

## *Research Article*

# **Real-Time Vehicle Routing for Repairing Damaged Infrastructures Due to Natural Disasters**

**Huey-Kuo Chen,<sup>1</sup> Huey-Wen Chou,<sup>2</sup> Ping-Shan Ho,<sup>1</sup>  
and Hsuan Wang<sup>1</sup>**

<sup>1</sup> *Department of Civil Engineering, National Central University, 300, Jung-Da Road, Jung-Li, Taoyuan 320, Taiwan*

<sup>2</sup> *Department of Information Management, National Central University, 300, Jung-Da Road, Jung-Li, Taoyuan 320, Taiwan*

Correspondence should be addressed to Huey-Kuo Chen, [ncutone@ncu.edu.tw](mailto:ncutone@ncu.edu.tw)

Received 22 August 2010; Revised 10 January 2011; Accepted 28 February 2011

Academic Editor: J. J. Judice

Copyright © 2011 Huey-Kuo Chen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We address the task of repairing damaged infrastructures as a series of multidepot vehicle-routing problems with time windows in a time-rolling frame. The network size of the tackled problems changes from time to time, as new disaster nodes will be added to and serviced disaster nodes will be deleted from the current network. In addition, an inaccessible disaster node would become accessible when one of its adjacent disaster nodes has been repaired. By the “take-and-conquer” strategy, the repair sequence of the disaster nodes in the affected area can be suitably scheduled. Thirteen instances were tested with our proposed heuristic, that is, Chen et al.’s approach. For comparison, Hsueh et al.’s approach (2008) with necessary modification was also tested. The results show that Chen et al.’s approach performs slightly better for larger size networks in terms of objective value.

## **1. Introduction**

Taiwan is located at the active arc-continent collision region between the Luzon arc of the Philippine Sea. Due to the collision of the Eurasian plate and the Philippine plate, Taiwan is at high risk of earthquakes. A 1999 Chi-Chi earthquake with a Richter magnitude of 7.3 [1] resulted in 2415 deaths, the collapse of 40,845 buildings, and damage to 44 major roads [2]. In addition, Taiwan also suffers from typhoons each year. Typhoons such as Kalmaegi and Sinlaku in the summer of 2008 and Morakot in the summer of 2009 cause severe fatalities and uncountable damage to agricultural products and transportation infrastructures.

The Federal Emergency Management Agency [3] published Guide for All-Hazard Emergency Operations Planning which is to develop, in partnership with state and local governments, a national emergency management system that is comprehensive, risk based,

and all-hazard in approach. Crucial to this system are emergency operations plans (EOPs), which describe who will do what, as well as when, with what resources, and by what authority—before, during, and immediately after an emergency. It is apparent that all-hazard emergency operations are interdisciplinary in nature and indeed too broad to be handled by staff in a specialized field. Therefore, to focus, this paper will only address the sequence for repairing damaged infrastructures due to natural disasters.

When a severe natural disaster occurs, transportation infrastructures (either areas, roadways, or both) in the affected area are often seriously damaged, which makes evacuation and rescue operations difficult. The number of fatalities and property losses can increase if damaged transportation infrastructures are not repaired in a reasonably short period of time. To accelerate the completion of repairs, a systematic approach is needed. In this research, we address the task of repairing damaged infrastructures, named the real-time vehicle-routing problem with time windows, as a series of vehicle-routing problems with time windows in a time-rolling frame.

The rest of the paper is organized as follows. Section 2 conducts a literature review. Section 3 describes the problem. Section 4 formulates a multidepot vehicle-routing problem with time windows (VRPTWs) as a mixed integer programming model. Section 5 elaborates the solution algorithm, which is demonstrated with numerical examples in Section 6. Section 7 concludes the paper.

## 2. Literature Review

Scheduling the repair sequence of damaged infrastructures is only part of emergency operations plans. The related literature is relatively scarce, and moreover, the approaches adopted to tackle this type of problems are pretty diverse, covering both simulation and optimization techniques, as well as varying from single objective to multiobjective, or even from one level to bilevel optimization. Literature review is, thus, briefly conducted in the following.

