Dispatching/Routing of Emergency Vehicles in a Disaster Environment using Data Fusion Concepts

Qiang Gong
University at Buffalo (SUNY)
CMIF†
438 Bell Hall
Buffalo, NY 14260
U.S.A.
qgong@eng.buffalo.edu

Arun Jotshi
University at Buffalo (SUNY)
CMIF
438 Bell Hall
Buffalo, NY 14260
U.S.A.
aajotshi@eng.buffalo.edu

Rajan Batta*
University at Buffalo (SUNY)
CMIF
438 Bell Hall
Buffalo, NY 14260
U.S.A.
batta@eng.buffalo.edu

Abstract – *The goal of our work is to develop a scalable methodology to a variety of applications requiring data fusion support for decision-making, which has direct applicability to a disaster or homeland security situation. In this work we have considered a dynamic disaster environment where an earthquake has struck and thousands of casualties are to be attended to. The fusion center outputs both L1 fusion and L2 fusion constructs. Using this information ambulances need to be dispatched to casualty pick-up locations and then deliver patients to an appropriate hospital. The key factors that affect the dispatching of ambulances to patient locations include patient priority, cluster information, and distance. Similarly, those affecting the dispatching of ambulances from patient locations to hospitals include waiting time estimates at hospital emergency rooms, hospital capacities, injury type, and distance. Routes also need to be generated for these ambulances. Route generation needs to take into account road damage information and have back-up routes ready whenever needed.*

Keywords: Data fusion, dispatching, routing, disaster environment.

1 Introduction

When there is an emergency situation, data begin to flow from the site to police, and other emergency personnel. When the situation is much more grave and threatening, and is on a large scale, the data flow becomes much more complex. The science of efficiently organizing and interpreting massive amounts of such data is called "Data Fusion". It appears in the scientific literature in the late 1960s [1] and later it has been implemented in multiple disciplines in the 1970s and 1980s [2]. A typical situation is one in which information flowing from multiple sources has a highly variable character (e.g., human intelligence, signal intelligence, etc.). It is necessary to align the data and develop a comprehensive picture rapidly and accurately in order to take full advantage of it in our emergency-relief services.

We consider a dynamic disaster environment (earthquake) in which we have, in addition to the first impact effects, aftershocks that cause additional effects. Management of rescue resources in such an environment requires an efficient dispatch and routing method that provides rapid response for casualty pick up and delivery. The information of casualties and the road status is reported by sensors. Information on each patient is composed of his/her location and injury class. Casualties are usually classified into three priority categories, with priority j (j = 1, 2, 3) corresponding to severe, moderate and mild injuries, respectively. Information on each link (road) is composed of its level of damage and the probability associated with it. The information comes in through satellite images, sensor systems embedded in the infrastructure, police reports, property owners and other individuals. Our research is to develop a dispatching and routing method that utilizes this fused data to minimize the response time and increase efficiency.

* Author for correspondence.

† Center for Multisource Information Fusion.

2 Literature Review

Dispatch strategies in emergency response systems have been investigated for decades. Use of queueing theory, specifically priority queueing methods, to police patrol systems was studied by Schaack and Larson [3]. They formulated an N-server, T-priority problem, with Poisson arrivals and negative exponential service times. They calculated the steady state probabilities based on a probabilistic approach. Building up this work, they derived waiting time distributions as well as expressions for the expected waiting times for all priority customers. The results of the N-server, T-priority problem were applied to a target dispatching problem in a military setting by Mishra, Batta and Szczerba [4]. They also developed two other methods−target initiated dispatch strategy and aircraft initiated dispatch strategy. Whenever a target occurs and at least one aircraft is available, instead of always dispatching the nearest aircraft, they considered sending the one that minimizes expected response time to the current target and a potential future target. On the other hand, if an aircraft becomes available while some targets are waiting to be responded to, a shortest travel time rule rather than the first come first served method is used to determine which target is to be served first.

Due to the similarity between aircraft dispatching and ambulance dispatching, both dispatch strategies are applicable to our problem. Target initiated method is extended in our problem by considering the existence of clusters of patients, and aircraft initiated method is broadened by considering a small time windows.

Another piece of work related to this topic was done by Choi, Rump and Rogova [5]. They addressed the single machine stochastic scheduling problems in which both the processing times and due dates of jobs are random variables with arbitrary distributions. Their scheduling problems can be transferred to our problem if we assign each patient a due date which means a patient may die if he/she doesn't get serviced before that time. Their objective of minimizing the expected number of tardy jobs is equivalent to the objective of minimizing the expected number of deaths in our problem.

