangerous Goods Transportation. *INFOR, 33*(2), 84-99.

Zografos, K., & Androustopoulos, K. (2004). A Heuristic Algorithm for Solving Hazardous Materials Distribution Problems. *European Journal of Operational Research, 152*, 507-519.