Fiedrich et al. [4] suggest a resource allocation decision supporting system for post-earthquake search-and-rescue (SAR) missions. Both simulation and analytical modeling techniques are employed in the system. The system aims to minimize the number of fatalities and injuries and optimally allocate the available resources. To solve the problem, three methods were compared: simulated annealing, tabu search, and hill climbing. Simulated annealing outperformed the other two methods.

Chang [5] presents the damaged infrastructure repair problem as a multiple objective vehicle-routing problem with time windows. The objectives were to minimize the total travel time of all construction teams, the total rescue operation time, as well as the risk of construction. Two different approaches were used to solve the problem. The first approach adopts the weighted sum of all three objectives as a single objective, while the second approach uses the “maxmin” operator.

Barbarosoğlu et al. [6] develop a mathematical model for helicopter mission planning during a disaster relief operation. The decisions inherent in the problem decompose hierarchically into two subproblems, where tactical decisions are made at the top level, and the operational routing and loading decisions are made at the base level. Consistency between the decomposed problems is achieved with an interactive coordination procedure which transfers anticipated information from the base level to improve the top level decisions. The existence of conflicting multiple objectives in this hierarchical structure requires the development of a multicriteria analysis, and an interactive procedure is designed with the top level decision maker to assess the preference of alternative nondominated solutions.

Lee [7] uses two models to tackle the damaged infrastructure repair problem. The first model determines the sequence for repairing disaster spots by a fuzzy ranking method and assumes that each disaster spot can only be handled by one construction team. The second model assumes that the travel times are deterministic and allows each disaster spot to be handled by one or more construction teams.

Özdamar et al. [8] propose a mathematical model that integrates the multicommodity network flow problem and the vehicle-routing problem for emergency logistics planning in natural disasters. The model is readily decomposed into two multicommodity network flow problems, the first one being linear (for conventional commodities) and the second integer (for vehicle flows). In the solution approach, these submodels are coupled with relaxed arc capacity constraints using Lagrangean relaxation. The convergence of the proposed algorithms is tested on small test instances as well as on an earthquake scenario of realistic size.

To consider both the overall efficiency and total completion time, Liao [9] views the damaged infrastructure repairing problem as a bilevel programming model. The upper-level subproblem maximizes the overall efficiency of the repair work, whereas the lower-level subproblem minimizes the total completion time subject to the time constraint for each disaster spot.

Pettit and Beresford [10] examine the processes involved in emergency relief and highlight the relationships between the participating bodies, that is, military and nonmilitary involvement. A composite model is proposed, which incorporates Jennings et al.'s [11] model as well as the military/nonmilitary dimension, and it appears to be robust and workable in a range of geopolitical and operation circumstances.

Hsueh et al. [12] optimize the rescue operation using a time-constrained vehicle-routing approach. In their scheme, a fleet of vehicles is dispatched to simultaneously deliver emergency materials and medical equipment to the affected areas and to pick up materials and equipment that are no longer needed. The time-rolling horizon approach is adopted to take care of demands that are either known in advance or that appear in real time. The severity of each disaster node is reflected by the associated time window (the tighter the time window, the more severe the disaster node). In addition, it is also assumed that there is no need for the dispatched vehicles to return to the depot. A two-stage solution heuristic is proposed. In the first stage, the minimum cost insertion method is used to find an initial solution; in the second stage, Or-opt method is adopted.

Wang [13] addresses the damaged infrastructure repair problem as a multidepot vehicle-routing problem that adapts fuzzy repair times. A time rolling horizon approach is used to solve the problem. At each specific time instant, only the part of the network consisting of accessible damaged infrastructure segments is treated. The damaged infrastructure segments shown in the partial network are optimally scheduled for repair by the construction teams. In other words, those inaccessible damaged infrastructure segments are not considered for repair at the moment. However, an inaccessible damaged infrastructure segments may become accessible later when its adjacent disaster nodes are repaired either partially or fully. Once a damaged infrastructure segment becomes accessible, it will become visible in the updated network and can be scheduled for repair. At each specific instant, the model minimizes the total travel time. A two-stage solution algorithm is proposed. The first stage uses the nearest/farthest insertion method for generating an initial solution, while the second stage uses the 2-opt, swap, and or-opt methods for improving the initial solution. The disadvantage associated with this approach is that the disaster nodes not appeared in the earlier partial network cannot be considered, and the severity as well as time

window constraint of these “invisible” disaster nodes is not suitably reflected in the very beginning of the entire optimization procedure.