Much recent research has been concentrated on routing of vehicles in a dynamic environment. Gendreau and Potvin [6] gave an overview of the dynamic vehicle routing (DVR) problems. DRV refers to a wide range of problems where information of the problem is revealed to the decision maker concurrently with the determination of the solution. In these problems a set of vehicles is routed over a particular time horizon while new services are occurring in real-time. But none of this has been concerned directly with a dynamic disaster environment. Not much has been done in the area of routing in emergency situations given fused data. The core requirement for the development of an automated response system in such an environment is to find routes for a given origin/destination pair and also have some backup routes in case some of them become unusable. The concept of having backup routes and hence dissimilar paths for an O-D pair has been extensively used in the routing of hazardous materials. It has been used in a military context as well. Akgun, Erkut and Batta [7] in their paper have described various method of finding dissimilar paths. They first describe three existing methods of finding spatially dissimilar paths and then an alternative method in which they first generate a large set of paths and then apply the p-dispersion heuristic to generate a small subset of paths. The three methods which they described were Iterative Penalty Method (Johnson, Joy, Clarke and Jacobi [8]), Gateway Shortest Path problem (Lombard and Church [9]) and MinMax method (Kuby, Zhongyi and Xiaodong [10]). Tyagarajan, Batta, Karwan and Szczerba [11] used the method of [6] for the routing of aircraft in order to minimize the chance of detection during mission ingress. In our problem, since excess ambulances travelling on similar paths may cause congestion and thus delay the response to the patients, it is necessary to take account of dissimilarity in our problem. All the above methods are modified and used along with data fusion.

3 Dispatch and Routing Strategies with Fusion Information

The basic idea is to find reliable and efficient methodologies to pick-up and deliver casualties to the hospitals in a dynamic disaster environment. The task of dispatch and routing is broken into two interrelated decision systems. The Dispatcher makes the decisions about which ambulance to send to which casualties and then to which hospital. The Router finds a path for each ambulance to complete its service, always having backup routes in case of a blockage on roads.

3.1 Data Fusion

Original data are generated in a network that has been given in the form of Tele-Atlas database file. This has point features, line features and area features. Each of the links is represented by a 256-character line containing information about the link. Out of this, we first derive the information we want for our network and then construct our network for the region (for more information on Tele-Atlas go to: http://www.teleatlas.com). We are using the Mapbase file for Los Angeles and we are trying to generate the whole network of roads and bridges from the Mapbase data file.

After original data have been collected, the fusion center fuses them and reports the fused data to the dispatcher and router. We currently have considered level 1 (L1) fusion and level 2 (L2) fusion.

3.1.1 L1 Fusion

Due to imprecision of the original information, the fusion center cannot determine precisely which injury class a patient belongs to. Instead, it outputs a distribution as follows:

$$\begin{cases} p_{i1}, & \text{if patient } i \in \text{injury class 1}, \\ p_{i2}, & \text{if patient } i \in \text{injury class 2}, \\ p_{i3}, & \text{if patient } i \in \text{injury class 3}. \end{cases}$$

Here we consider three injury classes. Based on this probabilistic information, we calculate the priority value for each patient. The larger the value is, the higher the dispatching priority will be. If we assign a weight w_j to class j, the average priority value for each patient is given by $\sum_{j=1}^{3} w_j p_{ij}$.

It is also difficult to provide exact spatial coordinates of each patient. The fusion center estimates a confidence region in which the patient is located with a probability no smaller than p. Such region will be used later in L2 fusion.

For road condition, again the fused data give us, associated with each link, a percentage of damage with some probability as shown in Table 1. In terms of the average level of damage that is given by $\sum_i x_i D_i$ we have certain speeds with which we can travel through/on them. The levels of damage we are considering here are shown in Table 2. The expected value will correspond to some level of damage according to the range it is in. Since that level of damage has a speed related to it, we can calculate the corresponding increase in the time to traverse the link, by comparing it with the speed of travel on the link had it been not damaged. Thus we can get the expected travel time from an

Table 1: Fused Data of Road Condition

Percentage of Damage	Probability of information accuracy
D_1	x_1
D_2	x_2
.	.
.	.
.	.

origin to a destination. So now we update the traversal time for each affected link with its expected traversal time and the network is modified. After the network is updated, we will have to rerun the code and generate a new set of paths.

