DEVELOPMENT OF ADVANCED APPLICATIONS USING BLUETOOTH-GENERATED TRAFFIC FLOW DATA

2010 Annual Report

Prepared by

Ali Haghani
Abbas M. Afshar

University of Maryland, College Park

January 2010

Abstract

The goal of this research is to develop a comprehensive model that describes the integrated logistics operations in response to natural disasters. We propose a mathematical model that controls the flow of several relief commodities from the sources through the supply chain and until they are delivered to the hands of recipients. The structure of the network is in compliance with FEMA's complex logistics structure. The proposed model not only considers details such as vehicle routing and pick up or delivery schedules; but also considers finding the optimal locations for several layers of temporary facilities as well as considering several capacity constraints for each facility and the transportation system. Such an integrated model provides the opportunity for a centralized operation plan that can eliminate delays and assign the limited resources to the best possible use.

A set of numerical experiments is designed to test the proposed formulation and evaluate the properties of the optimization problem. The numerical analysis shows the capabilities of the model to handle the large-scale relief operations with adequate details. However, the problem size and difficulty grows rapidly with adding the equity constraints and extending the length of the operations.

Two sets of heuristic algorithms are proposed to solve the proposed mathematical model. To solve the multi-echelon facility location problem, four heuristic solution techniques are proposed. Also four heuristic algorithms are proposed to solve general integer vehicle routing problem. The proposed heuristics were successful in efficiently solving the mathematical model. In one example, the heuristics were able to solve the problem in less than two minutes compared to the commercial solver that would take several hour of CPU time.

The organization of the paper is as follow: in the first chapter the attempts required to introduce, define, and mathematically model the problem is described. In the second chapter, solution approaches are investigated and two sets of heuristic algorithms are described that can solve the model proposed in the first chapter.

The first part of this research presented in chapter 1 was partially done with the help of Grant DTRT07-G-0003 from Mid-Atlantic Universities Transportation Center (MAUTC). It is reported in this document to keep the continuity and lay the foundation for the rest of the research which is conducted under current project from CITSM.

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION AND MODELLING

1.1 INTRODUCTION

In today's society that disasters seem to be striking all corners of the globe, the importance of emergency management is undeniable. Much human loss and unnecessary destruction of infrastructure can be avoided with more foresight and specific planning. Emergency management (or disaster management) is the discipline of avoiding risks and dealing with risks (Haddow, et al. 2007). No country and no community are immune from the risk of disasters. However, it is possible to prepare for, respond to and recover from disasters and limit the destructions to a certain degree. Emergency management is a discipline that involves preparing for disaster before it happens, responding to disasters immediately, as well as supporting, and rebuilding societies after the natural or human-made disasters have occurred. Emergency management is a continuous process. It is essential to have comprehensive emergency plans and evaluate and improve the plans continuously. The related activities are usually classified as four phases of Preparedness, Response, Recovery, and Mitigation. Appropriate actions at each phase in the cycle lead to greater preparedness, better warnings, reduced vulnerability or the prevention of disasters during the next iteration of the cycle.

During emergencies various aid organizations often face significant problems of transporting large amounts of many different commodities including food, clothing, medicine, medical supplies, machinery, and personnel from different points of origin to different destinations in the disaster areas. The transportation of supplies and relief personnel must be done quickly and efficiently to maximize the survival rate of the affected population and minimize the cost of such operations.

Federal Emergency Management Agency (FEMA) is the primary organization responsible for preparedness and response to federal level disasters in the United States. The primary mission of FEMA is "to reduce the loss of life and property and protect the nation from all hazards, including natural disasters, acts of terrorism, and other man-made disasters, by leading and supporting the nation in a risk-based, comprehensive emergency management system of preparedness, protection, response, recovery, and mitigation." (www.fema.gov)

FEMA has a very complex logistics structure to provide the disaster victims with critical items after a disaster strike which involves multiple organizations and spreads all across the country. There are seven main components in the supply chain to provide relief commodities for disaster victims that are briefly described here:

FEMA Logistics Centers (LC) - permanent facilities that receive, store, ship, and recover disaster commodities and equipment. FEMA has a total of 9 logistics centers.

Commercial Storage Sites (CSS) - permanent facilities that are owned and operated by private industry and store commodities for FEMA. Freezer storage space for ice is an example.

Other Federal Agencies Sites (VEN) - representing vendors from whom commodities are purchased and managed. Examples are Defense Logistics Agency (DLA) and General Services Administration (GSA).

Mobilization (MOB) Centers - temporary federal facilities in theater at which commodities, equipment and personnel can be received and pre-positioned for deployment as required. In MOBs commodities remain under the control of FEMA logistics headquarter and can be deployed to multiple states. MOBs are generally projected to have the capacity to hold 3 days of supply commodities.

Federal Operational Staging Areas (FOSAs) - temporary facilities at which commodities, equipment and personnel are received and pre-positioned for deployment within one designated state as required. Commodities are under the control of the Operations Section of the Joint Field Office (JFO) or Regional Response Coordination Center (RRCC). Commodities are usually being supplied from MOB Centers, Logistics Centers or direct shipments from vendors. FOSAs are generally projected to hold 1 to 2 days of commodities.

State Staging Areas (SSA) - temporary facilities in the affected state at which commodities, equipment and personnel are received and pre-positioned for deployment within that state. Title transfers for delivered federal commodities and cost sharing are initiated in SSAs.

Points of Distribution (PODs) Sites - temporary local facilities in the disaster area at which commodities are distributed directly to disaster victims. PODs are operated by the affected state.

Figure 1 better illustrates this structure. At the top of the pyramid there are 3 types of facilities: FEMA Logistics Centers, Commercial Storage Sites, and Other Federal Agencies or Vendors.

These permanent facilities store and ship commodities and equipment and are considered as "sources" in the chain. Mobilization Centers, Federal Operational Staging Areas, and State Staging Areas are 3 types of facilities that mainly play the role of "transshipment" points. These are temporary facilities at which commodities, equipment and personnel are received and pre-positioned for deployment to the lower levels. At the end, Points of Distribution Sites are temporary local facilities at which commodities are received and distributed directly to disaster victims and can be considered as "demand" points. PODs can be local schools, churches, or big parking lots inside the affected area.

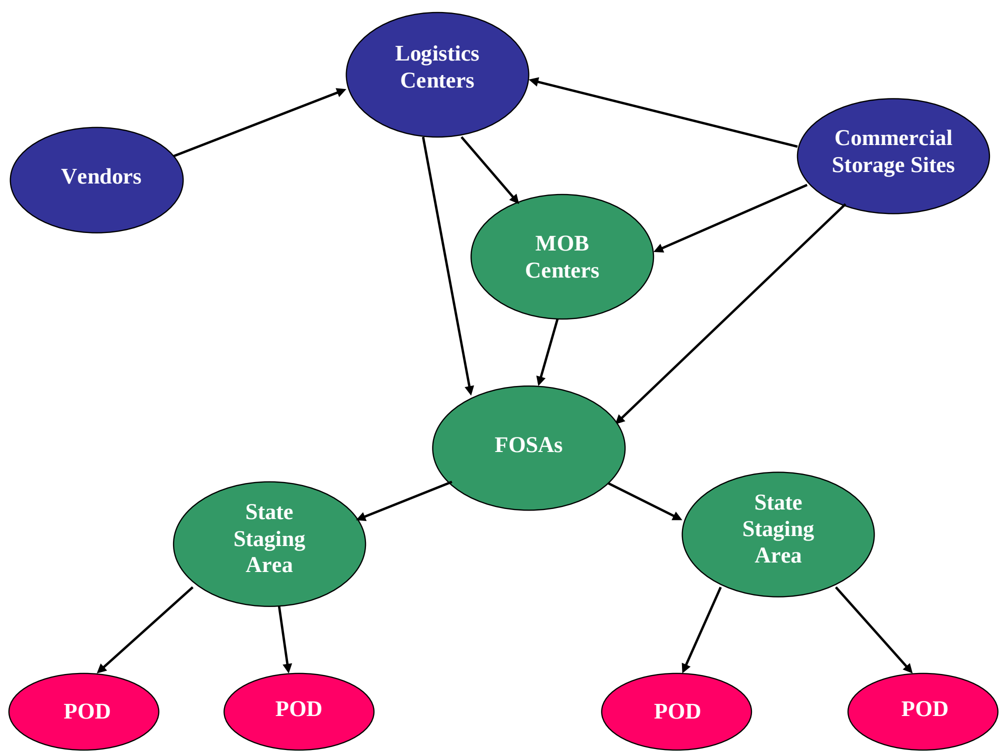

Figure 1 FEMA's Supply Chain Structure

Even this simplified presentation of the FEMA's logistics supply chain indicates the complex structure of the system. Finding the optimal sites for 4 levels of temporary facilities is a complicated location finding problem. Delivering several types of commodities to disaster victims is a multi-commodity capacitated network flows problem. Optimizing the movement of vehicles in the network is a dynamic vehicle routing problem with mixed pick up and delivery operations. Usually more than one transportation mode is used in disaster response operations which makes the problem a multimodal transportation problem. Other characteristics that make the problem unique include, but are not limited to, importance of quick response and fast delivery, shortage of supply versus overwhelming demands, insufficient capacity of facilities and transportation system, and dynamic environment of the emergency situations.

The goal of this research is to develop a comprehensive model that describes the integrated supply chain operations in response to natural disasters. An integrated model that captures the interactions between different components of the supply chain is a very valuable tool. It is ideal to have a model that controls the flow of relief commodities from the sources through the chain and until they are delivered to the hands of recipients. This research will offer a model that not only considers details such as vehicle routing and pick up or delivery schedules; but also considers finding the optimal location for temporary facilities as well as considering the capacity constraints for each facility and the transportation system. Such a model provides the opportunity for a centralized operation plan that can eliminate delays and assign the limited resources in a way that is optimal for the entire system.

1.2 LITERATURE REVIEW

Altay and Green (2006) surveyed the existing literature of emergency disaster management. They concluded that most of the disaster management research was related to social sciences and humanities literature. However, they realized the literature trend that more studies are focusing on OR/MS techniques in recent years and emphasized the need for more research in future. In the following, a summery of studies is presented that used OR/MS techniques to model and optimize the emergency disaster management activities. This is not an exclusive list of publication in the field and is only intended to focus on key studies in the past that successfully used techniques that are relevant to the subject of this research.

Haghani and Oh (1996) proposed a formulation and solution of a multi-commodity, multi-modal network flow model for disaster relief operations. Their model can determine detailed routing and scheduling plans for multiple transportation modes carrying various relief commodities from multiple supply points to demand points in the disaster area. They formulated the multi-depot mixed pickup and delivery vehicle routing problem with time windows as a special network flow problem over a time-space network. The objective was minimizing the sum of the vehicular flow costs, commodity flow costs, supply/demand storage costs and inter-modal transfer costs over all time periods.

Barbarosoglu et al. (2002) focused on tactical and operational scheduling of helicopter activities in a disaster relief operation. They proposed a bi-level modeling framework to address the crew assignment, routing and transportation issues during the initial response phase of disaster management in a static manner. The top level mainly involves tactical decisions of determining the helicopter fleet, pilot assignments and the total number of tours to be performed by each helicopter. The base level addresses operational decisions such as the vehicle routing of helicopters from the operation base to disaster points in the emergency area given the solution of the top level.

Barbarosoglu and Arda (2004) developed a two-stage stochastic programming model for transportation planning in disaster response. They expanded on the deterministic model of Haghani and Oh (1996) by including uncertainties in supply, route capacities, and demand requirements. The authors designed 8 earthquake scenarios to test their approach on real-world problem instances. It is a planning model that does not deal with the important details that might be required at strategic or operational level. It does not address facility location problem or vehicle routing problem.

Ozdamar et al. (2004) addressed an emergency logistics problem for distributing multiple commodities from a number of supply centers to distribution centers near the affected areas. They formulated a multi-period multi-commodity network flow model to determine pick up and delivery schedules for vehicles as well as the quantities of loads delivered on these routes, with the objective of minimizing the amount of unsatisfied demand over time. The structure of the proposed formulation enabled them to regenerate plans based on changing demand, supply quantities, and fleet size.

Yi and Ozdamar (2007) proposed a model that integrated the supply delivery with evacuation of wounded people in disaster response activities. They considered establishment of emergency facilities in disaster area to serve the medical needs of victims immediately after disaster. They used the capacity of vehicles to move wounded people as well as relief commodities. Their model resulted in a more compact formulation but post processing was needed to extract detailed vehicle routing and pick up or delivery schedules.

In a more recent study, Balcik and Beamon (2008) proposed a model to determine the number and locations of distribution centers in relief operations. They formulated the location finding problem as a variant of maximum covering problem for a set of likely scenarios. Their objective function maximizes the total expected demand covered by the established distribution centers. They also solve for the amount of relief supplies to be stocked at each distribution center to meet the demands. Their study is one of the first to solve location finding problem in relief operation; however, they do not consider the location problem as part of a supply chain network.

Based on our literature review, there are not many publications that directly applied network modeling and optimization techniques in disaster response. Among those studies, there is no model that has integrated the interrelated problems of large-scale multi-commodity multimodal network flow problem, vehicle routing problem with split mixed pick up and delivery, and optimal location finding problem with multiple layers. Also to the best of our knowledge, there is no mathematical model that describes the special structure of FEMA's supply chain system.

1.3 PROBLEM DESCRIPTION

Logistics planning in emergencies involves sending multiple relief commodities (e.g., medicine, water, food, equipment, etc) from a number of sources to several distribution points in the affected areas through a chain structure with some intermediate transfer nodes. The supplies may not be available immediately but arrive over time. It is a difficult task to decide on the right type and quantity of relief items, the sources and destinations of commodities, and also how to dispatch relief items to the recipients in order to minimize the pain and sufferings for disaster victims.

It is necessary to have a quick estimation of the demands during the initial response time. It is essential to know the types of required commodities, the amount of each commodity per person or household, an estimation of the number of victims, and the geographical locations of the demands. The list of commodities includes but is not limited to water, food, shelter, electric generators, medical supplies, cots, blankets, tarps and clothing. Some of the demand items are one-time demand while others are recurring (e.g. tent vs. water) and some demands are subject to expiration while others may be carried over (e.g. food vs. clothing). The demand usually overwhelms the capacity of the distribution network. The demand information might not be complete and accurate at the beginning but it is expected to improve over time.

Different aid organizations may employ their unique supply chain structure that governs the types of facilities to be used and the relationships among components of the chain. For example FEMA has its own supply chain structure for disaster response which is previously introduced in section 2. FEMA has distinguished 7 layers of facilities in its logistics chain. First 3 layers are permanent facilities to store and ship the relief items while the next 4 layers are temporary transfer facilities that their numbers and locations will be chosen during the response phase.

During the initial response time it is also necessary to set up temporary transfer facilities to receive, arrange, and ship the relief commodities through the distribution network. In risk mitigation studies for disasters, possible sites where these facilities can be situated are specified. Logistics coordination in disasters involves the selection of sites that result in the maximum coverage of affected areas and the minimum delays for supply delivery operations. Usually the number of these temporary facilities is limited because of the equipment and personnel constraints.

Each facility in the chain is subject to some capacity constraints. Capacities are defined for operations such as sending, receiving, and storing commodities. These capacities are different for each facility and are determined based on the type, size and layout of that facility. Also the availability of personnel and equipment may influence the capacities. In general, the capacity constraints can be defined in terms of the weight or volume of the commodities or they can be defined in terms of the numbers of the vehicles that are sent, received, or parked at the facility at a certain time. These are two different aspects and it is recommended to consider both capacities for each facility.

The transportation capacity is usually very limited in early hours or days after a disaster. It is very critical to assign the available fleet to the best possible use at any time. There is usually a shortage of vehicles in emergency operations so the model must keep track of the empty trucks in order to assign them to new missions after each delivery. More than one transportation mode may be hired to facilitate emergency response logistics. Consequently, the coordination and cooperation between transportation modes are necessary for managing the response operations and providing a seamless flow of relief commodities toward the aid recipients. The intermodal transfer of commodities is expected to happen in specific facilities but may be subject to some capacity constraints and transfer delays.

Vehicle routing and scheduling during the disaster response is also very important. A large number of vehicles might be used in response to large-scale disasters. The model should be able to keep track of routings for each individual vehicle. Also, it is required to have a detailed schedule for pick up and delivery of relief commodities by each vehicle in each transportation mode. Nonetheless, the vehicle routing in disaster situations are quite different from conventional vehicle routings. The vehicles do not need to form a tour and return to the initial depot, but they might be assigned to a new path at any time. They are expected to perform mixed pickup and delivery of multiple items between different nodes of the network as the supplies and demands arise over time.