Since disaster nodes in the affected area are often arranged in order and the number of construction teams available is usually smaller than the number of disaster nodes, the vehicle-routing approach is appropriate to address the damaged infrastructure repair problem. Therefore, this research treats the damaged infrastructure repair problem as a multidepot vehicle-routing problem with time windows and formulates it as a series of mixed integer programming problems. This is analogous to Hsueh et al.’s [12] approach but with necessary modifications, especially in scheduling the sequence for repairing the damaged infrastructures, which are largely, affected by the changeable link travel times and the completion time of repairing adjacent disaster nodes.

### 3. Problem Description

Natural disasters like earthquakes and typhoons often damage infrastructures that need to be repaired. The damaged infrastructure repair problem has the following special features.

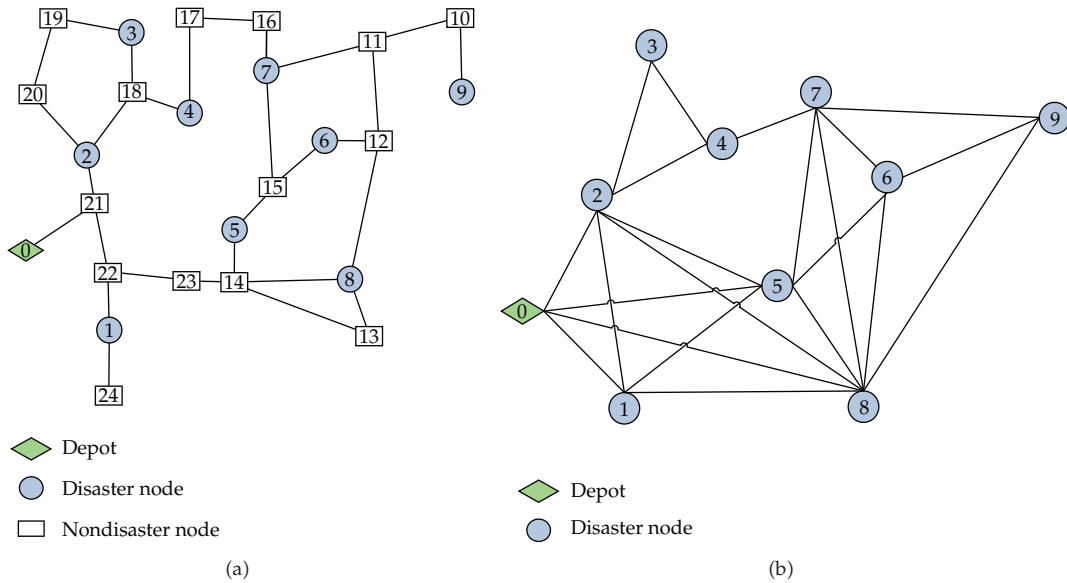
- (1) Repair sequence must be complied between some disaster nodes. One disaster node may be unreachable if it is surrounded by adjacent disaster nodes. In other words, this disaster node can be repaired only when one of adjacent disaster nodes has been repaired. This feature is unique and has to be esteemed in scheduling the repair sequence of disaster nodes.
- (2) Disaster-related information is changeable. The information such as road travel times, the severity of disaster nodes, or the number of disaster nodes, and construction teams may change from time to time, implying the damaged infrastructure repair plan must be updated in a real-time fashion.
- (3) Construction teams do not have to return to associated depots when their assigned work has been completed. This assumption is mainly due to the consideration that additional disaster nodes may appear after the work of a construction team has been completed. Note that assigning a construction team on-the-spot to a new disaster node for repair is usually faster than assigning a construction team that is already back to depot.

All these special features need to be properly tackled by the following proposed approach.