Table 2: Levels of Damage

Level of Damage	Range of Percentage of Damage	Speed of Travel(mph)
L_0	0%	40
L_1	5% - 15%	35
L_2	15% - 25%	30
L_3	25% - 35%	25
L_4	35% - 45%	20
L_5	45% - 55%	15
L_6	55% - 65%	10
L_7	65% - 85%	< 5
L_8	85% - 100%	≈ 0

3.1.2 L2 Fusion

Based on the location data available from L1 fusion, L2 fusion considers the vicinity of each patient and deduces if there is a cluster in a small area. Generally speaking more concern should be focused on clusters in an efficient dispatch policy. In consideration of that, we introduce a parameter α. If a patient belongs to a cluster, we associate α to him/her. Suppose a patient belongs to a cluster k, the revised priority value (RPV) of him/her is calculated as $\sum_{i \in cluster\ k} \alpha \sum_{j=1}^{3} w_j p_{ij}$. An example is shown in Table 3. As to road condition, when we have the link status of various links, we can check if some links with damage are from the same vicinity. If such a case does exist, we can infer that some kind of mishap might have occurred there, e.g., a collapsed building. We can use this information and take that region out of the network database we are working with, thus helping in accurate and fast methods of calculating route timings and distances. In addition to the damages to various links we might have congestion on links, thus making travel through them take a longer time. By congestion we mean that the road/path is not passable at normal speed. For instance, the road/path might be congested due to any of the following reasons:

1. It is the only way to go to a particular destination. All other roads might be damaged and not passable.

2. The traffic density is very high on the particular path/road.

3. Traffic from other regions is being diverted to this path/road.

4. Congestion due to road damage (with usual traffic).

The effect of congestion is reduced speed and hence more time of travel. This effect can be taken care of by increasing the link traversal durations in the network data base by a factor corresponding to the reduction in speed due to the congestion.

3.2 Dispatch Strategies

The critical issue of dispatch is to determine which available ambulance should be sent to respond to which cluster of patients and which hospitals those patients should be delivered to. Therefore this task is divided into two subtasks. The first is the patient pickup problem, and the other is the patient delivery problem. In the first subproblem we consider two methods--patient initiated dispatch and ambulance initiated dispatch.

Table 3: Patient Priority Values

Patient NO.	Fused Data	Original Patient Value	Belongs to a cluster?	Cluster NO.	Revised Patient Value
1	$p_{11} \mid p_{12} \mid p_{13}$	$\sum_{j=1}^{3} w_j p_{1j}$	Yes	2	$\sum_{cluster\ 2} \alpha \sum_{j=1}^{3} w_j p_{1j}$
2	$p_{21} \mid p_{22} \mid p_{23}$	$\sum_{j=1}^{3} w_j p_{2j}$	Yes	1	$\sum_{cluster\ 1} \alpha \sum_{j=1}^{3} w_j p_{2j}$
3	$p_{31} \mid p_{32} \mid p_{33}$	$\sum_{j=1}^{3} w_j p_{3j}$	No	-	$\sum_{j=1}^{3} w_j p_{3j}$
4	$p_{41} \mid p_{42} \mid p_{43}$	$\sum_{j=1}^{3} w_j p_{4j}$	Yes	1	$\sum_{cluster\ 1} \alpha \sum_{j=1}^{3} w_j p_{4j}$
5	$p_{51} \mid p_{52} \mid p_{53}$	$\sum_{j=1}^{3} w_j p_{5j}$	No	-	$\sum_{j=1}^{3} w_j p_{5j}$
6	$p_{61} \mid p_{62} \mid p_{63}$	$\sum_{j=1}^{3} w_j p_{6j}$	Yes	2	$\sum_{cluster\ 2} \alpha \sum_{j=1}^{3} w_j p_{6j}$
7	$p_{71} \mid p_{72} \mid p_{73}$	$\sum_{j=1}^{3} w_j p_{7j}$	Yes	1	$\sum_{cluster\ 1} \alpha \sum_{j=1}^{3} w_j p_{7j}$

3.2.1 Patient Pickup: Patient Initiated Dispatch

It was noted by Mishra, Batta and Szczerba [4] that always sending the first preferred (closest) aircraft does not necessarily guarantee the minimum overall expected response time (measured over a set of targets). This rule also holds true in our problem. When a cluster of patients is reported at time t, rather than always dispatching the nearest ambulance we investigate a small time window $[t, t+\delta t]$ in which a potential cluster of patients may occur.

A set of roads divides the entire area into units of cells. Denote λ_{kr} to be the arrival rate of a cluster with k patients within cell r, which is related to the population density in that cell. Let λ be the arrival rate over that area. Therefore if the probability of no patients occurring in the time interval $[t,\ t+\delta t]$ is given by $e^{-\lambda\cdot\delta t}$, the probability of a cluster with k patients occurring in cell r can be calculated as $(1 - e^{-\lambda\cdot\delta t})\dfrac{\lambda_{kr}}{\lambda}$.