The disaster area is a dynamic environment and emergency logistics are very time sensitive operations. The disaster might still be evolving when the response operations start. Also the lack of vital information about available infrastructure, supplies, and demands in the initial periods after the disaster may complicate this dynamic environment even more. The high stake of life-or-death for disaster victims urges the needs for higher levels of accuracy and tractability. Despite all the necessary preparedness and planning at strategic level, dealing with the problem at operational level is very important. Modeling and optimization at operation level is a necessary approach to capture the realities of time sensitive emergency response operations.

The other important issue is considering equity and fairness among aid recipients. Based on the geographical dispersion of victims and availability of resources over time and space, it is easy to favor the demands of one group of victims over another. Even though some variations are inevitable, the ideal pattern is to distribute the help items evenly and fairly among the victims. The models and procedures with general objective functions are prone to ignore the equity and level of service requirements in order to get a better numerical solution. It is very important to realize the need for procedures and constraints that prevent any sort of discrimination among victims, as much as possible.

The equity constraint between populations can be defined over time, and over commodities. It is not appropriate to satisfy all the demands of one group in early stages while the other group of victims does not receive any help until very later times. It is more acceptable to fairly distribute the available relief items among all recipients even though it might not be enough for every one at the current instance. The relief operations will continue over time as more resources are expected to become available. The equity over commodities is also important. For example, it is not acceptable to send all the available water to one group of victims and send all of the available meals to another group. It is expected to fairly share the limited resources of transportation capacity and disaster relief commodities.

Some main characteristics of the modeling approach can be summarized as follow:

- **Operational Level:** to capture time sensitive details of the emergency response operations, the problem is formulated at operational level.
- **FEMA Structure:** the proposed model is in compliance with FEMA's 7-layer supply chain structure.
- **Time-Space Network:** to account for the dynamic decision process, the physical network must be converted to a time-space network. The nodes of this network represent the facilities in FEMA structure. The links consist of existing physical links, delay or storage links, and intermodal transfer links.
- **Facility Location:** the optimal locations to establish temporary facilities are selected from a set of potential sites. The maximum number of each facility type and their locations are dynamic and can change over time as the relief operations proceed.
- **Facility Capacity:** each facility has maximum capacities for sending, receiving, and storing commodities as well as vehicles.
- **Demand:** the demand is multi-commodity and usually overwhelms the capacity of the distribution network. Specific decision variables are defined that keep track of unsatisfied demand at each demand point for each commodity and during all time periods.
- **Supply:** similar to the demand, the supply is multi-commodity and may come from various sources. The problem is formulated as a variation of multi-commodity network flow problem.
- **Multi-Modal:** since more than one mode of transportation may be hired in the emergency response logistics, the problem is a variation of multi-modal network flow problem.
- **Vehicle Routing:** in order to model the complicated routing and delivery operations in disaster response, the vehicles are treated as flow of integer commodities over a time-

space network. This results in a mixed integer multi-commodity formulation which is very flexible.

- **Network Capacity:** a set of constraints is used to link the relief commodities with the vehicles. As a result, the flow of commodities is only possible when accompanied by vehicles with enough capacity for that specific time and route.
- **Integrated Model:** all decisions of facility location, supply delivery, and vehicle routing, are interrelated. Our approach provides an integrated model to find the global solution for this problem.
- **Equity:** equity and fairness among disaster victims is modeled through a set of constraints that enforce a minimum level-of-service for each victim. The equity can be enforced for each relief item and over all time periods.
- **Objective Function:** the objective of this model is to minimize the pain and suffering of the disaster victims. It is formulated as weighted total of unsatisfied demand summed over all victims, for all relief items, and during all time periods.

1.4 PROBLEM FORMULATION

In this section initially the notations and required parameters for the formulation are introduced. After that, the decision variables of the mathematical model are defined. Then the objective function formulation is presented followed by formulation and introduction of the constraints of the problem.

1.4.1 Notations

N = Set of all nodes. $i, j \in N$ are indices

LC = Set of Logistic Center sites

CSS = Set of Commercial Storage Sites

VEN = Set of commodity Vendor sites

MOB = Set of potential sites for Mobilization Centers

FOSA = Set of potential sites for Federal Operational Staging Areas

SSA = Set of potential sites for State Staging Areas

POD = Set of Points of Distribution (demand nodes)

U = Set of supply nodes and transshipment nodes (LC, VEN, CSS, MOB, FOSA, SSA)

V = Set of Permanent Facilities (LC, CSS, VEN)

W = Set of potential sites for all Temporary Facilities (MOB, FOSA, SSA)

C = Set of Commodities, $c \in C$ is an index

M = Set of transportation Modes, $m \in M$ is an index

T = Time horizon of response operations. $t, t' \in T$ are indices

1.4.2 Parameters

Supply and Demand

Sup_{it}^{c} = Amount of exogenous supply of commodity type c in node i at time t

Dem_{it}^{c} = Amount of exogenous demand of commodity type c in node i at time t

AV_{it}^{m} = Number of vehicles of mode m added to the network in node i at time t, negative if vehicles removed

RU_{it}^{c} = Relative urgency of one unit of commodity c, in node i at time t

Number of Facilities

$MOB_{\max}^{t}$ = Maximum number of Mobilization centers at time t

$FOSA_{\max}^{t}$ = Maximum number of Federal Operational Staging Areas at time t

$SSA_{\max}^{t}$ = Maximum number of State Staging Areas at time t

Facility Capacity

$Ucap_{it}^{m}$ = Unloading capacity for the facility in node i for mode m at time t

$Scap_{it}$ = Storage capacity for the facility in node i at time t

$Lcap_{it}^{m}$ = Loading capacity for the facility in node i for mode m at time t

$VRcap_{it}^{m}$ = Maximum number of mode m vehicles that can be received at the facility in node i at time t

$VPcap_{it}^{m}$ = Maximum number of mode m vehicles that can be parked (carried over) at the facility in node i from time t to time t + 1

$VScap_{it}^{m}$ = Maximum number of mode m vehicles that can be sent out from the facility in node i at time t

Vehicle Capacity

cap_{m} = Loading capacity of vehicles of mode m

w_{c} = Unit weight of commodity c

Transportation

t_{ijm} = Travel time from node i to node j for vehicles of mode m

$K_{mm'}$ = Time required to transfer commodities from mode m to mode m'

1.4.3 Decision Variables

Location Problem

Loc_i^t = 1 if temporary facility of appropriate type is located at potential site i, at time t; equal to 0 otherwise. The temporary facility will be a Mobilization Center if $i \in MOB$, a Federal Operational Staging Area if $i \in FOSA$, and a State Staging Area if $i \in SSA$.

Commodity and Vehicle Flow

X_{ijt}^{cm} = Flow of commodity type c shipped from node i to node j by mode m at time t

Y_{ijt}^{m} = Flow of vehicles of mode m from node i to node j at time t

CX_{it}^{c} = Amount of commodity type c in node i which is carried over from time period t to t + 1

CY_{it}^{m} = Number of vehicles of mode m in node i which is carried over from time period t to t + 1

$XT_{it}^{cmm'}$ = Amount of commodity type c in node i which is transferred from mode m to mode m′ at time t

UD_{it}^{c} = Amount of unsatisfied demand of commodity type c in node i at time t

1.4.4 Objective Function

Minimize $$\sum_{i \in V} \sum_{t} \sum_{c} RU_{it}^{c} \cdot UD_{it}^{c} \qquad (1)$$

The objective function in equation (1) minimizes the total amount of weighted unsatisfied demand over all commodities, times, and demand points. RU_{it}^{c} is the relative urgency associated with each commodity, time, and demand point. If there is any desire to consider a commodity being more important than others at any time or for any demand point, RU_{it}^{c} can enforce that desire. Higher values of RU_{it}^{c} translate into higher urgencies. If all commodities happen to be of the same importance, RU_{it}^{c} can be set equal to 1.

1.4.5 Constraints

Commodity Flow Constraints

Supply nodes and Transfer nodes:

$$\sum_j X_{ji(t-t_{jim})}^{cm} + \sum_{m'} XT_{i(t-k_{m'm})}^{cm'm} + CX_{i(t-1)}^{c} + Sup_{it}^{c}$$
$$= \sum_j X_{ijt}^{cm} + \sum_{m'} XT_{it}^{cmm'} + CX_{it}^{c} \qquad \forall i \in U, c, m, t \tag{2}$$

Demand nodes:

$$\sum_m \sum_j X_{ji(t-t_{jim})}^{cm} + UD_{it}^{c} = Dem_{it}^{c} + UD_{i(t-1)}^{c} \qquad \forall i \in POD, c, t \tag{3}$$

Equations (2) and (3) enforce the conservation of the flow for all commodities and modes at all nodes and time periods. Equation (2) requires that for supply nodes and transfer nodes, the sum of the flows entering each node plus exogenous supply should be equal to the sum of the flows that leave the same node. Equation (3) shows that the total flow entering each demand node plus the unsatisfied demand is equal to the exogenous demand at that node plus any unsatisfied demand from the previous time period.

Vehicular Flow Constraints

$$\sum_j Y_{ji(t-t_{jim})}^{m} + CY_{i(t-1)}^{m} + AV_{it}^{m} = \sum_j Y_{ijt}^{m} + CY_{it}^{m} \qquad \forall i \in N, m, t \tag{4}$$

Equation (4) represents the conservation of flow for the vehicles. At any node i and time period t, total number of available vehicles of mode m is equal to the number of vehicles of mode m that left node j for node i at time $t - t_{ijm}$, plus the number of vehicles that were carried over from the previous time period, plus the number of vehicles that are added or removed to the fleet at that time. These vehicles are either sent out of the node or carried over to the next time period.

Linkage between Commodities and Vehicles

$$Cap_m \times Y_{ijt}^{m} \geq \sum_c w_c X_{ijt}^{cm} \qquad \forall i, j \in N, m, t \tag{5}$$

Constraint (5) makes sure that commodities are not sent out of a node unless a number of vehicles with enough capacity are available at that node to carry those commodities.

Facility Capacities for Permanent Facilities

$$\sum_c \sum_j X_{ijt}^{cm} \le Lcap_{it}^m \qquad \forall i \in V, m, t \tag{6}$$

$$\sum_c \sum_j X_{ji(t-t_{jim})}^{cm} \le Ucap_{it}^m \qquad \forall i \in LC, m, t \tag{7}$$

$$\sum_m \sum_c \sum_j X_{ji(t-t_{jim})}^{cm} + \sum_c CX_{i(t-1)}^c + \sum_c Sup_{it}^c \le Scap_{it} \qquad \forall i \in V, t \tag{8}$$

$$\sum_j Y_{ijt}^m \le VScap_{it}^m \qquad \forall i \in V, m, t \tag{9}$$

$$\sum_j Y_{ji(t-t_{jim})}^m + AV_{it}^m \le VRcap_{it}^m \qquad \forall i \in V, m, t \tag{10}$$

$$\sum_j Y_{ji(t-t_{jim})}^m + AV_{it}^m + CY_{i(t-1)}^m \le VPcap_{it}^m \qquad \forall i \in V, m, t \tag{11}$$

Equations (6), (7), and (8) are the maximum capacity for loading, unloading, and storage of commodities at permanent facilities. Equations (9), (10), and (11) require the maximum number of vehicles that are sent, received, and parked at each facility to be less than the relevant capacities.

Facility Location and Capacities for Temporary Facilities

$$\sum_c \sum_j X_{ijt}^{cm} \le Lcap_{it}^m \times Loc_i^t \qquad \forall i \in W, m, t \tag{12}$$

$$\sum_c \sum_j X_{ji(t-t_{jim})}^{cm} \le Ucap_{it}^m \times Loc_i^t \qquad \forall i \in W, m, t \tag{13}$$

$$\sum_m \sum_c \sum_j X_{ji(t-t_{jim})}^{cm} + \sum_c CX_{i(t-1)}^c + \sum_c Sup_{it}^c \le Scap_{it} \times Loc_i^t \qquad \forall i \in W, t \tag{14}$$

$$\sum_j Y_{ji(t-t_{jim})}^m + AV_{it}^m \le VRcap_{it}^m \times Loc_i^t \qquad \forall i \in W, m, t \tag{15}$$

$$\sum_j Y_{ji(t-t_{jim})}^m + AV_{it}^m + CY_{i(t-1)}^m \le VPcap_{it}^m \times Loc_i^t \qquad \forall i \in W, m, t \tag{16}$$

$$\sum_j Y_{ijt}^m \le VScap_{it}^m \times Loc_i^t \qquad \forall i \in W, m, t \tag{17}$$

$$\sum_i Loc_i^t \le MOB_{\max}^t \qquad \forall i \in MOB, t \tag{18}$$

$$\sum_i Loc_i^t \le FOSA_{\max}^t \qquad \forall i \in FOSA, t \tag{19}$$

$$\sum_i Loc_i^t \le SSA_{\max}^t \qquad \forall i \in SSA, t \tag{20}$$

Equations (12) through (14) enforce the loading, unloading, and storage capacity for temporary facilities. If the facility is selected to be set up at potential site i, the respected capacity constraint is enforced. If it is decided not to set up the temporary facility at location i, the same constraints require that all the flows in and out of that node to be equal to zero.

Equations (15) through (17) require the maximum number of vehicles that are sent, received, and parked at each temporary facility to be less than the relevant capacities. The numbers are zero if the facility is not selected for that node. Equations (18) through (20) oblige the maximum number of each temporary facility type to be limited by the maximum allowable numbers for that facility type during the chosen time periods.

Capacities for PODs:

$$\sum_{c}\sum_{j} X_{ji(t-t_{jim})}^{cm} \leq Ucap_{it}^{m} \qquad \forall i \in POD, m, t \tag{21}$$

$$\sum_{j} Y_{ji(t-t_{jim})}^{m} \leq VRcap_{it}^{m} \qquad \forall i \in POD, m, t \tag{22}$$

$$\sum_{j} Y_{ji(t-t_{jim})}^{m} + CY_{i(t-1)}^{m} \leq VPcap_{it}^{m} \qquad \forall i \in POD, m, t \tag{23}$$

Equation (21) enforces the commodity unloading capacity at points of distribution. Equation (22) and (23) represent the vehicle receiving and vehicle parking capacities for each point of distribution.

Equity Constraint:

$$\frac{\sum_{t'}\sum_{m}\sum_{j} X_{ji(t'-t_{jim})}^{cm}}{\sum_{t'} Dem_{it'}^{c}} \geq \alpha_{\min} \qquad \forall i \in POD, c, t \tag{24}$$

$$\frac{\sum_{t'}\sum_{m}\sum_{j} X_{ji(t'-t_{jim})}^{cm}}{\sum_{c}\sum_{t'}\sum_{m}\sum_{j} X_{ji(t'-t_{jim})}^{cm}} \geq \beta_{\min} \qquad \forall i \in POD, c, t \tag{25}$$

$$\frac{\sum_{c}\sum_{t'}\sum_{m}\sum_{j} X_{ji(t'-t_{jim})}^{cm}}{\sum_{i}\sum_{c}\sum_{t'}\sum_{m}\sum_{j} X_{ji(t'-t_{jim})}^{cm}} \geq \gamma_{\min} \qquad \forall i \in POD, t \tag{26}$$

Equation (24) enforces a minimum percentage of total demand for a specific commodity c, to be satisfied by the time period t. It might not be always possible to deliver the required amount to all demand nodes by time t; in that case, this constraint can cause infeasibility. Equation (25) requires that from all commodities being delivered to node i by time t, at least $\beta_{\min}$ percent to be commodity c. Equation (26) ensures that sum of total commodities delivered at point i to be more than a minimum percentage of all the commodities that are being delivered among all demand points.

Nonnegativity and Integrality:

$X_{ijt}^{cm}, CX_{it}^{c}, XT_{it}^{cmm'}, UD_{it}^{c} \geq 0$ Real-valued variables

$Y_{ijt}^{m}, CY_{it}^{m} \geq 0$ General integer variables

$LOC_{i}^{t} \in (0,1)$ Binary integer variables

1.4.6 Formulation Summary

The proposed mathematical model can be summarized as follows:

Minimize Total Weighted Unsatisfied Demand

Subject to:

Commodity Flow Constraints

Vehicular Flow Constraints

Constraints that Link Commodities and Vehicles

Facilities Location Constraints

Facility Capacities Constraints

Equity (recipients/commodities) Constraints

Nonnegativity and Integrality Constraints

1.5 NUMERICAL EXPERIMENT

In this section, a set of numerical experiments are conducted to evaluate the features of the proposed formulation. The problem size is kept small so it can be solvable by commercial solver and the results can be analyzed easier. However, the small-size problem still fully represents all elements of the proposed model. The experimental study complies with FEMA's structure and scale of the problem is comparable to the real-world-size problems.