#### 3.1. VRP Network

To address the damaged infrastructure repair problem as a series of vehicle-routing problems (VRPs), a VRP network must be constructed from the corresponding physical transportation network. To show how a physical network can be transformed into a corresponding VRP network, we use a numerical example for illustration. Suppose that we have the physical network of a disaster area as shown in Figure 1(a). One depot is marked in green diamonds; 9 damaged infrastructure segments are marked in gray circles (in which the first 8 disaster nodes are identified before performing the construction work and disaster node 9 is reported later), and 15 intersections and/or landmarks are marked in white rectangles. The current link travel times are fixed and externally determined.

To transform a physical transportation network into its corresponding VRP network, the following two steps are performed.



**Figure 1:** (a) Physical transportation network for the disaster area. (b) VRP network for the disaster area.

*Step 1.* Omit the intersections marked in white rectangles from the VRP network.

*Step 2.* Represent headquarters and damaged infrastructure segments as nodes in the VRP network and denote link travel times by the shortest distances between nodes (i.e., measured from the physical transportation network for each pair of nodes in Figure 1(a)).

The travel time on link  $3 \rightarrow 4$  in the VRP network, as shown in Figure 1(b), is represented by the travel time on path  $3 \rightarrow 18 \rightarrow 4$  in Figure 1(a) (which can be obtained by adding the link travel times for links  $3 \rightarrow 18$  and  $18 \rightarrow 4$ ). When the two nodes are disconnected due to natural disasters, then the corresponding link travel time is represented by a prohibitive large number.

In Figure 1(b), the VRP network is incomplete; that is, some nodes are not reachable and can only be accessed with prohibitive large link travel times. This special feature does not exist in traditional VRP problems, which implies that the solution algorithms for traditional VRP cannot be directly used. For instance, damaged infrastructure 4 cannot be repaired unless damaged infrastructure 3 or 2 has been repaired. This implies that the repair work must be performed in a one-by-one sequential manner.

### 3.2. Repair Sequence

To show how the sequence for repairing damaged infrastructure segments proceeds in natural disasters, we use the hypothetical network shown in Figure 1(b) for illustration. At the beginning, this network consists of one depot (numbered as “0”) and eight disaster nodes (numbered from “1” to “8”). A depot is where construction teams initially reside; disaster nodes represent damaged infrastructures to be repaired. Two types of links are used. A solid line denotes a normal link, that is, the corresponding link travel time is finite, and its end nodes are connected. A dotted line indicates that the corresponding link travel time is

prohibitively high, and its end nodes are adjacent but cannot be accessed by passing through it, that is, an isolated link. The accessibility of one disaster node by the construction team is clearly dependent on whether there are one or more solid links associated with it.

In Figure 2(a), the construction team is initially located at headquarter “0,” and only nodes 1, 2, 5, and 8 can be accessed and repaired. If by some criterion, the construction team decides to repair node 2, then nodes 1, 3, 4, 5, and 8 will become accessible via node 2. In other words, the link travel times for links (2,1), (2,3), (2,4), (2,5), and (2,8) will change. With this updated travel time information, as shown in Figure 2(b), node 2 now acts as the new starting node for the dispatched vehicle in the network. Referring to Chen et al. [14], this new starting node is called the critical node. The critical node plays an important role in distinguishing between a serviced disaster node and an unserviced disaster node. A critical node is defined as a node that is currently using a construction team or to where a construction team is heading. Critical nodes need to be identified instantly when a demand or travel time has changed in real time so that the route can be reconstructed.

Upon repairing node 2, nodes 1, 3, 4, 5, and 8 become accessible. We assume by some criterion that the construction team selects node 1 among five accessible nodes for repair. As shown in Figure 2(c), upon repairing node 1, nodes 3, 4, 5, and 8 become accessible. Of the four accessible nodes, the construction team next chooses node 3 for repair. Figure 2(d) indicates that nodes 4, 5, and 8 become accessible after repairing node 3. Figure 2(e) shows nodes 5, 7, and 8 become accessible when the construction work for node 4 is completed. In addition, disaster node 9 is identified now. Figures 2(f), 2(g), 2(h), 2(i), and 2(j) indicate that nodes 5, 9, 7, 6, and 9 are repaired sequentially, and the entire construction work has been completed. Figure 2(k) shows the entire sequence for repairing all disaster nodes.