We consider the measure of response time, which is the waiting time of a patient until he/she is picked up by an ambulance (including the on-scene service time). In our problem we need to consider clusters rather than individuals. Thus we incorporate the revised patient priority value to calculate a response time for each cluster. Then our problem is given by

$$\min\ \overline{RT}_a = e^{-\lambda\cdot\delta t} * RT_a * RPV + \sum_r \sum_k [(1 - e^{-\lambda\cdot\delta t})\frac{\lambda_{kr}}{\lambda} * (RT_a * RPV + RT_{next} * RPV')] \quad (1)$$

where

$\overline{RT}_a$ = clustered response time to current cluster and the next potential cluster given ambulance a is dispatched;

RT_a = response time to current cluster given ambulance a is dispatched;

RT_{next} = response time to next cluster given the nearest ambulance is dispatched;

RPV = revised priority value for current cluster;

RPV' = revised priority value for next cluster;

The ambulance a with minimum $\overline{RT}_a$ is dispatched.

3.2.2 Patient Pickup: Ambulance Initiated Dispatch

If all ambulances are busy, the patients have to join a spatial queue and wait to be serviced. When an ambulance becomes available at time t, instead of dispatching it in a first come first served or a shortest travel time manner, we again consider a small time window $[t,\ t+\delta t]$ in which some potential ambulances may become available.

Unlike the occurrence of patients which is difficult to predict, the time that a busy ambulance becomes available is relatively easy to be estimated. After an ambulance has picked up patients, through routing information the fusion center can also calculate the fused data about the time needed to complete this service. It outputs such a distribution--it takes s units of time for ambulance i to finish its service with a probability p_{is}. Then we can calculate the expected service time of each ambulance for each service.

Armed with the information of the finish time point of every ambulance, the fusion center calculates which ambulances become available in that time interval $[t,\ t+\delta t]$. By considering that, we dispatch the ambulances in order to minimize the total expected travel time (including on-scene service time). The current ambulance will be dispatched to the patients under such consideration. Thus the problem is given as follows:

$$\min\ TTT_b = TT_b * RPV_b + \sum_{i\in S} TT_i * RPV_i \quad (2)$$

where

S = set of potential ambulances that may become available in $[t,\ t+\delta t]$;

TTT_b = total expected travel time to the patients that are serviced by the current ambulance and the potential ambulances in $[t, t+\delta t]$;

TT_b = the expected travel time to the cluster of patients that are serviced by the current ambulance;

RPV_b = revised priority value for the cluster of patients that are service by the current ambulance;

TT_i = the expected travel time to the cluster of patients that are serviced by ambulance i, $i \in S$;

RPV_i = revised priority value for the cluster of patients that are serviced by ambulance i, $i \in S$.

The cluster of patients b with the minimum TTT_b is responded by the current ambulance.

3.2.3 Patient Delivery: Minimum Expected Travel Time and Waiting Time at Hospitals

Our objective is to have the patients treated as early as possible. Even if an ambulance can deliver patients to a hospital in a short time, it is undesirable for the patients to wait at that hospital for a long time. Therefore, instead of only considering the travel time to hospitals, we incorporate the expected waiting time at hospitals--which is obtained from the hospital module. The delivery rule is defined as follows:

$$\min\ TT_c + WT_c \quad (3)$$

TT_c = expected travel time to hospital c;

WT_c = expected waiting time in hospital c.

The patients are delivered to hospital c with the minimum $TT_c + WT_c$.

3.3 Routing Strategies

Some of the issues which have to be addressed in our routing methodology include methods to generate a path for each of the origin and destination pairs for the ambulance, effects of any ambulance route on the decision of the route for another ambulance and the link status. The link status of the whole network is given. Associated with each link is a level of damage and with each level of damage is associated some probability of that damage. That means for each link we are given, with some probability, the type and the level of damage we have for the links of the network. So in our methodology we will have to consider this for the calculation of the various (dissimilar) shortest paths.

For robust path generation with backup routes we need a set of paths as opposed to one single path. A simple routing procedure cannot be applied here, since a single best path is adequate for static route planning where there is no uncertainty and the link status does not change. Dynamic problems with uncertain links and uncertain patients and ambulance locations, which are based on fused data estimates, need a set of paths so as to have backup routes. We need to generate various sets of paths for the same origin and destination pair. This is done in order to be sure that if along a path we have some link failure, we should be able to have other possible routes that share the minimum number of links with other routes for the same O-D pair. The idea is to generate paths sharing a minimum number of links and having durations that are within a set percentage of optimal. Also we might want to have temporal dissimilarity between any sets of ambulances serving the casualties. This will ensure that not all the ambulances end up in the same locality and there is no one to serve another region, in case of a subsequent disaster event. Also we do not want to have the majority of the ambulances in any one region and get stuck there. This concept is also used in a military setting to avoid enemy attack [11] and in the routing of hazardous materials to spread the risk of exposure to the same set of population [12]. So in the problem the initial items are given as follows:

- n origin and destination (O-D) pairs,
- m ambulances,
- a network over which the ambulances have to be routed.