The following example is an imaginary scenario where a natural disaster such as a hurricane strikes the southern coast of the United States. It is assumed that two separate regions, one in Mississippi and one in Louisiana, are affected.

For this example, it is assumed that only the Atlanta logistics center (LC) is used. One commercial storage site (CSS) in Charlotte, North Carolina and one vendor (VEN) in Nashville, Tennessee are also used to store the relief items. For temporary facilities at federal level, four potential sites for mobilization centers (MOB) are suggested. There are also four potential sites for federal operational staging areas (FOSA). These facilities are able to send supplies to both disaster areas. At the state level, a total of 10 potential sites for state staging areas (SSA) are suggested. Four potential SSA are planned to serve the disaster area in Mississippi and six potential SSA are suggested for Louisiana. The initial post-disaster surveys estimate that approximately 20'000 person are affected and twenty points of distribution (POD) are needed to serve this population. Eight PODs are selected for Mississippi area and twelve PODs will serve the victims in Louisiana. For this numerical study, there are a total of 41 permanent and temporary facilities in the network. Figure 2 illustrates the locations of these facilities on the map.

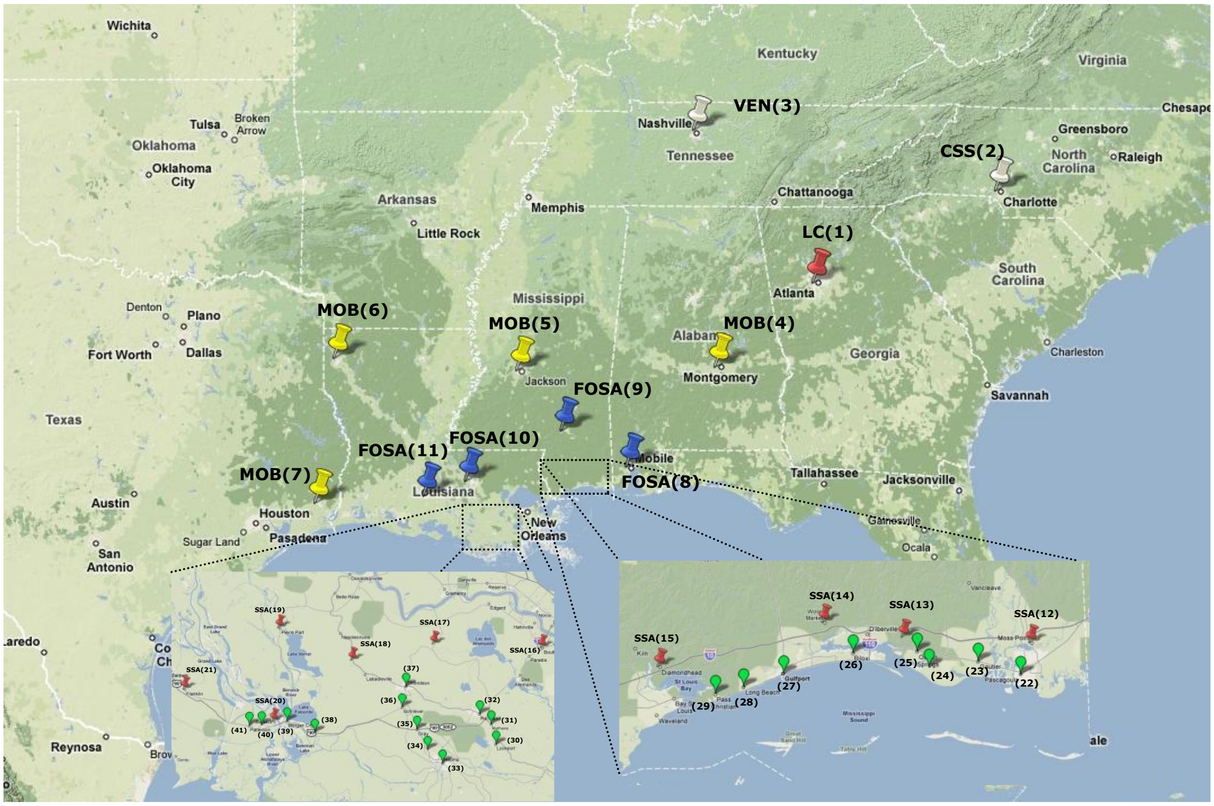

Figure 2 - Map of Federal Level and State Level Facilities

Supply and Demand

There are several commodities that need to be distributed among the disaster victims. The type and amount of each commodity depends on many factors such as type of disaster, level of destruction, weather conditions, etc. Table 1 suggests a list of required items and the amount per day per survivor. Since most items are bulky, volume capacity is expected to be binding versus the maximum weight load for each vehicle. Adding up the last column of Table 1, it can be seen that for each survivor a total of about 30 cubic ft of relief items per day are required. For the sake of simplicity, it is assumed that only 2 types of commodities (commodity 1 and commodity 2) are required in this numerical experiment. However, to preserve the scale of demands, the total amount per each survivor is kept at 30 ft^3 per day. It is also assumed that survivors in disaster zone 1 (Mississippi), need 20 cft of commodity 1 and 10 cft of commodity 2, per day. On the other hand, survivors in disaster zone 2 (Louisiana), assumed to need 10 cft of commodity 1 and

20 cft of commodity 2, per day. This will provide the opportunity to analyze the effects of different demand types on the model.

Table 1. List of Required Items for Survivors of a Disaster

Item	Quantity per day per survivor	Survivors served	Notional dimensions (ft)			Volume (ft^3)	Total requirement per survivor (ft^3)
			L	W	H		
Water (drinking)	1 gallon	1	1.0	1.0	1.0	1.0	1.000
Water (non-potable)	1 gallon	1	1.0	1.0	1.0	1.0	1.000
Meals (MREs)	3 meals	1	1.0	1.0	1.5	1.5	4.500
Portable shelter	1 shelter	4	6.0	2.0	1.5	18.0	4.500
Basic medical kit	1 kit	3	1.0	1.0	1.0	1.0	0.333
Cot	1 cot	2	3.0	2.0	1.0	6.0	3.000
Blanket	1 blanket	1	2.0	2.0	0.5	2.0	2.000
Tarp	1 tarp	3	3.0	3.0	1.0	9.0	3.000
Ice	1 gallon	10	1.0	1.0	1.0	1.0	0.300
Baby supplies	1 box	5	1.0	1.0	1.0	1.0	0.600
Generator	1 generator	500	8.0	8.0	6.0	384.0	0.768
Clothing	1 bag	1	2.0	2.0	1.0	4.0	4.000
Plywood	2 sheets	3	4.0	8.0	0.1	3.2	2.133
Nails	1 box	3	1.0	1.0	1.0	1.0	0.333

Source www.Fema.org

Supply sources are the Logistics Center, the Commercial Storage Site, and the Vendor. It is assumed that 40% of total supply is stored at the LC, 20% at the CSS, and 40% at the vendor site. Total demand for 20,000 survivors will be 600,000 ft3 per day. The demand for Commodity 1 is 280,000 ft3 per day and the demand for Commodity 2 is 320,000 ft3 per day. For this problem, it is assumed that supplies for one day are available and are stored at those three supply sources.

Vehicles

For this problem, only one transportation mode is used which is trucking. The common vehicle is a 53ft trailer truck which has the volume capacity of approximately 6000 cft. For the base case, 100 trucks are available at the beginning of the operations. Initially, 40 trucks are located at LC, while 30 trucks are at CSS and VEN sites, each.

Network links and Travel times

There are 2 types of flows in this problem, flow of commodities and flow of vehicles. The commodity flows must comply with the hierarchical structure of FEMA explained in Figure 1. For example, supplies from a VEN can only be sent to LC, or supply from LC can be sent to all MOBs and FOSAs. Supplies in MOBs can be sent to other MOBs or to FOSAs. Supplies from FOSAs can be sent to other FOSAs and to SSAs, as long as it remains in the same State. Supplies received at each SSA can be sent to other SSAs in the same State or must be delivered to PODs of that State.

The flow of vehicles in the network is much less restricted compared to commodity flows. It is assumed that there is a link between each pair of nodes in the network. Basically, empty vehicles are free to travel between each two nodes of the network without the need to visit any intermediate nodes. As a result, when a vehicle is carrying supplies, it must follow the more restricted hierarchical network of FEMA. But when the vehicle unloads all its supply, either at intermediate nodes or final PODs, it is free to go to any other node in the network to pick up supplies and start a new round of delivery.

Link travel time functions for the proposed formulation can be completely arbitrary. The formulation is capable of dealing with time-variable travel times as well as fixed travel times. For this numerical study, the travel distance between any two nodes of the network is assumed to be equal to their Euclidian distance. The travel speed is assumed to be fixed for all the vehicles on the federal level network (between LC, CSS, VEN, MOBs, and FOSA) and to be equal to 50 miles per hour. However, for State level network (between FOSAs, SSAs, and PODs) the travel speed is assumed to be 40 miles per hour.

Time Scale

Selection of appropriate time step is a very important factor that can affect the performance of time-space networks dramatically. For each time period in the planning horizon, one layer of physical network will be added to the problem. This makes the problem size grow extremely fast with the number of time steps in the planning horizon. For example if the planning horizon is only 1 day, with the choice of time step t = 1 minute, it will be 24 * 60 = 1440 layers of the network. So to keep the problem at a reasonable size, it is favorable to have longer time steps. On the other hand, shorter time steps will improve the accuracy of modeling the emergency response operations. For example if the time step is 1 hour, it is possible to model the state of the system only at every hour and not at the times in between. So from the accuracy perspective, it is favorable to have shorter time steps.

The other important issue in determining the time-step in this problem is the issue of dealing with very long and very short links. At the federal level network, nodes are usually far from each other and the links can range from a hundred miles to a few thousand miles. The travel time on those links with ground transportation can range from a few hours to up to one day or more. However, the nodes at the lower levels in the State networks can be very close to each other. It is very common to have PODs that are only a few miles apart. In this case, link travel times can be in the order of minutes. Figure 3 better shows the issue of scale in this problem on the disaster area map.

It is a difficult challenge to select a time-step that is suitable for very short links and very long links, at the same time. A very short time-step is necessary to model the short links even though it will increase the problem size very quickly. But the main issue is the sensitivity of travel times to the selected time-step. If a very short time-step is chosen, say 1 minute, it might be good for short links but the travel times on very long links will not be sensitive to that. It is very difficult, if not impossible, to predict the travel time between two nodes that are several hundred miles apart, with accuracy of 1 minute. For those links the 1-hour unit or 30-minute unit is more meaningful.

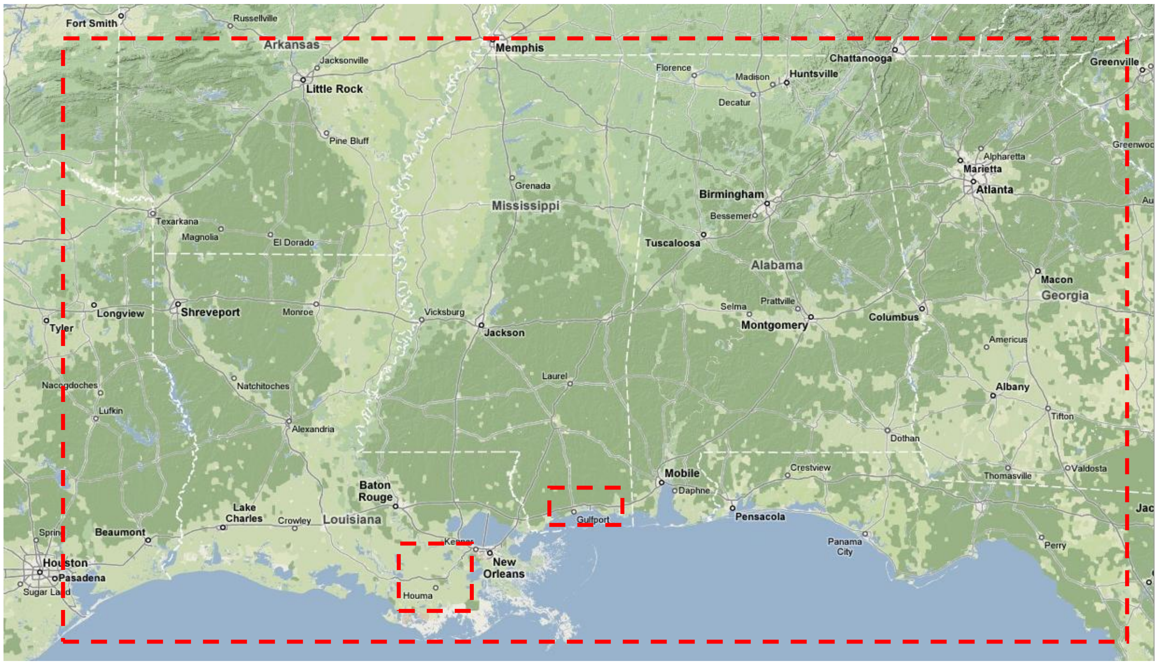

Figure 3 - Issue of Scale in Disaster Area

Geographical Decomposition

To deal with this issue, a geographical decomposition method is proposed. The nodes at federal level (LC, CSS, VEN, MOB, FOSA) will be in one subset and the nodes at each State (FOSA, SSA, POD) will form another subset. Since the travel times between nodes in federal level network are usually long, it is possible to use a large time-step for them. Using similar argument, the State level nodes and links can be modeled with a short time-step. Figure 4 shows this decomposition.

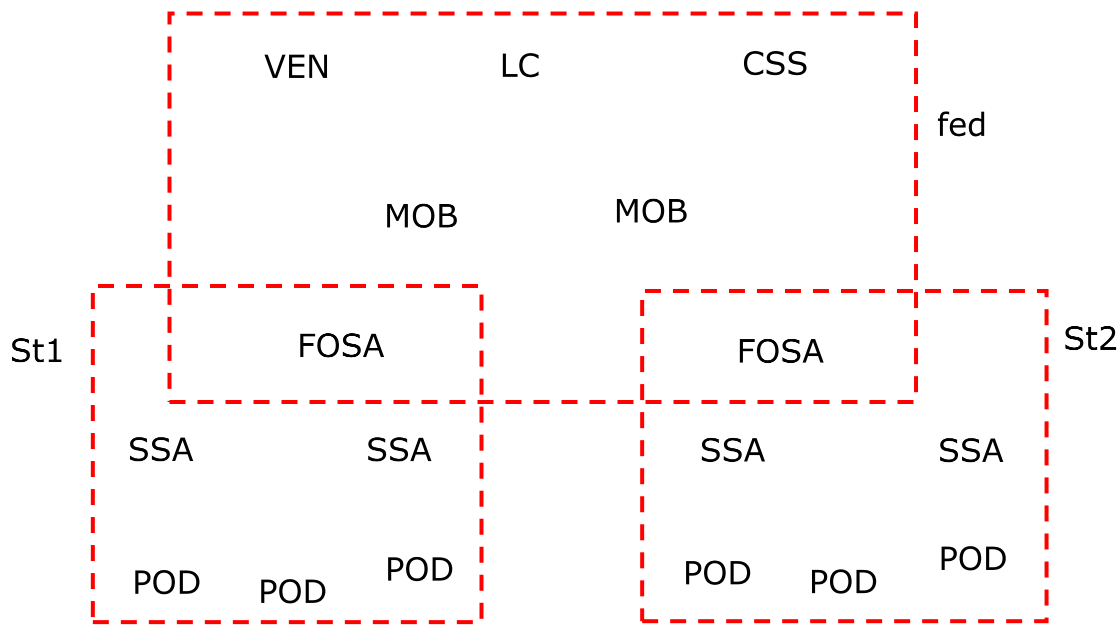

Figure 4 - Geographical Decomposition for Time Steps

Now the important issue is how to connect these separate time-space networks. Luckily, the special structure of FEMA's supply chain offers the candidates. Federal Operational Staging Areas (FOSA) are the one and only interface between flow of commodities in federal level facilities and the designated state level facilities. We take advantage of this opportunity and select the FOSAs as transfer terminals between the sub-networks. For this numerical study, time-step for federal zone, t_1, is chosen to be 30 minutes and time-step for state level zones, t_2, is selected to be 5 minutes. The travel times for this study are calculated based on the distance and a fixed average travel speed explained earlier. So based on the newly defined time steps of t_1 and t_2, travel times of federal zone links are being rounded to the nearest 30 minute interval and the travel times of state level zone links are being rounded to the nearest 5 minute.