From the above description, it is clear that a disaster node that is initially inaccessible by the construction team can become accessible if its adjacent disaster nodes are repaired. This description naturally motivates a “take-and-conquer” solution concept for the damaged infrastructure repair problem, which is essentially equivalent to solving a series of VRP problems in a time-rolling horizon. Each VRP network to be tackled at a specific instant is constructed based on updated information such as link travel times and real-time repairing requests from the disaster area. The detailed steps of the proposed solution algorithm will be discussed later.

## 4. Model Formulation

### 4.1. Assumptions

The damaged infrastructure repair problem is very complex, and there is no common definition in the literature. For the purpose of our research, we make the following assumptions.

- (1) The sequence for repairing damaged infrastructure segments cannot be predetermined. It should be determined by a series of VRP problems in a time-rolling horizon. Each VRP problem is formulated at a specific time instant and solved based on updated information, including the number of disaster nodes yet serviced, the estimated repair time for each unserviced node, and the estimated link travel time for each pair of nodes.
- (2) The estimated repair time for each unserviced node is externally determined. In a natural disaster, damaged infrastructure segments are usually treated in two stages. In the first stage, partial or minimum function of the infrastructure segments

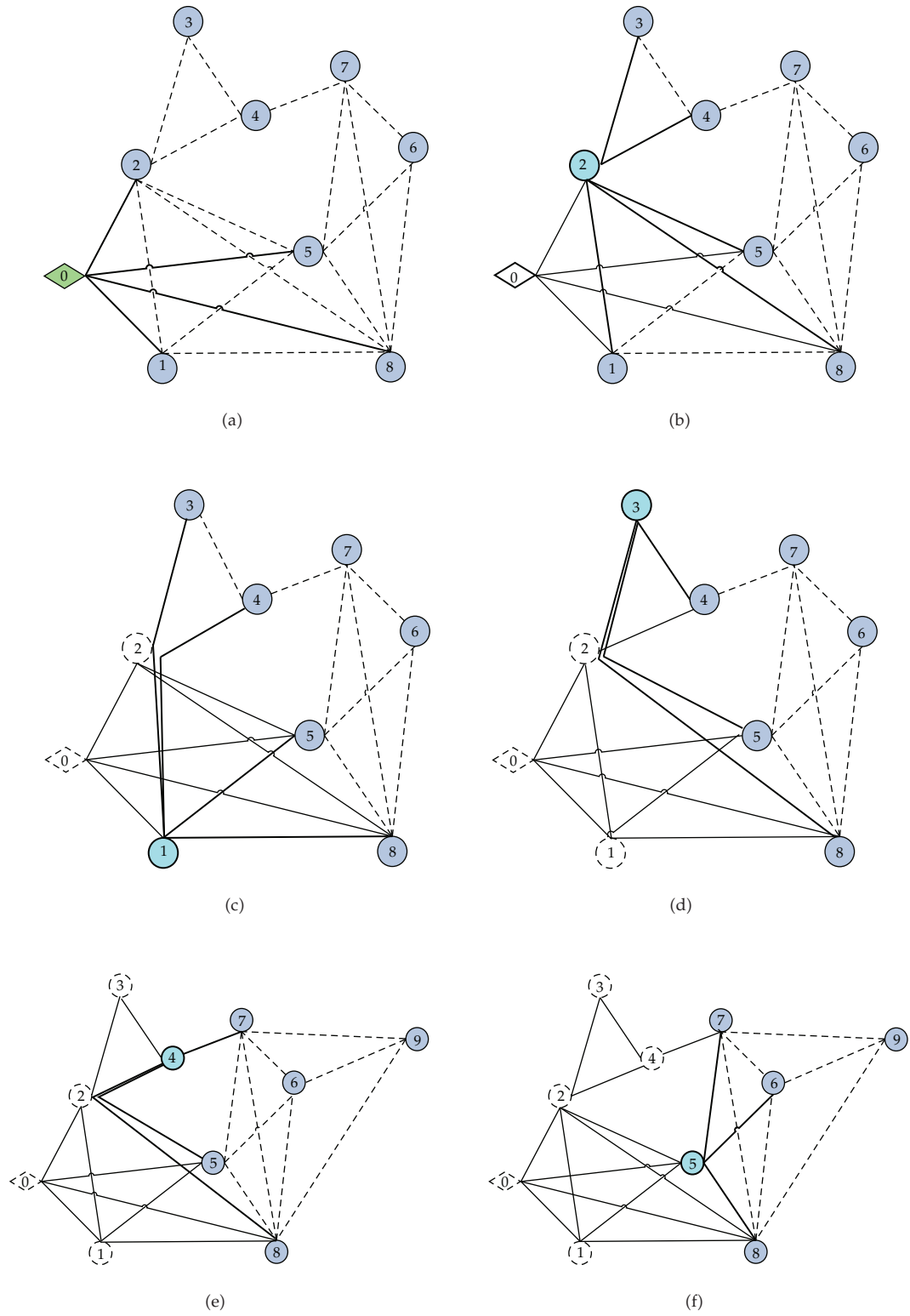
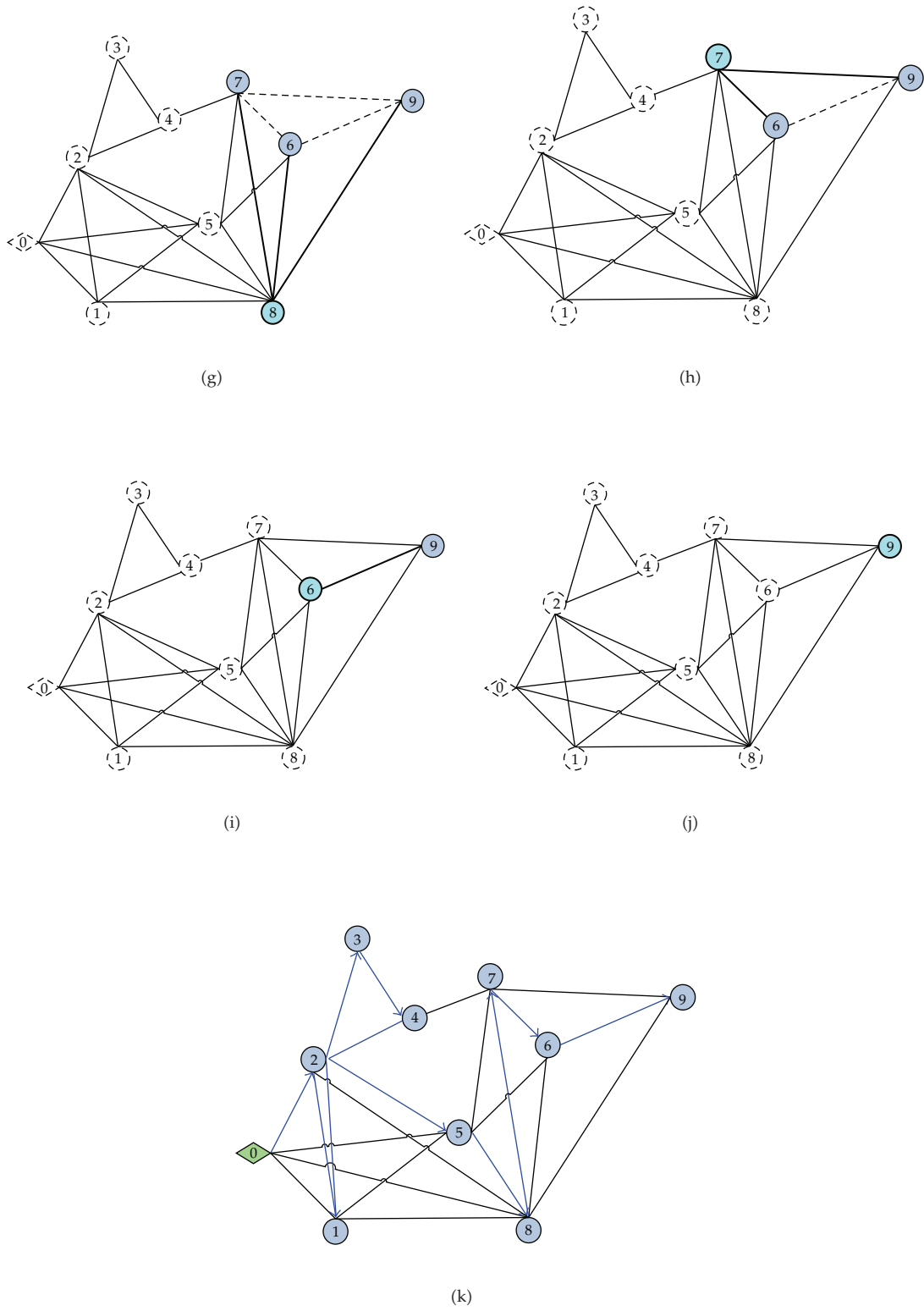