Our goal is to find paths that are spatially and temporally dissimilar.

3.3.1 *Problem Formulation*

The problem is formulated as a Quadratic Semi-Assignment Problem (QSAP)--see Domschke, Forst and Voss [13] for a general discussion on the QSAP. Here we only force each ambulance to select a path, but not each path to be selected by an ambulance; hence not a full assignment problem.

$Notation$:

- m = number of ambulances;
- I = set of O-D pairs labelled $i = 1, \ldots, |I|$, where $|I| = n$;
- J_i = set of the possible paths incorporating dissimilarity with respect to space and time for the ith O-D pair, labelled $j = 1, \ldots, |J_i|$;
- R_{ij} = jth possible path for the ith O-D pair, $j = 1, \ldots, |J_i|$.

$Variables$:
Define a set of (0-1) variables

$$X_{ij} = \begin{cases} 1, & R_{ij} \text{ is selected,} \\ 0, & \text{otherwise.} \end{cases}$$

Therefore the problem can be formulated as:

$$(P) \max \sum_{i_1 \in I} \sum_{j_1 \in J_{i_1}} \sum_{i_2 \in I} \sum_{j_2 \in J_{i_2}} X_{i_1 j_1} X_{i_2 j_2} W_{i_1 j_1 i_2 j_2} \quad (4)$$

s.t.

$$\sum_{j \in J_i} X_{ij} = 1, \qquad \forall i \in I, \quad (5)$$

$$X_{ij} \in \{0, 1\}, \qquad \forall i \in I,\ j \in J_i. \quad (6)$$

Where
$W_{i_1 j_1 i_2 j_2}$=robustness index between paths $R_{i_1 j_1}$ and $R_{i_2 j_2}$.

Here the objective function tries to maximize the total robustness of the paths by choosing two such paths that have the maximum dissimilarity between them. Constraint (5) ensures that only one path is chosen for each one of the n O-D pairs. Constraint (6) is a binary variable which either chooses the jth path for the ith O-D pair or not. Final output of solution would be one route for each O-D pair that is maximally dissimilar from all other routes of the other O-D pairs and hence robust.

3.3.2 *Solution Methodology*

The selection process of a path involves generating a candidate path set with large number of paths for each ambulance (say 50-60) and then out of this candidate set we generate a small set of paths for each ambulance (say 5-10). Then we choose for each ambulance, from the generated set, a single path that satisfies our criteria.

3.3.2.1 Generating a Candidate Set of Paths

In the formulation we have assumed that we are given a set J_i, which is a set of possible paths incorporating dissimilarity with respect to space and time for the ith O-D pair. In the present scenario, we are looking for such candidate routes which are both spatially and temporally dissimilar. This can be posed a p-dispersion problem. The classical form of p-dispersion problem is to select p out of n given candidate points, such that the minimum distance between pairs of selected points is maximized. The objective is to have a dispersed set of points in space. Erkut [14] described the problem in detail. The author gave two integer programming formulation models and gave two Branch and Bound methods and a two stage heuristic procedure to solve the problem. Erkut, Iksal and Yenicerioglu [15] compared 10 different heuristics available for solving the p-dispersion problem. The p-dispersion problem has been used in various contexts like military installations to avoid enemy attack, location of fast food franchise in an urban area, etc.

For our problem, we have used the two stage heuristic to solve the p-dispersion problem given in [14]. In the classical p-dispersion problem, p out of m given points ($1 < p < m$) are selected in some space, where the objective is to maximize the minimum distance between any two of the selected points [7]. If M is the set of candidate points ($|M| = m$), and $P \subseteq M(|P| = p)$ and W_{ij} is the distance between the candidates i and j, the p-dispersion problem can be presented as:

$$\max_{P \subseteq M} \left[\min_{i,j=1,\ldots,p,\ i \neq j} \right] \{W_{ij}\}. \qquad (7)$$

Here in our case, W_{ij} is the dissimilarity between two paths. Hence, given m paths p paths are chosen in such a way so as to maximize the minimum dissimilarity between any two paths. The basic idea of using the two phase heuristic is to construct an initial solution in a semi-greedy fashion and then to perform a local search to improve the initial solution [14].