The way in which the FOSA nodes connect two sub-networks with different time steps is shown in Figure 5. This graph indicates that the arcs entering FOSA from federal network or leaving the FOSA toward the federal network can exist only at t_1=30-minute intervals. But the arcs that connect FOSA to state level facilities exist for every t_2=5-minute interval. The implication is that the downward flows (from federal network to state network) entering a given FOSA can leave that FOSA at any 5-minute period after that. However, the upward flows (from state network to federal network) that enter a FOSA at any time other than 30-minute intervals, need to wait at the FOSA until the first available 30-minute interval.

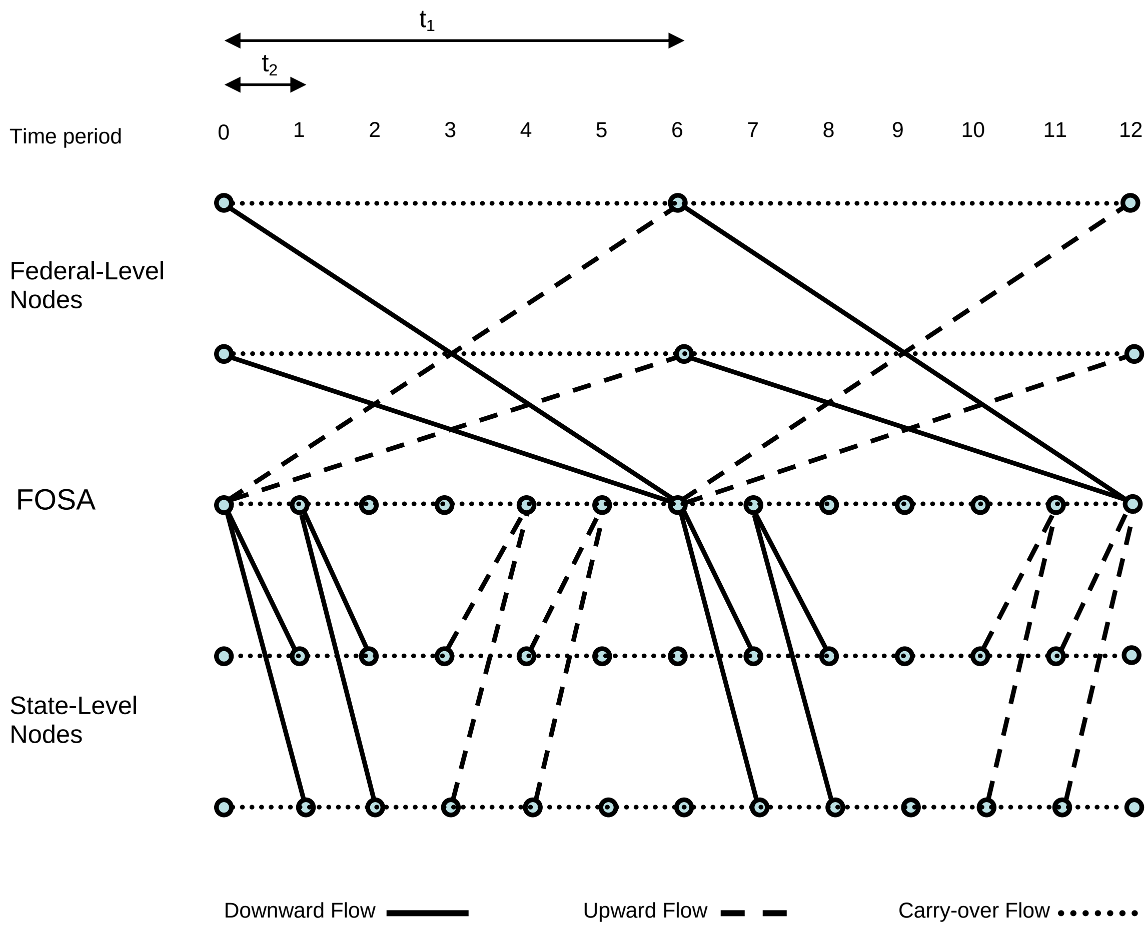

Figure 5. Time-Space network with Different Time Steps at FOSA

Case Study Scenarios

To better evaluate the characteristics of the proposed model, 10 numerical case studies are generated. All case studies are based on the described disaster scenario with variations in the subset of enforced constraints and some parameter values. Table 2 describes the considered case studies. In general, the case studies in Table 2 start from simple and become more complicated toward the end. For example, the first case study only considers the conservation of flow and vehicle capacity constraints. Other constraints are gradually added to the formulation in the other case studies up to Case 7 which has the largest number of constraint types for a one day operation. First seven case studies consider only one day of operations while in the last three cases two days of operations are formulated.

Table 2. Numerical Case-Study Descriptions

Case No	Constraints Used	Details	Variables		Constraints	File Size (Kb)
			Real Val.	Integer		
1	Flow Conservation + Vehicle Capacity	1 day 100 Trucks	133,275	157,972	81,891	13,331
2	Flow Conservation + Vehicle Capacity	1 day 200 trucks	133,275	157,972	81,891	13,331
3	Flow Conservation + Vehicle Capacity + Facility Capacity	1 day 100 Trucks	133,275	157,972	87,094	15,846
4	Flow + Facility Location (2,2,5)* + Facility Capacity	1 day 100 Trucks	133,275	157,972	87,094	15,846
5	Flow + Facility Location (2,2,2) + Facility Capacity	1 day 100 Trucks	133,275	157,972	87,094	15,846
6	Flow + Facility Capacity Const.+ Equity-1 Const	1 day 100 Trucks	133,275	157,972	87,174	17,214
7	Flow + Facility Location (2,2,5) + Facility Capacity + Equity-1,2,3	1 day 100 Trucks	133,275	157,972	87,294	61,084
8	Flow Conservation + Vehicle Capacity, day by day Supply	2 days 100 Trucks	265,995	315,316	163443	27,439
9	Flow + Facility Location (2,2,5) + Facility Capacity , day by day Supply	2 days 100 Trucks	265,995	315,316	173,878	32,673
10	Flow + Capacity + location (2,2,5) , 2 day supply available	2 days 100 Trucks	265,995	315,316	173,878	32,673

* Facility location with maximum number of (MOB, FOSA, SSA)

CPLEX commercial solver is used to solve the MIP model formulations. Table 3 summarizes the optimization results for all 10 case studies. Case-1 is the “base case” with only conservation of flow constraint and vehicle capacity constraints modeled for one day of operations. The solver found the optimal solution in approximately 4 minutes. Figure 6 shows the percent of unsatisfied demand for all victims over time. The first delivery to the nearest demand point took about 7 hours. Fifty percent of the total demand was satisfied after 11 hours and 40 minutes. The last demand was served after 21 hours and 40 minutes.

Table 3. Summary of Optimization Results

Case Number	Objective Value	Last UD (hr:min)	Temp. Facilities	Root Sol. Time (s)	Iterations	CPU Time (sec)†
1	9.0798 E+07	21:40	(4,4,10)	33.89	14,957	230
2	8.6118 E+07	15:10	(4,4,10)	10.36	5,502	20
3	1.0412 E+08	22:05	(4,4,10)	42.73	18,642	778
4	1.0412 E+08	22:05	(2,2,5)	33.59	17,308	945
5	1.0978 E+08	24:00§	(2,2,2)	204.19	205,588	5575
6	1.0439 E+08	21:50	(4,4,10)	42.22	5,810,980	45856*
7	1.0417 E+08	22:05	(2,2,5)	63.09	7,888,315	81642*
8	1.7985 E+08	39:10	(4,4,10)	786.34	63,960	4779
9	2.0859 E+08	44:45	(2,2,5)	2450.91	408,351	14635
10	1.8921 E+08	48:00§	(2,2,5)	10117.11	2,963,071	231035

* The solver stopped prematurely with “out of memory” error message.
§ The relief operations were not finished by the assumed horizon.
† On a 3.0 GHz Intel Pentium CPU with 2.0 GB RAM

Case-2 is similar to Case-1 but the only difference is that there are 200 trucks available in Case-2 versus 100 trucks in Case-1. Even though the number of vehicles was increased, the optimal solution was found in only 20 seconds. As it can be seen in Table 3, the size of the formulation (number of variables and constraints) for Case-2 is equal to Case-1 and this is one of the important advantages of current formulation. Since this formulation treats the vehicles as commodities, the number of available vehicles appears only as a right-hand-side parameter and does not have an effect on the problem size. Figure 4 shows the percent of unsatisfied demand over time for Case-2 at optimality. Since there were enough vehicles at the beginning, the vehicles did not need to return to the sources to pick up supplies once they had left. As a result, the delivery operations were completed after only 15 hours and 10 minutes.

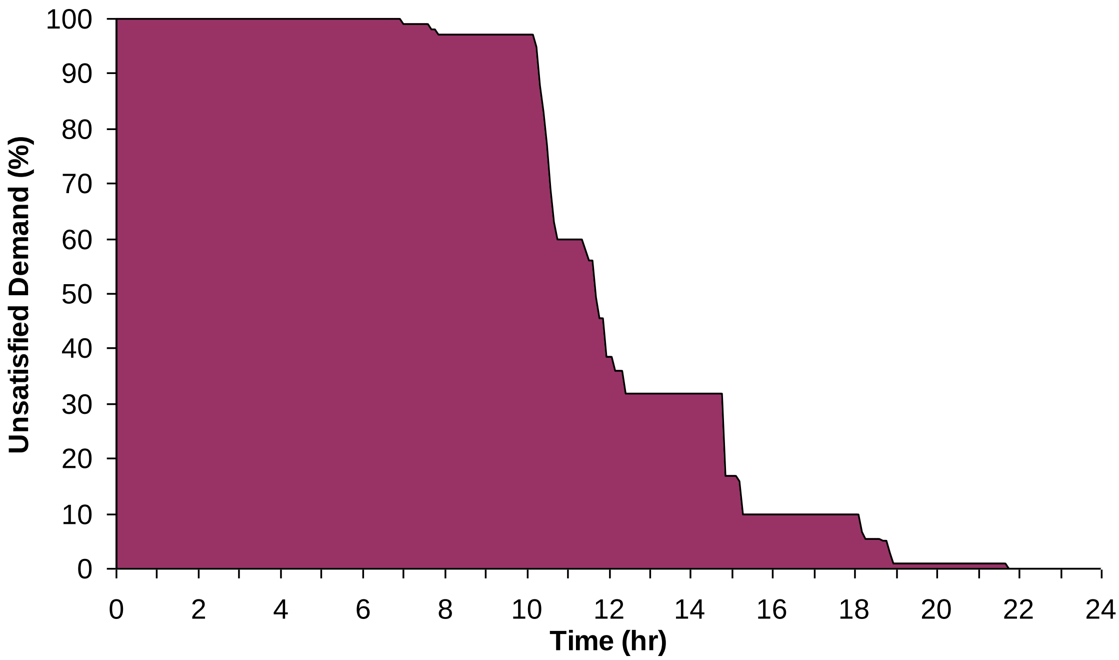

Figure 3. Percent of unsatisfied demand over time for CASE 1

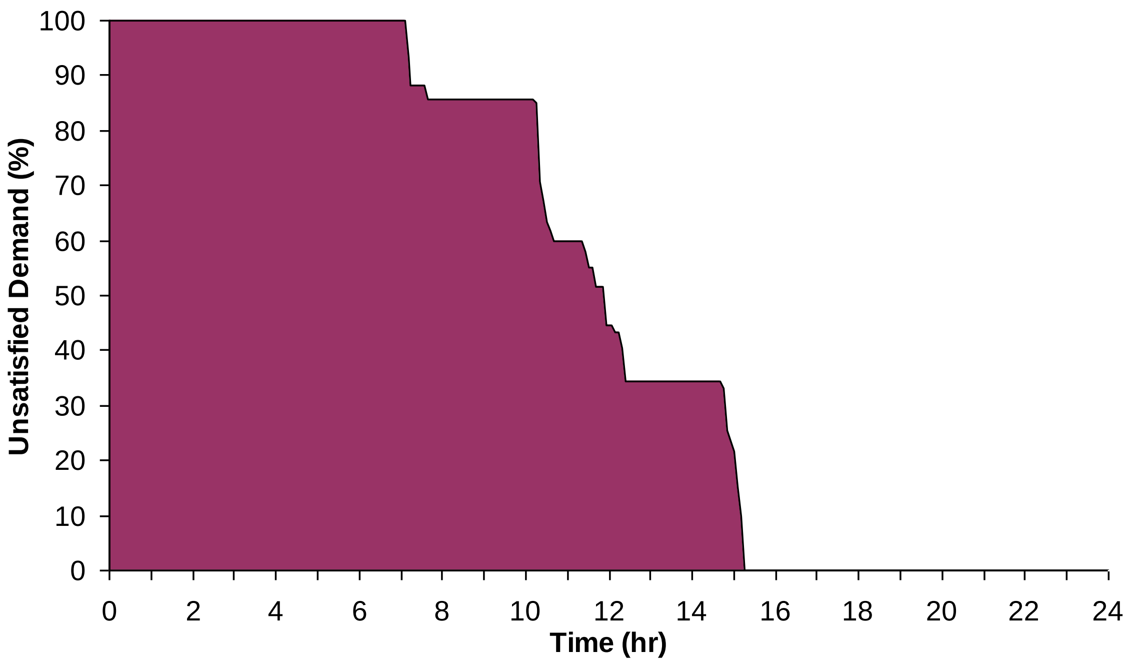

Figure 4. Percent of unsatisfied demand over time for CASE 2

Case 3 is similar to the base-case with the addition of loading, unloading, and storage capacities for all facilities. In this case, there is no limitation on the maximum number of temporary facilities and all the potential sites can be active. Figure 5 shows the variation of unsatisfied demand for Case-3. The addition of facility capacities prevented the shipment and delivery of large quantities of supplies. Instead, the relief commodities are delivered more uniformly over time compared to Case-1 and Figure 3. Consequently, the objective function value was higher and the operation took 22 hours and 5 minutes, 25 minutes more than Case-1. The running time was also increased to about 13 minutes to find the optimal solution.

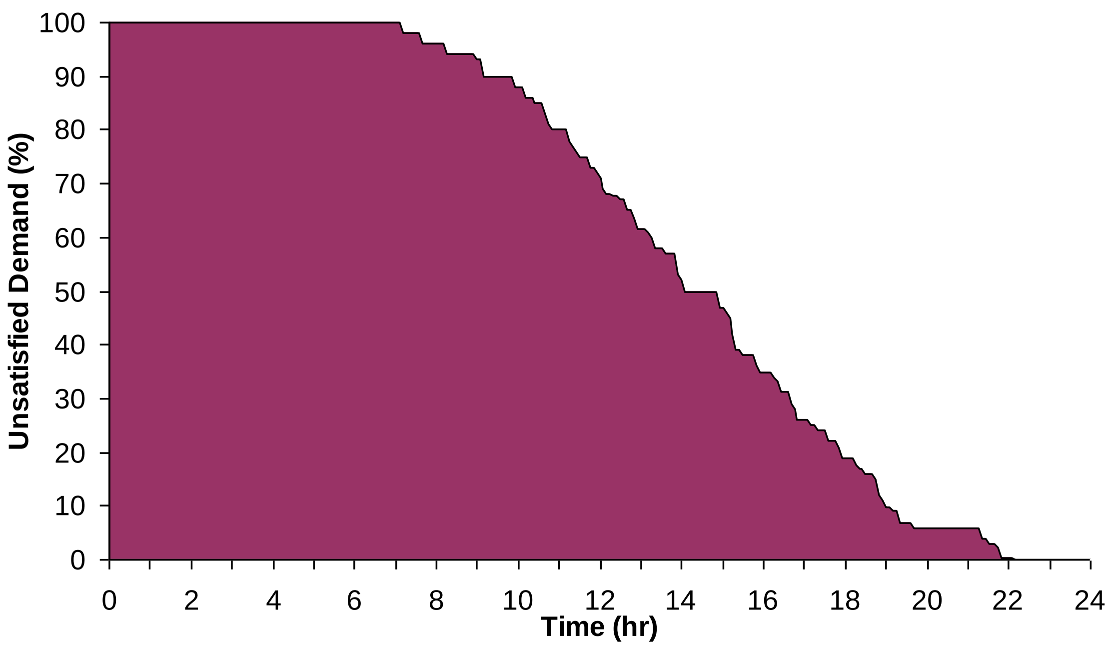

Figure 5. Percent of unsatisfied demand over time for CASE 3

Finally in Case-7, all constraints are considered. The constraints include conservation of flow for the commodities and vehicles, the linkage between commodities and vehicles and capacity of each vehicle, facility location with maximums of (2, 2, 5); loading, unloading and storage capacities for all facilities, and finally the 3 equity constraints (Equation 24, 25, 26). The full problem is very large and difficult problem. After around 23 hours of CPU time and more than 7.8 million iterations, CPLEX solver stopped and it could not find the optimal solution. However, the best integer solution found is very close to the best MIP bound (0.03% gap).