There exist several variants of shortest path algorithms that can be used to obtain the candidate set of paths for the p-dispersion problem. Some of the methods we have considered are:

1. ***k*-shortest path method**

   Yen [16] presented an algorithm to find the k shortest loopless paths. These paths tend to be very similar, i.e., sharing various links, implying that a very large value of k needs to be used.

2. **Iterative Penalty Method (IPM)**

   The IPM is based on a repetitive application of an appropriate shortest path algorithm. After each application a penalty on links is imposed so as to generate a different path. This method was suggested by Johnson, Joy, Clarke and Jacobi [8].

3. **Gateway Shortest Paths (GSPs)**

   Proposed by Lombard and Church [9] this method tries to find dissimilar paths by forcing the paths to go through a series of specific nodes called "gateways".

4. **Minimax Method**

   Proposed by Kuby, Zhongyi and Xiaodong [10], this method aims to generate a set of dissimilar paths by selecting a subset of a large set of paths. First k-shortest paths are generated and then a dissimilar subset is constructed iteratively.

Generally the IPM and k-shortest path method give the best results.

3.3.2.2 Robust path generation

The p-dispersion heuristic described above is used to get a set of paths for a single O-D pair. In our problem we are having n different O-D pairs, so we need to apply the p-dispersion heuristic n times.

After we get a set of paths for each O-D pair, we try to find the dissimilarity index between any two sets of O-D pairs (W_{ij}). A dissimilarity index described in [7] is used here and is modified for our purpose. The similarity index between two O-D pairs is defined as

$$S(P_i, P_j) = \frac{\dfrac{L(P_i \cap P_j)}{L(P_i)} + \dfrac{L(P_j \cap P_i)}{L(P_j)}}{2}, \qquad (8)$$

hence dissimilarity = $W_{ij} = 1 - S(P_i, P_j)$.

This is the dissimilarity index considering just the spatial element. To account for temporal dissimilarity we modify this index as:

$$W_{ij} = 1 - S^{\theta t}, \qquad (9)$$

where θ is a model parameter (needs to be calibrated) and t is the difference between the two start times of the ambulances for which we are calculating dissimilar routes.

Since we know the start time of the ambulance from its current location and the expected travel time to the destination, we also know its expected arrival time. Therefore, care is taken so that not many ambulances reach at the destination at the same time. This is done by varying by 30 minutes, 60 minutes, 90 minutes and 120 minutes and then seeing which one is giving the best dissimilarity index. Also sometimes we might not want to consider this temporal dissimilarity in the route generation. This is the case where we have to send a number of ambulances to the same spot given that a cluster of patients have been located and need to be taken to a hospital.

3.3.2.3 Solution to Quadratic Semi-Assignment Problem (QSAP)

The problem is solved using a Tabu search heuristic. After generating a set of paths for each O-D pair by using the p-dispersion heuristic, we determine the dissimilarity indices between various paths (among and across O-D pairs). A Tabu search technique is then used to find the solution. Indices are given as input to the Tabu search table. For each O-D pair a single route is generated which is spatially and temporally dissimilar.

Tabu search starts with an initial solution and then keeps improving it until the optimum is reached. This initial solution can be taken as any feasible solution or can be calculated by a simple regret heuristic given by Domschke, Forst and Voss [13]. Tabu search technique involves moving from one neighborhood solution to another. An active Tabu (forbidden) list is maintained to keep track of moves so that no move is repeated in the near future. The final output of this Tabu search technique is one solution for each O-D pair.

4 An Illustrative Numerical Example

We consider a road network as shown in Fig. 1. Suppose the status for one of the links obtained from the data fusion center is shown in Table 4. Hence the expected level of damage is

$$0.20 * 0.10 + 0.70 * 0.25 + 0.10 * 0.40 = 23.5\%$$

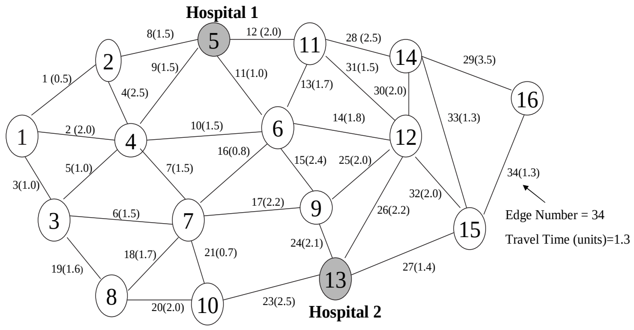

Fig. 1: An Example of a Road Network

This falls within the level 2 damage. Hence the speed of travel through link 10 will now be 30 mph instead of 40 mph. That means the corresponding increase in duration of link 10 is 4/3 times the original duration. Hence link 10's duration is updated in the network. We do this for all the other links for which we have reports of some damage.