Another idea was to extend the relief operations duration from 1 day to 2 days and analyze its effect on the problem size and behavior. Case-8 through Case-10 was created to address that idea. As can be seen in Table 3, by extending the operations duration from 1 day to 2 days, the problem size rapidly grows. For example in Case-10, the CPLEX solver went over 2.9 million iterations and it took more than 2 days and 16 hours of CPU time to find the optimal solution. It is clear that a problem with complete set of constraints (if the equity constraints were to be added to the problem) with 2-days of operations, cannot be solved by the commercial solver.

1.6 SUMMARY OF CHAPTER 1

The global increase in the number of natural disasters highlights the need for a better planning and operation of the responding agencies. During emergencies various aid organizations often face significant problems of transporting large amounts of many different relief commodities from different points of origin to different destinations in the disaster areas. The transportation of supplies and relief personnel must be done quickly and efficiently to maximize the survival rate of the affected population and minimize the cost of such operations. It is very difficult, if not impossible, to efficiently operate such a complex system without comprehensive mathematical models.

Offering a centralized comprehensive model that describes the specifics of disaster supply chains was the main goal of this research. We aimed at developing a system of computer and mathematical models to keep track of operational details of large-scale disaster response operations and find the optimal allocation of scarce resources to the most critical tasks in order to minimize loss of life and human sufferings.

Initial investigations in this research showed that FEMA has a complex supply chain spreading across the country to coordinate with its state and local government counterparts and with nonprofit and for-profit organizations. To the best of our knowledge, there was no study in the academic literature that provided a systematic view of the FEMA’s supply chain. This research was able to investigate and summarize FEMA’s structure into seven main components and showed the relations between them as a network. The proposed network representation was the key factor that made the mathematical modeling of the FEMA’s special logistics structure possible.

The results of this research extended the state-of-the-art by presenting an integrated model at the operational level that describes the details of supply chain logistics in major emergency management agencies such as FEMA, in response to immediate aftermath of a large-scale disaster. The proposed model controls the flow of all the relief commodities from the sources through the chain and until they are delivered to the hands of recipients. The proposed model not only considers details such as vehicle routing and pick up or delivery schedules; but also considers finding the optimal location for temporary facilities as well as considering the capacity constraints for each facility and the transportation system. This model provided the opportunity for a centralized operation plan that can eliminate delays and assign the limited resources in a way that is optimal for the entire system.

Applying the proposed model on a series of case-study scenarios verified the model and showed its capabilities to handle large-scale problems. Using the proposed model provided high level of transparency and control over the disaster response operations that was not available before. For simpler cases, the commercial solver was able to find the optimal solutions, however, when the more difficult constraints such as Equity Constraints were added or when the time horizon was extended from 1-day to 2-days, CPLEX was unable to find the optimal solution within a reasonable CPU time.

It is concluded that better solution algorithms or heuristics are needed to address very large problem instances. For future steps, two main approaches are suggested to develop heuristic solution techniques. The first suggestion is to decompose the model and try to solve a number of smaller or easier problems and then aggregate the results. The structure of the model allows for spatial decomposition as well as temporal decomposition. In the second approach, the idea is to develop heuristics that can find near optimal solutions for the entire model. Various relaxation techniques may be used for this type of heuristics. Then the challenge is to find good feasible solutions and show whether they are within an acceptable range from a lower bound.

The result of this research enables central emergency management agencies such as FEMA to implement better practices in real-time disaster response at the operational level. However, high level of accuracy and control provided in this research can be effectively used toward emergency management at strategic and planning levels as well. To do so, a variety of potential disaster scenarios can be built and analyzed. Consequently, the planners can investigate the best potential locations for temporary facilities or the effect of different fleet size on the operation’s performance in various disaster scenarios. The questions about the best amounts and locations for preposition relief supplies can also be investigated.

CHAPTER 2: SOLUTION APPROACHES

In this chapter, initially some general integer programming solution approaches from previous studies in the literature are reviewed in section 2.1. Then in section 2.2, the solution approaches that specifically used in emergency logistics literature are reviewed. After literature review, in section 2.3 a number of solution techniques are proposed for the mathematical model of section 1.4. Two sets of algorithms are proposed to solve the different parts of the problem. In section 2.4 solution algorithms are proposed to solve the hierarchical location finding problem. Finally in section 2.5, some heuristic algorithms are proposed to solve the general integer vehicle routing problem.

2.1. GENERAL SOLUTION APPROACHES FOR INTEGER PROGRAMS

In General, integer programming problems are very difficult to solve. Over the years, different researchers have proposed several very different solution algorithms. Today, the question is how to select the best approach. Algorithm selection has become an art as some algorithms work better on some specific problem instances. A brief discussion of algorithms is presented in this subsection, attempting to expose readers to their characteristics. More detailed review of integer and combinatorial optimization algorithms can be found in the integer programming literature (e.g. Nemhauser and Wolsey (1999))

Historically, linear programming (LP) has been the base for integer programming (IP) solution approaches. LP was invented in the late 1940's. Those examining LP relatively quickly came to the realization that it would be desirable to solve problems which had some integer variables (Dantzig, 1960). This led to algorithms for the solution of pure IP problems. The first algorithms were cutting plane algorithms as developed by Dantzig, Fulkerson and Johnson (1954) and Gomory (1963). Land and Doig (1960) subsequently introduced the branch and bound algorithm. More recently, implicit enumeration (Balas 1965), decomposition (Benders 1962), lagrangian relaxation (Geoffrion, 1974) and heuristic approaches have been used to solve various integer programs.

McCarl and Spreen (1997) suggested the following classification of general algorithms for integer programming problems:

2.1.1 CUTTING PLANES

The first formal IP algorithms involved the concept of cutting planes. Cutting planes iteratively remove parts of the feasible region without removing integer solution points. The basic idea behind a cutting plane is that the optimal integer point is close to the optimal LP solution, but does not fall at the constraint intersection so additional constraints need to be imposed. Consequently, constraints are added to force the non-integer LP solution to be infeasible without eliminating any integer solutions. This is done by adding a constraint forcing the nonbasic variables to be greater than a small nonzero value. The simplest form of a cutting plane would be to require the sum of the nonbasic variables to be greater than or equal to the fractional part of one of the variables.

The cutting plane algorithms continually add such constraints until an integer solution is obtained. Methods for developing cuts appear in Gomory (1963) in more details.

Several points need to be made about cutting plane approaches. First, many cuts may be required to obtain an integer solution. For example, Beale (1977) reports that a large number of cuts is often required (in fact often more are required than can be computationally afforded). Second, the first integer solution found is the optimal solution. This solution is discovered after only enough cuts have been added to yield an integer solution. Consequently, if the solution algorithm runs out of time or space the modeler is left without an acceptable solution (this is often the case). Third, given comparative performance with other algorithms, cutting plane approaches have faded in popularity (Beale,1977).

2.1.2 BRANCH AND BOUND

The second solution approach developed was the branch and bound algorithm. Branch and bound, originally introduced by Land and Doig (1960), pursues a divide-and-conquer strategy. The algorithm starts with a LP solution and also imposes constraints to force the LP solution to become an integer solution similar to cutting planes. However, branch and bound constraints are upper and lower bounds on variables.

The branch and bound solution procedure generates two problems (branches) after each LP solution. Each problem excludes the unwanted noninteger solution, forming an increasingly more tightly constrained LP problem. There are several decisions required. One must both decide which variable to branch on and which problem to solve (branch to follow). When one solves a particular problem, one may find an integer solution. However, one cannot be sure it is optimal until all problems have been examined. Problems can be examined implicitly or explicitly. Maximization problems will exhibit declining objective function values whenever additional constraints are added. Consequently, given a feasible integer solution has been found, then any solution, integer or not, with a smaller objective function value cannot be optimal, nor can further branching on any problem below it yield a better solution than the incumbent (since the objective function will only decline). Thus, the best integer solution found at any stage of the algorithm provides a bound limiting the problems (branches) to be searched. The bound is continually updated as better integer solutions are found.

The problems generated at each stage differ from their parent problem only by the bounds on the integer variables. Thus, a LP algorithm which can handle bound changes can easily carry out the branch and bound calculations.

The branch and bound approach is the most commonly used general purpose IP solution Algorithm and it is implemented in many commercial solvers. However, its use can be expensive. The algorithm does yield intermediate solutions which are usable although not optimal. Often the branch and bound algorithm will come up with near optimal solutions quickly but will then spend a lot of time verifying optimality. Shadow prices from the algorithm can be misleading since they include shadow prices for the bounding constraints.

A specialized form of the branch and bound algorithm for zero-one programming was developed by Balas (1965). This algorithm is called implicit enumeration.

2.1.3 LAGRANGIAN RELAXATION

Lagrangian relaxation (Geoffrion (1974), Fisher (1981)) is another area of IP algorithmic development. Lagrangian relaxation refers to a procedure in which some of the constraints are relaxed into the objective function using an approach motivated by Lagrangian multipliers. The basic Lagrangian relaxation problem for the mixed integer program involves discovering a set of Lagrange multipliers for some constraints and relaxing that set of constraints into the objective function. The main idea is to remove difficult constraints from the problem so the integer programs are much easier to solve. IP problems with structures like that of the transportation problem can be directly solved with LP. The trick then is to choose the right constraints to relax and to develop values for the lagrangian multipliers leading to the appropriate solution.

Lagrangian Relaxation has been used in two settings: 1) to improve the performance of bounds on solutions; and 2) to develop solutions which can be adjusted directly or through heuristics so they are feasible in the overall problem (Fisher (1981)). An important Lagrangian relaxation result is that the relaxed problem provides an upper bound on the solution to the unrelaxed problem at any stage. Lagrangian relaxation has been heavily used in branch and bound algorithms to derive upper bounds for a problem to see whether further branching down on that branch is worthwhile.

2.1.4 BENDERS DECOMPOSITION

Benders Decomposition is another algorithm to solve integer programs. This algorithm solves mixed integer programs via structural exploitation. Benders (1962) developed the procedure which decomposes a mixed integer problem into two problems; an integer master problem and a linear subproblem. Then these problems are solved iteratively. Consider the following decomposable mixed IP problem:

$$
\begin{array}{llcll}
\text{Maximize} & FX & + & CZ & \\
\text{s.t.} & GX & & & \leq b_1 \\
 & HX & + & AZ & \leq b_2 \\
 & & & DZ & \leq b_3 \\
 & \multicolumn{2}{l}{X \text{ is integer,}} & Z \geq 0 &
\end{array}
$$

Assuming X* is a feasible set of points for integer variables X, then the subproblem for any given X* would be:

$$
\begin{array}{lll}
\text{Maximize} & CZ & \\
\text{s.t.} & AZ \leq b_2 - HX^* & (\alpha) \\
 & DZ \leq b_3 & (\beta) \\
 & Z \geq 0 &
\end{array}
$$

Solution to this subproblem yields the dual variables in parentheses. In turn a "master" problem is formed as follows:

$$
\begin{array}{lll}
\text{Maximize} & FX + Q & \\
X, \alpha, \beta, Q & & \\
\text{s.t.} & Q \leq \alpha_i (b_2 - HX) + \beta_i b_3 & \text{for } i = 1,2,3\ldots, p \\
 & GX \leq b_1 & \\
 & X \text{ is integer , } Q \text{ is unrestricted} &
\end{array}
$$

This problem contains the dual information from above and generates a new X value. The constraint involving Q gives a prediction of the subproblem objective function arising from the dual variables from the ith previous guess at X. In turn, this problem produces a new and better guess at X. Each iteration adds a constraint to the master problem. The objective function consists of FX + Q, where Q is an approximation of CZ. The master problem objective function therefore constitutes a monotonically nonincreasing upper bound as the iterations proceed. The subproblem objective function (CZ) at any iteration plus FX can be regarded as a lower bound. The lower bound does not increase monotonically. However, by choosing the larger of the current candidate lower bound and the incumbent lower bound, a monotonic nondecreasing sequence of bounds is formed. The upper and lower bounds then give a monotonically decreasing gap between the bounds. Benders decomposition convergence occurs when the difference between the bounds is driven to zero. When the problem is stopped with a tolerance, the objective function will be within the tolerance, but there is no relationship giving distance between the variable solutions found and the true optimal solutions for the variables.

Convergence will occur in a practical setting only if for every X a relevant set of dual variables is returned. This will only be the case if the subproblem is bounded and has a feasible solution for each X that the master problem yields. This may not be generally true. Also the boundedness and feasibility of the subproblem says nothing about the rate of convergence. The real art of utilizing Benders decomposition involves the recognition of appropriate problems and/or problem structures which will converge rapidly. The procedure can work very poorly for certain structures (Sherali 1981).

In general:

1. The decomposition method does not work well when the X variables chosen by the master problem do not yield a feasible subproblem. Thus, the more accurately the constraints in the master problem portray the conditions of the subproblem, the faster will be convergence.
2. The tighter (more constrained) the feasible region of the master problem the better.
3. When possible, constraints should be entered in the master problem precluding feasible yet unrealistic (suboptimal) solutions to the overall problem.

The most common reason to use Benders is to decompose large mixed integer problem into a small, difficult master problem and a larger simple linear program. This allows the solution of the problem by two pieces of software which individually would not be adequate for the overall problem. It should be noted that in Benders decomposition method, the master problem is still an integer program that might be very difficult to solve.

2.1.5 HEURISTICS

Many IP problems are combinatorial and difficult to solve by nature. In fact, the study of NP complete problems (Papadimitrou and Steiglitz (1982)) has shown extreme computational complexity for problems such as the traveling salesman problem. Such computational difficulties have led to a large number of heuristics. These heuristics are used when: a) the quality of the data does not merit the generation of exact optimal solutions; b) a simplified model has been used, and/or c) when a reliable exact method is not available, computationally attractive, and/or affordable.

Arguments for heuristics are also presented regarding improving the performance of an optimizer where a heuristic may be used to save time in a branch and bound code, or if the problem is repeatedly solved. Many IP heuristics have been developed, some of which are specific to particular types of problems. For example, there have been a number of traveling salesman problem heuristics as reviewed in Golden et al (1980). Zanakis and Evans (1981) provide a general review of heuristics.

Generally, heuristics perform well on special types of problems, quite often coming up with errors of smaller than two percent (McCarl and Spreen (1997)). Zanakis and Evans (1981) provide discussions of selections of heuristics vis-a-vis one another and optimizing methods.

2.1.6 STRUCTURAL EXPLOITATION

Past experiences on IP have indicated that general-purpose IP algorithms do not work satisfactorily for all IP problems. Recently, the most promising developments have involved structural exploitation, where the particular structure of a problem has been used in the development of the solution algorithm. Benders Decomposition and Lagrangian Relaxation are two examples of structural exploitation. Some problem reformulation approaches and also specialized branch and bound algorithms adapted to particular problems are examples of structural exploitation. The main mechanisms for structural exploitation are to develop an algorithm especially tuned to a particular problem or, more generally, to transform a problem into a simpler problem to solve. The application of such algorithms has sometimes led to spectacular results, with problems with thousands of variables being solved in seconds of computer time (McCarl and Spreen (1997)).

Unfortunately, none of the available algorithms have been shown to perform satisfactorily for all IP problems. However, certain types of algorithms are good at solving certain types of problems and a number of efforts have concentrated on algorithmic development for specially structured IP problems. The following section reviews some of approaches used in emergency logistics literature.

2.2. SOLUTION APPROACHES USED IN EMERGENCY LOGISTICS LITERATURE

section 1.2 provided an extensive review of previous research in the emergency logistics literature. From the number of researches discussed in section 1.2 only four publications are found to have a mathematical model that are partially similar to the mathematical model proposed in this research. In the following paragraphs the solution approaches used in these four publications are reviewed.

Haghani and Oh (1996) proposed a formulation and solution of a multi-commodity, multi-modal network flow model for disaster relief operations. Their model can determine detailed routing and scheduling plans for multiple transportation modes carrying various relief commodities from multiple supply points to demand points in the disaster area. They formulated the multi-depot mixed pickup and delivery vehicle routing problem with time windows as a special network flow problem over a time-space network. The objective was minimizing the sum of the vehicular flow costs, commodity flow costs, supply/demand storage costs and inter-modal transfer costs over all time periods. Structurally, their model was composed of two network flow problems; one with only real-valued variables and the other with integer variables were connected with a set of capacity constraints called linkage constraints.