Suppose we have two hospitals which are located at node 5 and node 13, respectively. Two ambulances are servicing this area. Each cluster contains no more than 3 patients. The data are shown in Table 5.

By using the expression developed in Sec. 3.1.2, the RPV of current cluster can be easily calculated by $\sum_{i \in\ current} \alpha \sum_{j=1}^{3} w_j p_{ij}$. The expected clustered priority value of next potential cluster is given by the same method.

Then we can get the following results $\overline{RT}_1 = 29.86$, $\overline{RT}_2 = 23.99$, which shows that our method has an improvement of 19.66% compared to the method that always dispatches the closest ambulance.

Table 4: An Example of Fused Data of a Road

Link NO.	Level of Damage	With Probability
10	10%	20%
	25%	70%
	40%	10%

The patients join a spatial queue if both ambulances are busy. Suppose ambulance 1 is available at node 5 at time t. At that time four clusters of patients are waiting to be serviced at nodes 8, 9, 14 and 16, respectively. Without loss of generality, we assume the RPVs for the four clusters are all equal to 1. By using the shortest travel method, the patients at node 9 should be responded to. However, if we consider a small time window, it is not necessary to dispatch the ambulance to the closest cluster. Ambulance 2 may become available within that small time interval. In this case dispatching ambulance 1 to patients at node 8 is more efficient than the previous method, which can save approximate by 2.3 units of time.

After an ambulance has picked up patients, instead of only considering the travel time, we also incorporate the expected waiting time at the hospitals. Suppose at time t an ambulance picks up patients at node 7. At the same time the expected waiting times in hospital 1 and hospital 2 are 6 and 2 units of time, respectively. It is better to send the patients to hospital 2 rather than hospital 1, even though hospital 1 is closer to the patients.

After making decisions for dispatch, we need to allocate routes to the ambulances. Suppose we want to find dissimilar paths for O-D pairs 1-16, 2-15 and 3-14. We will first apply the k-shortest path method or the IPM method to find the candidate paths for using the p-dispersion heuristic. Then we apply the p-dispersion heuristic three times, once

Table 5: Parameters of the Example

Location of Ambulances	$\begin{cases} \text{Ambulance 1 at} & \text{node 5} \\ \text{Ambulance 2 at} & \text{node 13} \end{cases}$
Location of Current Cluster	node 7, 2 patients
Fused Data of Current Patients	Patient 1 $\begin{cases} p_{11} = 0.00 \\ p_{12} = 0.75 \\ p_{13} = 0.25 \end{cases}$ Patient 2 $\begin{cases} p_{21} = 0.00 \\ p_{22} = 0.25 \\ p_{23} = 0.75 \end{cases}$
Parameter of Injury Level	$\begin{cases} 10, & \text{severe injury class,} \\ 3, & \text{moderate injury class,} \\ 1, & \text{mild injury class} \end{cases}$
Parameter of Cluster	α = 1.5
Arrival Rates	$\lambda_{kr} = \begin{pmatrix} 30 & 20 & 30 & 20 & 30 & 20 & 20 & 10 & 10 & 20 & 20 & 20 & 10 & 10 \\ 20 & 30 & 20 & 20 & 30 & 30 & 30 & 10 & 10 & 30 & 10 & 10 & 10 & 10 \\ 30 & 20 & 30 & 10 & 20 & 10 & 20 & 10 & 10 & 30 & 10 & 10 & 10 & 10 \end{pmatrix}$
δt	0.05h

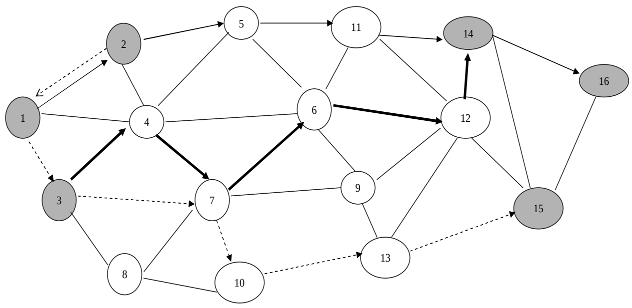

Fig. 2: Example of Finding Dissimilar Paths

for each of the three O-D pairs. Say 3 candidate paths are found for each O-D pair.