They developed two heuristic solution algorithms; the first one was a Lagrangian relaxation approach, and the second was an iterative fix-and-run process. The first solution algorithm decomposes the model into two subproblems based on the relaxation of linkage constraints. Lagrangian relaxation is used with penalty for shortage of capacity for linkage constraints. The algorithm was iteratively applied until two subproblems converge. The second solution algorithm was an ad hoc method that fixed integer variables gradually. First all integer variables were relaxed and LP relaxation is solved. Then based on the LP solution, the values of some of the integer variables were fixed to an integer value and the LP was solved again. This process was repeated iteratively until all integer variables are fixed to integer values.

Haghani and Oh (1996) solved several instances of numerical problems with both algorithms. For smaller size problems, they showed both algorithms were successful in solving integer problem instances much faster than commercial solvers. They also showed for larger problem instances that the commercial solver was unable to find the optimal solution, both algorithms were able to find close to optimal solution in relatively short CPU times. Comparing the two algorithms, they concluded that the proposed fix-and-run algorithm outperforms the Lagrangian relaxation algorithm both in CPU time and final solution quality.

Barbarosoglu and Arda (2004) developed a two-stage stochastic programming model for transportation planning in disaster response. Their study expanded on the deterministic multi-commodity, multi-modal network flow problem of Haghani and Oh (1996) by including uncertainties in supply, route capacities, and demand requirements. The authors designed 8 earthquake scenarios to test their approach on real-world problem instances. Their model is a planning model that does not deal with details required at strategic or operational levels. The model does not address facility location problem or vehicle routing problem as well.

To solve numerical examples, Barbarosoglu and Arda (2004) in the first stage generate random scenarios for supply, demand, and available capacity. In the second stage they used the commercial solver GAMS to solve the resulted network flow problem to minimize the cost. They did not propose any special solution algorithms but used GAMS software to solve the numerical studies.

Ozdamar et al. (2004) addressed an emergency logistics problem for distributing multiple commodities from a number of supply centers to distribution centers near the affected areas. They formulated a multi-period multi-commodity network flow model to determine pickup and delivery schedules for vehicles as well as the quantities of loads delivered on these routes, with the objective of minimizing the amount of unsatisfied demand over time. The structure of the proposed formulation enabled them to regenerate plans based on changing demand, supply quantities, and fleet size. They developed an iterative Lagrangian relaxation algorithm and a greedy heuristic to solve the problem.

The Lagrangian relaxation approach used in Ozdamar et al. (2004) was similar to the one previously discussed in Haghani and Oh (1996) with the only change that Ozdamar et al. (2004) used commercial solver GAMS to solve the linear relaxations. The proposed greedy algorithm solves the network flow problem without considering vehicles to find the best routes for the flow of commodities. Then the algorithm assigns the vehicles to the first available shipment so to minimize the shipment delay. If the vehicles are not available immediately, the shipment is postponed till the earliest available vehicle arrives.

The greedy approach is myopic in the sense that the vehicles are independently assigned to the first available job instead of considering the other combinations that might be more rewarding. Comparing the Lagrangian relaxation algorithm and the Greedy algorithm in Ozdamar et al. (2004), it was concluded that the greedy algorithm performs faster than Lagrangean relaxation algorithm. However, the greedy algorithm usually resulted larger gaps with global optimal compared to the Lagrangian relaxation. Greedy algorithm did not perform good especially when the capacity was tight which is usually the case in disaster response operations.

Finally, Yi and Ozdamar (2007) proposed a model that integrated the supply delivery with evacuation of wounded people in disaster response activities. They considered establishment of temporary emergency facilities in disaster area to serve the medical needs of victims immediately after disaster. They used the capacity of vehicles to move wounded people as well as relief commodities. Their model considered vehicle routing problem in conjunction with facility location problem. The proposed model is a mixed integer multi-commodity network flow model that treats vehicles as integer commodity flows rather than binary variables.

Their numerical experiment considered a potential earthquake scenario for the city of Istanbul in Turkey. The numerical problem had 20 nodes, 3 transportation modes, 2 relief commodities and modeled for 8 time periods. They used commercial solver CPLEX 7.5 to solve the IP model. They did not propose any new solution algorithm to the problem however they offers an algorithm to find the itinerary of vehicles from the optimal solution output of CPLEX integer programming solver. They reported that post processing algorithm was pseudo-polynomial in terms of the number of vehicles utilized.

Yi and Ozdamar (2007) took the network flow vehicle routing (where vehicles are treated as general integer-valued commodities) and compared it with classic 0-1 vehicle routing. They showed that the general integer formulation is more compact and it is much more efficient for solving. They experienced CPU times “in seconds” for general integer VRP versus “in minutes” for classic binary VRP. However in general integer VRP, post processing was needed to extract detailed vehicle routing and pickup or delivery schedules.

To summarize, it is shown that in previous publications only a few mathematical models can be found which have relatively similar structures to the model proposed in this research. In those publication, three solution approaches are proposed and tested; Lagrangian Relaxation, Fix-and-Run Heuristic, and Greedy Heuristic algorithm. Lagrangian relaxation is successful in proving a bound but it was shown to be the most time consuming algorithm. Greedy Heuristic algorithm was shown to be faster compared to Lagrangian relaxation algorithm. However, it lacked in the quality of final optimal solution and resulted in large optimality gaps especially when transportation capacity was limited. Fix-and-Run heuristic outperformed Lagrangian Relaxation in both categories of speed and solution quality. Fix-and-Run heuristic compared to Lagrangian relaxation found the final solution in less CPU time and resulted in smaller optimality gap.

2.3. SOLUTION TECHNIQUES FOR PROPOSED MATHEMATICAL MODEL

The mathematical model proposed in section 1.4 is a complex integrated model. Such an integrated model provides the opportunity for a centralized operation plan that can eliminate delays and assign the limited resources to the best possible use. However, the model is a large-scale mixed general integer programming model and solving such a comprehensive mathematical model is a big challenge. As it is shown in numerical experiments in section 1.5, the commercial solver was unable to find the optimal solution in a reasonable time.

Based on the analysis of solution techniques for similar models in the literature, it is concluded that exact solution algorithms will not be able to solve the model. Consequently, the best approach might be designing fast heuristic algorithms that can find near optimal solutions in relatively short computation times. On the other hand, since this model is more complicated than all the previous works in the literature, it would be favorable to structurally decompose this problem to some smaller or easier problems.

This model integrates commodity flow problem which is a linear multi-commodity network flow problem with multi-echelon facility location problem which is a binary mixed integer program, and multimodal vehicle routing problem which is a large-scale general integer-valued network flow problem. The Idea is to decompose this problem into smaller or easier problems while taking advantage of special structures that already exist.

The multi-commodity network flow problem is a linear program. LP models are considered easy-to-solve since efficient solution algorithms and commercial solvers exist that can quickly solve large-scale linear programs. The difficult parts are the two integer programming subproblems. In the following sections, a number of heuristic algorithms are proposed to solve the integer programming part of mathematical model. First in section 2.4, four heuristics are proposed to solve the hierarchical location finding problem. Then in section 2.5, four new heuristic algorithms are proposed to solve the general integer vehicle routing problem.

2.4 ALGORITHMS FOR SOLVING LOCATION PROBLEM

As discussed earlier, the mathematical formulation presented in section 1.4 is composed of three subproblems. The linear commodity flow subproblem is considered easy and can be solved in conjunction to the facility location problem. On the other hand, the general integer vehicle routing subproblem is a large-scale mixed integer program itself which is considered very difficult to solve.

This problem is not mathematically decomposable and it is important to keep the interrelations between the three subproblems. To do so, it is suggested to first relax the integrality condition of vehicle routing subproblem and try to solve the location problem. When the optimal locations are known, it would be much easier to solve the vehicle routing problem. Considering relaxed VRP problem inside the location finding problem is a big advantage because it is easier to solve meanwhile it still reflects the effects of the VRP and available transportation capacity on the location finding problem. The mathematical formulation of this location problem can be obtained by only relaxing the Y_{ijt}^{m} variables (general integer variables related to vehicle routing problem) in the original model presented in section 1.4.

In the following subsections, four solution approaches are proposed to solve the location finding problem.

2.4.1 EXPLICIT ENUMERATION

The candidate sites for temporary facility locations are chosen prior to emergency response. Consequently, the number of potential sites is known and the number of possible combinations for facility locations is a finite number. The simplest conceivable optimization approach is explicit enumeration. It is possible to generate all possible solutions, evaluate each of them, and keep the best.

To test the applicability of explicit enumeration, let's use the numerical example introduced in section 1.5:

Combinations for selecting 2 MOB out of 4 candidates: $\frac{4!}{2!\,2!}=6$

Combinations for selecting 2 FOSA out of 4 candidates: $\frac{4!}{2!\,2!}=6$

Combinations for selecting 4 SSA out of 10 candidates: $\frac{10!}{4!\,6!}=210$

Total number of combinations is equal to $6\times 6\times 210=7560$. For any given locations, the remaining problem is a linear program that has a network structure. Linear network problems are considered easy to solve since good algorithms and efficient commercial solvers are developed to solve that problem. For instance, for linear relaxation of the numerical experiment introduced in section 1.5 with given locations, CPLEX solver was able to solve the problem in around 7 seconds on average. If it is required to enumerate all combinations, the total CPU time is equal to $7560\times 7\sec = 52920\sec = 14.7 hours$.

It can be concluded that since it is easy to solve the problem after locations are given, it is still possible to explicitly enumerate all combinations and find the final optimal solution. It might not be wise to solve for every single combination, however, it indicates the level of difficulty of the IP problem and provides a benchmark for development and comparison of other solution algorithms. Some other heuristic algorithms are introduced in the following subsections.

2.4.2 BRANCH AND BOUND - HIERARCHICAL DECOMPOSITION

Branch and Bound algorithm is widely used to solve integer programs. It is especially successful when the integer variables are 0-1 binary variables as it is the case in location finding problems. Good algorithms and efficient commercial solvers are developed that use the branch and bound technique. CPLEX solver is a commercial solver that can apply Branch and Bound to solve binary mixed integer programs.

The proposed mathematical model contains three levels of temporary facilities. Mobilization Centers (MOB) are at the top. Federal Operational Staging Areas (FOSA) are the intermediate level facilities and receive commodities from MOB. Then there are State Staging Areas (SSA) that receive commodities from FOSAs. It is possible to use branch and bound to solve all three levels simultaneously. However, it is possible to hierarchically decompose the facility location problems and solve them consecutively.

Three decomposition approaches are proposed and tested:

1. Top then Bottom: Decompose the problem into federal level facilities and state level facilities. Assume all state level facilities are open (i.e. $Loc_i = 1 \quad \forall i \in SSA$). Solve the integer program to find the optimal locations for federal level facilities. Fix the solution for top level and solve the integer program for the state level facilities.
2. Bottom then Top: Decompose the problem into federal level facilities and state level facilities. Assume all federal level facilities are open (i.e. $Loc_i = 1 \quad \forall i \in FOSA \cup MOB$). Solve the integer program to find the optimal locations for state level facilities. Fix the solution for bottom level and solve the integer program to find the optimal state level facilities.
3. Tier by Tier: First solve the integer program to find the optimal locations for MOB level facilities assuming all other facilities are open. Then fix the optimal MOB, assume all SSA are open and solve IP for FOSA facilities. Finally, fix optimal MOB and FOSA then solve for SSA.

Table 2.1 shows the results of applying the abovementioned approaches to the numerical problem in section 1.5. Comparing the total CPU times, it can be seen that Tier by Tier decomposition resulted in the least computation time. It was able to reduce the CPU time from 379 seconds when all tiers are considered together, to about 203 seconds (a reduction of about 46%). The Top-then-Bottom approach also gives good results with a total of 215 seconds computation time (43% reduction). On the other hand, it seems that for the current example, Bottom-then-Top approach did not provide favorable results. Mainly, when all federal level facilities are forced to be open, it forces an unnecessarily large number of combinations. Exploring all those combinations resulted in higher than usual computation times in Bottom-then-Top approach.

Table 2.1- Branch and Bound and Hierarchical Decomposition

Case	Solution Time	Final Obj	Iterations	Total Time (S)
Solve for All Location Tiers	378.69	3.83595 E+7	204402	378.69
All State Level = 1, Solve for FED level	191.91	3.83595 E+7	181589	214.83
Given FED level, SOLVE for State	22.92	3.83595 E+7	43781	
ALL FED level = 1, Solve for State	819.23	3.77795 E+7	559223	958.2
Given State, Solve for FED	138.97	3.83595 E+7	106213	
Solve for MOB, Rest = 1	151.03	3.82113 E+7	139943	202.63
Given MOB, Solve for FOSA, SSA =1	28.66	3.83595 E+7	59960	
Given MOB & FOSA, Solve for SSA	22.94	3.83595 E+7	43781	

Solution times are for solver CPLEX 11.0 on a machine with 3GHz CPU and 4GB RAM

It is important to mention that all three proposed approaches provided the same optimal solution. Even though it is not a proof, it is a very favorable property to have a number of heuristics algorithms than find the exact solution. The design of proposed hierarchical decompositions allowed the heuristics to find the exact optimal solutions by not cutting the feasible region. For example in Tier-by-Tier approach, when solving for top tier (MOB level), all other lower level facilities are forced to be open regardless of the limitation on the maximum number of open facilities in lower levels. This provides the chance to find the optimal locations for the tier in hand because all lower levels facilities are at their best theoretical combinations.

2.4.3 HIGHEST CAPACITY RATIO

Solving linear relaxation of integer programming problems and analyzing the results can reveal very valuable insights. The idea in this heuristic is to use the linear relaxation to find the facility or facilities that are most important for the performance of the system. Returning to capacity constraints in the mathematical formulation in section 1.4, the following equation enforces the sum of all flows leaving facility i, to be less than Loading Capacity of facility i if it is selected to be open; or to be zero otherwise:

$$\sum_{c}\sum_{j} X_{ijt}^{cm} \leq Lcap_{it}^{m} \times Loc_{i} \qquad \forall i, m, t$$

If the binary integer variable Loc_i is relaxed to take any real number between 0 and 1, it can show the capacity ratio that is used in facility i. The facilities with higher capacity ratios are more favorable because they handle the most flow and their existence is more important to the entire response operations.

The steps of Highest Capacity Ratio (HCR) Algorithm are:

Step 1- Relax the integrality condition for all temporary facility variables

Step 2- Add $0 \le Loc_i \le 1$ for all relaxed binary variables

Step 3- Solve the Linear Relaxation problem and obtain optimal values for all Loc_i variables

Step 4- For each facility type; sort the Loc_i variables in descending order

Step 5- For each facility type; select the maximum number of facilities allowed from top of the list

By following the above steps, one can find the selected facilities in a single snapshot. However, it can be argued that selecting a facility may affect the selection of others. So it might be beneficial to select the one facility with the highest ratio, solve the linear relaxation again, and repeat until maximum number of each facility is selected.

The steps of Iterative Highest Capacity Ratio (IHCR) algorithm are:

Step 1- Relax the integrality condition for all temporary facility variables

Step 2- Add $0 \le Loc_i \le 1$ for all relaxed binary variables

Step 3- Solve the Linear Relaxation problem and obtain optimal values for all Loc_i variables

Step 4- Find the facility i with the highest Loc_i value

Step 5- If the maximum number of facilities are not reached, select Facility i, add $Loc_i = 1$ to the formulation and go to Step 3. Otherwise stop.

To test HCR algorithm, the LP relaxation of the numerical experiment formerly introduced in section 1.5 is solved again. Table 2.2 shows the values for relaxed Loc_i variables. The constraints on the maximum number of facilities required the selection of 2 MOB out of 4, 2 FOSA out of 4 and 4 SSA out of 10 potential SSA nodes. Solving the linear relaxation with additional $0 \le Loc_i \le 1$ constraints only took about 32 seconds. The resulted node selection shown in Table 2.2 is obtained with using the single snapshot HCR algorithm. It is worth mentioning that for this example, the HCR algorithm was able to find the exact optimal solution and did so incredibly faster than Branch and Bound method (32 seconds versus 379 seconds).

Table 2.2 Values of Loc_i variables for HCR Heuristic Algorithm

Facility Type	**MOB**	**FOSA**	**SSA**	
Loc_i Value	0.6807	1.0	1.0	0.6174
	1.0	0	0.5357	0.3131
	0	0.8532	0.4164	0.8359
	0.3193	0.1468	0	0.2054
	-	-	0	0.0762
Selected Nodes	**4 , 5**	**8 , 10**	**12 , 13 , 17 , 19**	

2.4.4 STATIC NETWORK-LOCATION

Considering a time varying structure and a time-space network is essential to capture the details of emergency response logistics at the operational level. However, it expands the size of the formulation drastically and makes the problem extremely difficult to solve. The idea for this heuristic is to build a static version of the formulation that can be solved much easier and faster. It should still consider the special structure of the network and account for supplies, demands, and facility capacities; but manage to aggregate over the time dimension in order to generate a smaller formulation. To do so, the following mathematical formulation is proposed:

$$Min \quad \sum_m \sum_c \sum_{i,j} t_{ij}^m . X_{ij}^{cm} \tag{2.1}$$

$$\sum_m \sum_j X_{ij}^{cm} - \sum_m \sum_j X_{ji}^{cm} \le Sup_i^c \qquad \forall i \in U, c \tag{2.2}$$

$$\sum_m \sum_j X_{ij}^{cm} - \sum_m \sum_j X_{ji}^{cm} = 0 \qquad \forall i \in W, c \tag{2.3}$$

$$\sum_m \sum_j X_{ij}^{cm} - \sum_m \sum_j X_{ji}^{cm} = Dem_i^c \qquad \forall i \in V, c \tag{2.4}$$

$$\sum_c \sum_j X_{ij}^{cm} \le Lcap_i^m \qquad \forall i \in U, m \tag{2.5}$$

$$\sum_c \sum_j X_{ji}^{cm} \le Ucap_i^m \qquad \forall i \in U + V, m \tag{2.6}$$

$$\sum_c \sum_j X_{ij}^{cm} \le Lcap_i^m.Loc_i \qquad \forall i \in W, m \tag{2.7}$$

$$\sum_c \sum_j X_{ji}^{cm} \le Ucap_i^m.Loc_i \qquad \forall i \in W, m \tag{2.8}$$

$$\sum_i Loc_i \le Loc_{\max} \tag{2.9}$$

$$Loc_i \in (0,1) \qquad \forall i \in W \qquad \text{and} \qquad X_{ij}^{cm} \ge 0 \qquad \forall i, j, c, m$$

The notations are similar to the original problem that is previously defined in section 3.5 with the exception that time index t is dropped from all variables and parameters. As a result, all variables and parameters are static and defined as the aggregate value of the original variables over all time periods. For example, Sup_i^c and Dem_i^c are the aggregate supply and aggregate demand of commodity c in node i, over the entire planning horizon. Decision variable X_{ij}^{cm} is the aggregate amount of commodity c that is shipped from node i to node j with transportation mode m, over the entire planning horizon.

In this formulation the details of unsatisfied demand over time is not available. Consequently, the objective function (2.1) is chosen to minimize the total travel time by all commodities. Equation (2.2) and (2.4) enforce the supply and demand constraints for each node and each commodity. Equation (2.3) imposes the conservation of flow at intermediate nodes. Loading and unloading capacity constraints are defined in equations (2.5) and (2.6) for the permanent facilities. Similar constraints for temporary facilities are required by equations (2.7) and (2.8). Finally, equation (2.9) enforces the maximum number of open facilities for each facility type.

In the proposed static formulation, vehicle routing constraints are dropped from the formulation. It is equivalent to assume that ample transportation capacity is available or the initial distribution of vehicles is done in such a way that does not affect the choice of temporary facilities. Also, time-space structure is removed from the original model. It can be explained if the variations of supplies, demands, and capacities over the planning horizon are not very drastic. No link capacity is imposed in this formulation; however capacity limitations are reflected in loading and unloading capacities for each facility.

It should be noticed that the static formulation is still an integer programming model. However, it is of much lower size and complexity compared to the original formulation while still reflecting the structure and important properties of the original model.

Similar to previous heuristics, Static Network Location Problem (SNLP) heuristic is also tested with the numerical example of section 1.5. CPLEX solver version 11.0 is used to solve the problem on a 3 GHz Dell desktop computer with 4GB of RAM. After presolve modifications, reduced MIP had 130 rows, 491 columns, and 1713 nonzeros. It took only 0.1 Seconds and 410 iterations to solve the modified problem which is extremely faster than the previous heuristics. However, optimal locations obtained from this formulation do not match with the optimal locations of the original IP problem. Using the locations suggested by SNLP results in 2.5% higher objective function value compared to the case that exact optimal locations are used.

To summarize, four heuristic approaches are proposed to solve the location finding problem. Computation times vary greatly across the algorithms ranging from 14 hours to 0.1 seconds. Firstly, Explicit Enumeration showed that even though LP solution when locations are given takes only 7 seconds, the large number of possible combinations makes it very difficult to explore all the combinations. Secondly, Hierarchical decomposition approach suggested that it is beneficial to choose it over the general branch and bound (46% faster). Among the three suggested Hierarchical decompositions, Tier-by-Tier decomposition was the fastest. Thirdly, Highest Capacity Ratio heuristic was the fastest among all other heuristics that could still find the exact optimal solution. And finally, SNLP proposed a new formulation that is very efficient and can be solved to find the locations for the original problem. SNLP was the fastest algorithm but the resulted locations were different from those of the exact optimal solution.

2.5 ALGORITHMS FOR SOLVING VEHICLE ROUTING PROBLEM

In section 2.2 the relevant literature that suggested solution methods was summarized. Mainly, three heuristic approaches were proposed to solve the general integer vehicle routing problem: Lagrangian Relaxation, Fix-and-Run Algorithm, and a Greedy Algorithm. Using the numerical results, it was also concluded that the Fix-and-Run algorithm proposed by Haghani and Oh (1996) had the performance. It was the fastest algorithm and it had the least optimality gap compared to the other algorithms.

In the following subsections, four heuristic algorithms are proposed to solve the general integer vehicle routing problem. The general idea is adopted from the successful experience of Fix-and-Run heuristic algorithm suggested by Haghani and Oh (1996). The main steps of the proposed algorithms are:

1. The mixed integer linear problem is solved with the relaxation of integer variables.
2. The values of some integer variables are fixed in an orderly manner and the problem is solved again with the relaxation of the remaining integer variables iteratively.
3. When all integer variables are fixed, the process is terminated.

2.5.1 T-Counter Heuristic

The steps of T-Counter algorithm are:

Step 1- Relax all general integer variables and solve the relaxed LP. Set t=0

Step 2- Check all Y_{ijt}^{m} variables for current time period t. If all Y_{ijt}^{m} variables are integer, then if t = t_{last}, terminate. Otherwise, set t = t + 1 and restart Step 2.

Step 3- For current time period t, fix all Y_{ijt}^{m} variables to the closest integer number

Step 4- Create a new problem by adding (Y_{ijt}^{m} = the fixed value from step 3) constraints to the problem

Step 4- Relax the rest of the integer variables and solve the new LP problem

Step 5- Set t = t + 1 and go to Step 2

In this algorithm, starting from the first time period, Y variables are fixed iteratively and in a chronological order. If the flow of the vehicles through the network is fixed to be integral at time period t, because of the network structure of the problem, it is more likely that the flows at time periods after t, also turn out to be integral. Conservation of flow in a time-space network requires that if the flows that enter a node are integer, then some of the flows that leave the same node must also be integral. This does not mean that every single flow leaving that node will definitely be integer but it is a necessary condition.

If the planning horizon of the problem is consisted of T time periods, then at the worst case the algorithm will go through only T iterations. It is the worst case scenario and not the average case because during an iteration if all Y variables are already integer, the algorithm directly proceeds to next t without solving a LP relaxation. This is a very important property to have because this algorithm will stop at most after T iterations. Fast convergence rate is expected from this algorithm.

2.5.2 Origin-Based T-Counter Heuristic

In the previous algorithm, in each iteration all the Y_{ijt}^{m} variables for current time period t are fixed at the same time. That approach reduces the flexibility of the algorithm to reroute the vehicles within one time period which can sometimes cause suboptimal assignments. To remedy this, Origin-based T-Counter heuristic algorithms is proposed. In this algorithm, outgoing flows from only one origin node will be fixed during each iteration. In other words, for current time period t, we start from node i = 1 and fix all outgoing Y_{ijt}^{m} variables, solve LP relaxation, then fix all flows from node i = 2, and move to the next node until all nodes are fixed. Then the same procedure is followed for the next time periods until the end of the planning horizon.

The steps of Origin-based T-Counter algorithm are:

1- Set t = 0 and i=1
2- Relax all general integer variables and solve the relaxed LP.
3- If all relaxed variables are integer, the IP solution is found, Terminate

4- For current t and i, fix all Y_{ijt}^{m} variables to the closest integer number

5- Create a new problem by adding (Y_{ijt}^{m} = the fixed value from step 4) constraints

6- If i < i_{last} then set i = i + 1 and go to step 2. Otherwise go to the next step

7- If t = t_{last} terminate otherwise set i = 1 and set t = t +1, go to step 2

This algorithm is more general compared to T-Counter algorithm. If the planning horizon of the problem is consisted of T time periods and N is the number of nodes in the network, then at most T × N iterations are required to solve the problem. Again this is a worst case scenario and in general the algorithm is expected to find the integer solution before going through all T × N iterations.

2.5.3 Y-List Heuristic

In the previous two algorithms, several Y variables are fixed during each iteration. For example in T-Counter algorithm, at first iteration all Y_{ijt}^{m} variables with t = 0 are fixed simultaneously that can lead to under utilization of the available vehicles. For more clarification assume a hypothetical scenario where there are 4 vehicles available at node i and 3 exactly similar arcs are leaving node i. Solving the linear relaxation of the problem will assign 1.33 vehicles to each path. Applying T-Counter algorithm or even Origin-based T-Counter algorithm to this example rounds down 1.33 and as a results it assignments 1 vehicle to each path and 1 vehicle will remain unused.

The idea of Y-List algorithm is to solve this problem by only fixing one Y variable in each iteration. This will allow the LP model to adjust itself and take advantage of any potential vehicles that might be available and are not being used due to rounding down. Returning to our hypothetical scenario, if the 3 arcs are fixed one by one then all available vehicles will be used. The vehicle assignment will be 1, 1, 2 and all 4 vehicles are utilized.

To run this algorithm, it is required to have a priority list of all Y variables. When the first LP relaxation is solved, the algorithm needs to select a Y variable among all non-integer Y variables to fix. It is faster to have a pre-populated list of all Y variables and then fix them one by one if they have a non-integer value. The steps of Y-List algorithm are:

1- Populate a sorted list of all Y_{ijt}^{m} variables
2- Relax all general integer variables and solve the relaxed LP
3- If all Y_{ijt}^{m} variables are integer, save the solution & terminate the algorithm, otherwise
4- Select the 1st Y_{ijt}^{m} from the list, Fix it to the closest integer number and remove it from the Y-list
5- Create a new problem by adding (Y_{ijt}^{m} = the fixed value from step 4) constraint
6- Go to step 2

Theoretically in the worst case scenario, the algorithm can go through $|Y|$ iterations. $|Y|$ is the total number of all Y_{ijt}^{m} variables and also the size of the Y-List set. In large scale numerical problems, $|Y|$ can be a very large number. For example in the numerical experiment in section 1.5, thousands of Y_{ijt}^{m} variables exist. In the worst case scenario the algorithm need to go over thousands of iterations and fix every single Y variable. However, as it will be shown, usually the algorithm does not need to fix every single Y variable before finding an IP solution. In fact due to having a network structure, an IP solution is found very quickly and the algorithm converges relatively fast in typical numerical examples.

2.5.4 Y-List Modal Heuristic

In large-scale logistic operations often multiple transportation modes are utilized. From theoretical perspective, each transportation mode can be considered as the flow of special commodity over the network. Different transportation modes are not competing for share resources and there is no explicit constraint that relates the flow of different modes. Consequently, it can be assumed that each transportation mode is acting somehow independently. It should be mentioned that relief commodities that are carried by each transportation mode can be transferred to another mode inside intermodal terminal but the vehicles of each mode are never interchangeable. For example, if 2 trucks and 2 helicopters enter a node, always the same 2 trucks and 2 Helicopters have to leave that node and it can never transform into 3 trucks and 1 helicopter.

Taking advantage of this independence among multiple transportation modes is the idea behind Y-List Modal heuristic. Y-List Modal is very similar to previously described Y-List heuristic, however it tries to fix a Y variable from each transportation mode during any iteration. For example, if two transportation modes exist, the algorithm will fix two Y variables in each iteration. Consequently, if $|M|$ is the number of available transportation modes, the algorithm will fix $|M|$ variables in each iteration and it can stop after $|Y|/|M|$ iterations in the worst case.

The steps of Y-List Modal algorithm are:

1- Populate a sorted list of all Y_{ijt}^{m} variables for each mode m
2- Relax all general integer variables and solve the relaxed LP
3- If all Y_{ijt}^{m} variables are integer, save the solution & terminate the algorithm, otherwise
4- For each mode m, Select the 1st Y_{ijt}^{m} from the list, Fix it to the closest integer number and remove it from it's Y-list
5- Create a new problem by adding (Y_{ijt}^{m} = the fixed value from step 4) constraints
6- Go to step 2

2.5.5 Comparing Performance of the Proposed Algorithms

In previous sections, four heuristic algorithms are proposed to solve the general integer vehicle routing problem. In this section, these algorithms are analyzed and their performance is compared. All four algorithms are applied to a similar numerical example that is previously defined in section 1.5. The facility location problem is solved in previous step and the optimal locations of the facilities are assumed to be known at this stage.

The mathematical model is generated and initially solved by CPLEX Sofware. Table 2.3 represents the statistics of the mathematical model and also the optimization results obtained by the commercial solver. It is shown that the problem is a large-scale mixed integer program with a large number of general integer variables. CPLEX version 11.0 is used on a Dell desktop computer with 3 GHz Intel CPU and 4 GB of RAM. As it can be seen in the table, the software got to as close as 0.5 percent gap but it was unable to find the exact solution for the problem even after a long computation time. 0.5 percent optimality gap should be acceptable in many applications, nonetheless it shows the difficulty of solving the MIP problem even with a strong commercial solver on a fast computer.

Table 2.3- Statistics and Optimization Results from CPLEX Solver

Problem Stats					
Objective nonzeros = 3881					
Variables : 110572 [Nonnegative : 48300, Binary : 18, General Integer : 62254]					
Linear Constraints : 36593 [Equality : 11960 , Non-equality : 24633]					
Nonzeros : 372305 [RHS nonzero : 1467]					
CPLEX Optimization Results					
Objective Value	Solution Time (s)	GAP (%)	Initial LP Bound	MIP Best Bound	Comments
3.8709 E+7	81000	0.51	3.8059E7	3.8511E7	Program Stopped by User after 22.5 hrs

Table 2.4 show the results of solving the same problem using the four heuristic algorithms proposed in this chapter to solve general integer vehicle routing problem. Comparing gap percentiles from the best IP, it is concluded that the proposed algorithms were generally successful. Three of the four proposed heuristic algorithms provided very small optimality gaps of between 1 and 2.5 percent to the best IP solution provided by the commercial software after 22.5 hours. Comparing the solution times is even more impressive. It can be seen that all algorithms found an IP solution and all of them converged in less than about 4 minutes. It is very important to quickly find close to optimal solution especially in this problem that deals with dynamic emergency response operations.

Table 2.4 – Numerical Results of proposed VRP Heuristic Algorithms

ALGORITHM	Objective Value (E+7)	% GAP (Initial LP)	%GAP (Best IP)	Iterations	Solution Time (s)
T-Counter	4.2525	10.85	9.85	98	113.3
Origin-Based T-Counter	3.9668	3.41	2.47	3977	247.1
Y-List	3.91615	2.09	1.16	851	89.1
Y-List Modal	3.9300	2.45	1.52	507	73.7

Comparing the four algorithms, the Y-List algorithm is shown to find the best solution quality with the minimum gap. Y-List Modal and Origin-Based T-Counter algorithms also resulted in very good objective functions and small optimality gap. T-Counter algorithm has the largest gap of about 10 percent. It should be reminded that the idea for T-Counter algorithm was adopted from Haghani and Oh (1996) which was the best practice available to this date, to the best of our knowledge.