- For 1-16 pair

  $R_{11} \rightarrow 1-3-7-10-13-15-16,$

  $R_{12} \rightarrow 1-3-4-6-12-15-16,$

  $R_{13} \rightarrow 1-2-5-11-14-16.$

- For 2-15 pair

  $R_{21} \rightarrow 2-5-11-12-15,$

  $R_{22} \rightarrow 2-5-6-12-14-15,$

  $R_{23} \rightarrow 2-1-3-7-10-13-15.$

- For 3-14 Pair

  $R_{31} \rightarrow 3-7-10-13-12-14,$

  $R_{32} \rightarrow 3-4-5-11-14,$

  $R_{33} \rightarrow 3-4-7-6-12-14.$

Then the dissimilarity indices are found for various combinations of t, between various paths (among and across O-D pairs). A Tabu search technique is then used to find the solution. Indices are given as input to the Tabu search table. For each O-D pair a single route is generated, so that they are spatially and temporally dissimilar. Hence the final network with the routes generated is as follows:

$R_{13} \rightarrow 1-2-5-11-14-16,$
$R_{23} \rightarrow 2-1-3-7-10-13-15,$
$R_{33} \rightarrow 3-4-7-6-12-14.$

5 Conclusions and Future Work

In this paper, an attempt has been made to find the best strategy for the dispatching and routing of emergency vehicles in a dynamic disaster environment. All the information about the ground truth is collected and fused by the fusion center. This fused information is then sent to us to help in making the dispatching and routing decisions. We incorporate the future information in a small time window in the patient pickup policy. It has been illustrated that our policy outperforms the one that always sends the nearest ambulance. In the patient delivery problem, by taking account of the status of the hospitals, it is much better to incorporate the expected waiting time at hospitals than only considering the travel time to hospitals.

Future work will consider that the condition of a patient may deteriorate while he/she is waiting in queue. Taking this into consideration, it is important to consider a due date for each patient. A multi-server dynamic dispatching policy that schedules stochastic jobs with arbitrary I.I.D. processing times and distinct due dates will be developed using dynamic programming methods. A new routing algorithm will also be used to deal with large networks.

Acknowledgement

The work is funded by a grant from the Air Force Office of Scientific Research. This funding is gratefully acknowledged.

References

[1] M. Abudlghafour and M.A. Abidi. Data fusion through non-deterministic approaches: a comparison. In *Proceedings of the SPIE, 2059, Sensor Fusion VI*, pages 37–53, Sept. 1993.

[2] X.E. Gros. *NDT Data Fusion*. Arnold, London, 1997.

[3] C. Schaack and R.C. Larson. An n-server cutoff priority queue. *Oper. Res.*, 34(2):257–266, 1986.

[4] S. Mishra, R. Batta, and R.J. Szczerba. A rule based approach for aircraft dispatching to emerging targets. *Submitted to Mil. Oper. Res.*, 2003.

[5] J.Y. Choi, C.M. Rump, and G.L. Rogova. Stochastic scheduling for minimizing the expected number of tardy jobs with arbitrary distributions. In *INFORMS 2002*, San Jose, Nov. 2002.

[6] M. Gendreau and J.Y. Potvin. Dynamic vehicle routing and dispatching. Working paper, Université de Montréal.

[7] V. Akgun, E. Erkut, and R. Batta. On finding dissimilar paths. *Euro. J. Oper. Res.*, 121(2):232–246, 2000.

[8] P.E. Johnson, D.S. Joy, D.B. Clarke, and J.M. Jacobi. An enhanced highway routing model: program, description, methodology, and revised user manual. Technical Report ORNL/TM-12124, Oak Ridge National Lab, Oak Ridge, Tennessee, 1992.

[9] K. Lombard and R.L. Church. The gateway shortest path problem: generating alternative routes for a corridor location problem. *Geog. Syst.*, pages 25–45, 1993.

[10] M. Kuby, X. Zhongyi, and X. Xiaodong. A minimax method for finding the k best differentiated paths. *Geog. Anal.*, 29(4):298–313, 1997.

[11] K. Tyagarajan, R. Batta, M.H. Karwan, and R. Szczerba. Routing aircraft to minimize the chance of detection during mission ingress. *Submitted to Mil. Oper. Res.*, 2003.

[12] R. Gopalan, K. Kolluri, Batta R., and M.H. Karwan. Modeling equity of risk in the transportation of hazardous materials. *Oper. Res.*, 38(6):961–973, 1990.

[13] W. Domschke, P. Forst, and S. Voss. *Tabu Search Techniques for the Quadratic Semi-assignment Problem*. Springer-Verlag, Berlin, 1992.

[14] E. Erkut. The discrete *p*-dispersion problem. *Euro. J. Oper. Res.*, 46(1):48–60, 1990.

[15] E. Erkut, Y. Iksal, and O. Yenicerioglu. A comparison of *p*-dispersion heuristics. *Comput. Oper. Res.*, 21(10):1103–1113, 1994.

[16] J.Y. Yen. Finding the k shortest loop less paths in a network. *Management Sci.*, 17(11):712–716, 1971.