Comparing the solution speed and rate of convergence, it can be seen that all algorithms are quite fast. Y-List Modal was the fastest algorithm with only 73.7 seconds CPU time. Y-List and T-Counter algorithms are in 2nd and 3rd place with relatively close solution times. Y-List Modal produced the longest solution time of about 4 minutes, mainly due to the large number of iterations that was required. It is very important to notice that the number of iterations is not directly related to the solution time, because different iterations take different CPU times. For example, Origin-Based T-Counter goes through about 4000 iterations in about 4 minutes compared to about 100 iterations of T-Counter that takes about 2 minutes. Also, Y-List Modal that recorded the least solution time, does not have the least number of iterations.

Figure 2.1 show the convergence rate of the four algorithms. All algorithms initially start from LP relaxed solution which is an infeasible solution for the IP problem. Over time, algorithms try to find integer solutions and reduce this infeasibility. As more and more integer variables are found, the objective function increases. In this way, as soon as an all-integer set of variables are found; the algorithms will stop and report the best solution that is feasible for the IP problem. Figure 2.1 shows a steep slope only for T-Counter algorithm and all other algorithms have a steady and very gradual slope. The main reason is that T-Counter algorithm fixes a large number of integer variables in each iteration which reduces the number of iterations but on the other hand does not permit the LP relaxation to re-adjust and utilize the vehicles that are left behind due to rounding down. All other three algorithms, fix a very small number of variables in each iterations. This allows the LP relaxation to adjust to the fixed values and re-route the commodities and vehicles to take advantage of any remaining transportation capacity.

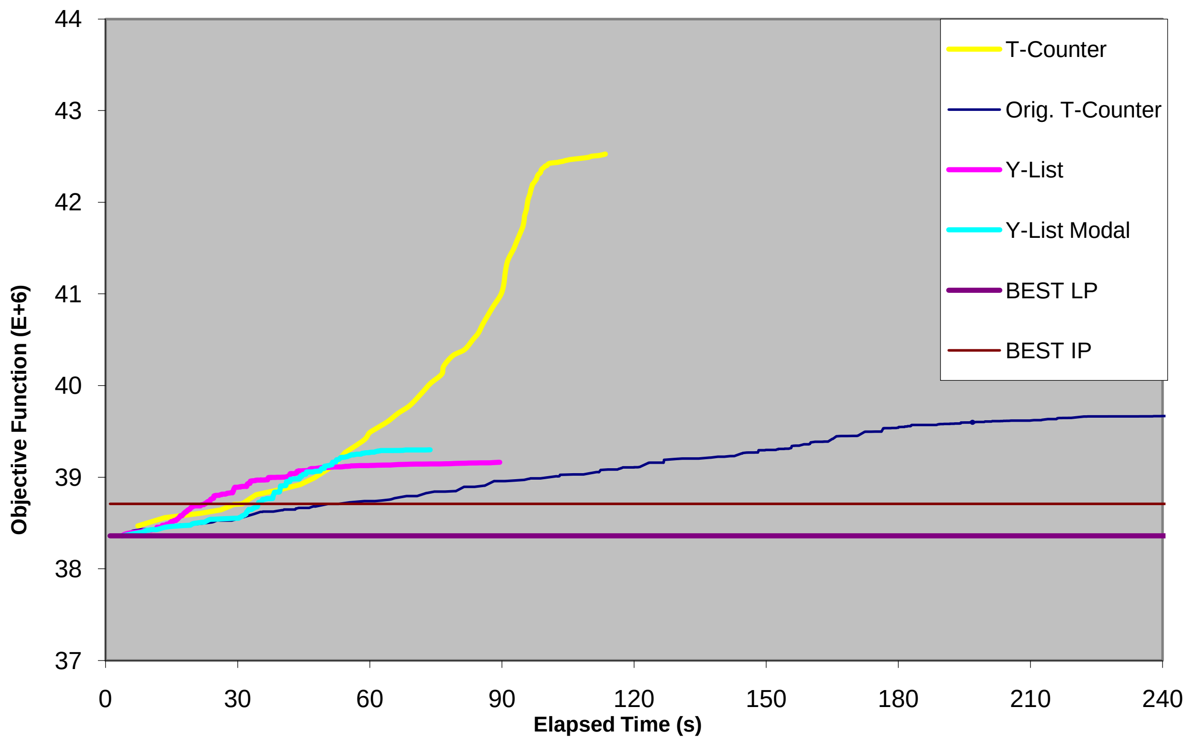

Figure 2.1 – Convergence Rate Comparison of the Proposed Algorithms

Table 2.5 summarizes the analysis and comparisons of the four proposed algorithms. Each row show a criteria and comparatively ranks the four algorithms for that criteria. For example in best solution criteria, Y-List and Y-List Mode are ranked one and two. Comparing the convergence rate, it can be seen that Y-List Modal was the fastest algorithm followed by Y-List algorithm. On the other hand when the Least number of iterations are compared, T-Counter is the winner. Also for theoretical worst-case criteria, T-Counter and Origin-Based T-Counter algorithms are ranked 1st and 2nd. As explained earlier, one iteration of each algorithm does not take the same amount of time as one iteration of other algorithms. Origin-Based T-Counter has the fastest time per iteration followed by Y-List algorithm.

Table 5.5 – Comparative Ranking of the Proposed Heuristic Algorithm

Comparative Ranking	T-Counter	Orig.Based T-Counter	Y-list	Y-list Modal
Best Solution Quality	4th	3rd	**1st**	**2nd**
Convergence Speed	3rd	4th	**2nd**	**1st**
Least No. of Iterations	**1st**	4th	3rd	**2nd**
Best worst-case Scenario	**1st**	**2nd**	4th	3rd
CPU Time per Iteration	4th	**1st**	**2nd**	3rd

2.5.6 Summary

To summarize, it is shown that all four algorithms are capable of finding good quality solutions in relatively short computational times. Having short computation time is the most important property of the proposed algorithm which makes it possible to apply them in real-world dynamic operations. In Fact, the applicability of proposed mathematical model in section 1.4 could not be justified without fast solution algorithms that can adjust and re-optimize in real-time.

Comparing the four algorithms, it is concluded that no single algorithm dominates the others in all ranking criteria. When solution quality and convergence speed is more important, Y-List and Y-List Modal are showed to perform better. On the other hand, when good performance under worst-case scenario is important, T-Counter and Origin-Based T-Counter algorithms are shown to have better statistics.

It should be noted that all four of the proposed algorithms are heuristics algorithms. Even though they showed very impressive results for the current numerical experiment, there is no proof that they will always have equally good performance for all problem instances. As explained in section 2.1.5, this is in the nature of most heuristic algorithms and is not limited to this study. However, detailed sensitivity analysis and performing more numerical experiences under a range of conditions can better assess the merit and applicability of any heuristic algorithm.

BIBLIOGRAPHY

1. Alexander, D.E. 1993, Natural Disasters. London: UCL Press
2. Altay, N., and Walter G. G. 2006. OR/MS research in disaster operations management. European Journal of Operational Research, 175, p475-493.
3. Ardekani S. A. and Hobeika A. (1988) Logistics problems in the aftermath of the 1985 Mexico City earthquake. Transportation Quarterly. 42(1), 107-124.
4. Balas, E. "An Additive Algorithm for Solving Linear Programs with Zero-One Variables." Operations Research 13(1965):517-546.
5. Barbarosoglu, G. and Arda,Y., A two-stage stochastic programming framework for transportation planning in disaster response. Journal of the Operational Research Society, 2004, 55, 43–53.
6. Barbarosoglu, G., Ozdamar, L. and Cevik, A., An interactive approach for hierarchical analysis of helicopter logistics in disaster relief operations. European Journal of Operational Research, 2002, 140, 118–133.
7. Beale, E.M.L. "Survey of Integer Programming." Operation Research Quarterly 16:2(1965):219-228.
8. Beamon, B., 1998. "Supply Chain Design and Analysis: Models and Methods", *International Journal of Production Economics*, Vol. 55, No. 3, pp. 281-294.
9. Beamon, B., Measuring supply chain performance. International Journal of Operations & Production Management; 1999, Vol. 19 Issue 3/4, p275-292.
10. Beamon, B. and Kotleba, S., Inventory Modeling for complex emergencies in humanitarian relief operations. International Journal of Logistics. Vol. 9, No. 1, March 2006, 1-18.
11. Beamon, B., Humanitarian relief chains: Issues and challenges, in Proceedings of the 34th International Conference on Computers and Industrial Engineering, San Francisco, CA, 2004.
12. Benders, J.F. "Partitioning Procedures for Solving Mixed-Variables Programming Problems." Numerical Methods. 4(1962):239-252.
13. Bodin, L.D., Golden, B.L., Assad, A.A., and Ball, M.O. (1983), "Routing and scheduling of vehicles and crews. The state of the art", Computers and Operations Research 10, 69-211.

14. Brown G. G. and Vassiliou A. L. (1993) Optimizing disaster relief: real-time operational and tactical decision support. Naval Research Logistics (NRL). 40, l-23.
15. Christofides, N., Mingozzi, A., and Toth, P. (1981), "Exact algorithms for the vehicle routing problem, based on spanning tree and shortest path relaxations", Mathematical Programming 20, 255-282.
16. Crainic TG, Rousseau J-M. Multicommodity, multimode freight transportation: a general modeling and algorithmic framework for the service network design problem. Transportation Research B: Methodological 1986; 20:225-242.
17. CRED, Annual Disaster Statistical Review: Numbers and Trends 2007. Center for Research on the Epidemiology of Disasters. http://www.cred.be/
18. Dantzig, G.B., D.R. Fulkerson, and S.M. Johnson. "Solution of a Large Scale Traveling Salesman Problem". Operations Research. 2(1954):393-410.
19. Daskin, M.S., 1995. Network and Discrete Location: Models, Algorithms, and Applications. Wiley & Sons, New York.
20. Desrochers M, Lenstra J K, Savelsbergh M W P (1990) A classification scheme for vehicle routing and scheduling problems. European Journal of Operational Research 46: 322–332
21. Dimitruk, Paul. Three keys to supply chain management in times of disaster. Healthcare Purchasing News. Dec 2005.
22. Drezner, Z., Hamacher, H., 2002. Facility Location: Applications and Theory. Springer-Verlag, Berlin.
23. Eksioglu, B., Network algorithms for supply chain optimization. Ph.D. Dissertation, University of Florida 2002.
24. EM-DAT, Emergency Disasters Data Base. http://www.em-dat.net/
25. FEMA Strategic Plan 2008-2013. http://www.fema.gov/about/strategicplanfy08.shtm
26. Fisher, M.L. “Optimal solution of vehicle routing problems using minimum K-trees”, Operations Research, vol. 42, pp. 626–642, 1994.
27. Fisher, M.L."The Lagrangian Relaxation Method for Solving Integer Programming Problems." Management Science 27(1981):1-18.
28. Geoffrion, A.M. "Lagrangian Relaxation and its Uses in Integer Programming." Mathematical Programming Study, 2(1974):82-114.

29. Golden, B., L. Bodin, T. Doyle and W. Stewart. "Approximate Traveling Salesman Algorithms." Operations Research. 28(1980):694-711.
30. Golden, B.L., and Assad, A.A. (1988), Vehicle Routing: Methods and Studies, North-Holland, Amsterdam.
31. Gomory, R.E."An Algorithm for Integer Solutions to Linear Programs." In Recent Advances in Mathematical Programming. (R.L. Graves and P. Wolfe, eds.). McGraw-Hill, New York, 1963:269,302.
32. Guelat J., Florian M. and Crainic T. (1990) A multimode multiproduct network assignment model for strategic planning of freight flows. Transportation Science. 24, 25-39.
33. Haddow. G., Jane A. Bullock. J, Coppola. D., Introduction to Emergency Management, Published by Butterworth-Heinemann, 2007
34. Haghani, A. and Oh, S.C., Formulation and solution of a multi-commodity, multi-modal network flow model for disaster relief operations. Transport. Res. Part A-Policy and Practice, 1996, 30(3), 231–250.
35. Harland, C.M., 1996. Supply chain management: relationships, chains and networks. British Academy of Management 7 (Special Issue), S63-S80.
36. Hughes, M.A., 1991. A selected annotated bibliography of social science research on planning for and responding to hazardous material disasters. Journal of Hazardous Materials 27, 91–109.
37. Kemball-Cook, D., Stephenson, R. (1984), "Lessons in logistics from Somalia", Disasters, Vol. 8 pp.57-66.
38. Knott R. (1987) The logistics of bulk relief supplies. Disasters 11, 113-115.
39. Knott R. (1988) Vehicle scheduling for emergency relief management: a knowledge-based approach. Disasters 12, 285-293.
40. Land, A.H. and A.G. Doig. "An Automatic Method for Solving Discrete Programming Problems." Econometrica 28(1960):497-520.
41. Laporte, G., and Nobert, Y. (1987), "Exact algorithms for the vehicle routing problem", in: S. Martello, G. Laporte, M. Minoux and C. Ribeiro (eds.), Surveys in Combinatorial Optimization, North-Holland, Amsterdam, 147-184.
42. Laporte, G., “The vehicle routing problem: an overview of exact and approximate algorithms”, European Journal of Operational Research, vol. 59, pp. 345–358, 1992.

43. G. Laporte, “Vehicle routing”. In Dell’Amico, Maffioli & Martello (Eds.) Annotated Bibliographies in Combinatorial Optimization. New York, Wiley, 1997.
44. Magnanti,TL,. 1981. Combinatorial Optimization and Vehicle Fleet Planning: Perspectives and Prospects. *Networks* 11, 179-2 14.
45. Mentzer, J.T., W. DeWitt, J.S. Keebler, S. Min, N.W. Nix, C.D. Smith, Z.G. Zacharia. "Defining Supply Chain Management," Journal of Business Logistics, (22:2), 2001, pp. 1–25.
46. Nemhauser G.L. and Wolsey L.A. (1999). *Integer and Combinatorial Optimization*, John Wiley, New York.
47. New, S.J., 1997. The scope of supply chain management research. Supply Chain Management 2 (1), 15-22.
48. New, S.J., Payne, P., 1995. Research frameworks in logistics: three models, seven dinners and a survey. International Journal of Physical Distribution and Logistics Management 25 (10), 60-77.
49. Oh, S.C. and Haghani, A., Testing and evaluation of a multi-commodity multi-modal network flow model for disaster relief management. J. Adv. Transport., 1997, 31(3), 249–282.
50. Oloruntoba, R. and Gray, R., Humanitarian aid: an agile supply chain, Supply Chain Management, 2006, 11(2), 115–120.
51. Owen, S. H. and M. S. Daskin, 1998, “Strategic Facility Location: A Review,” *European Journal of Operational Research*, 111, 423-447
52. Ozdamar, L., Ekinci, E. and Kucukyazici, B., Emergency logistics planning in natural disasters. Ann. Operations Res., 2004, 129, 217–245.
53. Pan American Health Organization. Natural disasters—protecting the public’s health. Washington DC. (2001)
54. Papadimitrou, C.H. and K. Steiglitz. Combinatorial Optimization: Algorithms and Complexity. Prentice-Hall, Englewood Cliffs, NJ, 1982.
55. Ray J. (1987) A multi-period linear programming model for optimally scheduling the distribution of food-aid in West Africa. MSThesis, University of Tennessee, Knoxville, TN.

56. ReVelle, C. S., Eiselt, H.A., and Daskin, M. S., 2008, “A Bibliography for Some Fundamental Problem Categories in Discrete Location Science,” European Journal of Operational Research, 184:3, 817-848.
57. Schilling, D.A., Jayaraman, V. and Barkhi, R., 1993, A review of covering problems in facility location, Location Science 1 (1) (1993) pp. 25-55.
58. Scott, C., Westbrook, R., 1991. New strategic tools for supply chain management. International Journal of Physical Distribution and Logistics 21 (1), 23-33.
59. Tan, K.C., 2001. A framework of supply chain management literature. European Journal of Purchasing & Supply Management 7, 39–48.
60. Thomas, A.S., Humanitarian logistics: enabling disaster response, Fritz Institute, 2007.
61. Thomas, A.S. and Kopczak. L.R., From logistics to supply chain management: The path forward in the humanitarian sector. Fritz Institute, 2005.
62. P. Toth, D. Vigo (Eds.) (2002).*The Vehicle Routing Problem,* SIAM Monographs on Discrete Mathematics and Applications, Philadelphia.
63. Van Wassenhove, L.N., Humanitarian aid logistics: supply chain management in high gear. J. Oper. Res. Soc., 2006,57(5), 475–489.
64. Yi. W. and Ozdamar, L., A dynamic logistics coordination model for evacuation and support in disaster response activities. European Journal of Operations Research 179 (2007) 1177-1193.
65. Zanakis, S.H. and J.R. Evans. "Heuristic Optimization: Why, When and How to Use it." Interfaces. 11(1981):84-90.