Figure 2: Continued.



**Figure 2:** Sequence for repairing disaster nodes.



must be restored by the emergency construction teams. In the second stage, full functioning of the infrastructure segments must be achieved. In this research, only the first stage is addressed.

- (3) The repair work for each damaged infrastructure segment cannot be interrupted until all repair work has been completed.
- (4) The number of construction teams available is known at each time instant. No consideration is given to the vehicle dispatching cost, shortage of machines/materials, or manpower limitation for dispatching construction teams. When a new construction team is formed during the rescue period, it must be dispatched from a depot.
- (5) Travel times on normal links are fixed and can be collected externally. This is because the entire disaster area is under the control of police; the right of way is given to the emergency construction teams, and no irrelevant personnel or vehicles are allowed to enter the area.
- (6) One disaster node can be serviced by only one construction team. If a disaster node is too severe to be handled by one construction team, it can be further divided into several subnodes. Each subnode can be repaired by one construction team in a reasonable period of time. In addition, no construction team can service two or more disaster nodes in the same time.
- (7) When the assigned repair work is complete, the construction team must stay there and wait for the next assignment. This assumption is different from the traditional VRP problem, which requires the dispatched vehicle to return to the depot when the assigned work has been completed.
- (8) The sequence for repairing disaster nodes is affected by the spatial layout (or topological structure). For two adjacent disaster nodes connected by a dotted link, one disaster node cannot be repaired unless the other disaster node has been repaired.

## 4.2. Notations

The following notations are used in this paper.

- (1) Index

0: depot designation,  
*c*: critical node designation,  
*i, j, h*: node designation,  
*k*: construction team designation,  
*u*: unserviced node designation.

- (2) Parameters and constants

$\Delta$ : CPU time estimated for solving the problem,  
 $\tau$ : time instant equal to the time for receiving new information plus the CPU time  $\Delta$  required for solving the problem. Note the CPU time can be neglected without causing significant effect, given the relatively short solution times and long travel and repair times,  
 $c_{ij}(\tau)$ : estimated travel time on link  $i \rightarrow j$  at time instant  $\tau$ ,

$r_i(\tau)$ : estimated repair time for disaster node  $i$  at time instant  $\tau$ ,  
 $l_i$ : latest time (or upper bound) for node  $i$  to be repaired,  
 $\bar{k}_i(\tau)$ : specific construction team that leaves critical node  $i$  at time instant  $\tau$ ,  
 $W_i(\tau)$ : estimated weight associated with the violation of time window, reflecting the importance of the time window associated with disaster node  $i$  at time instant  $\tau$ .

## (3) Node sets

$N_0(\tau)$ : set of depots at time instant  $\tau$ ,  
 $N_c(\tau)$ : set of critical nodes at time instant  $\tau$ ,  
 $N_u(\tau)$ : set of unserved disaster nodes at time instant  $\tau$ ,  
 $N_{cu}(\tau)$ : set of critical and unserved disaster nodes at time instant  $\tau$ ,  
 $N_{0u}(\tau)$ : set of depots and unserved disaster nodes at time instant  $\tau$ ,  
 $N_{0cu}(\tau)$ : set of depots, critical and unserved disaster nodes at time instant  $\tau$ .

## (4) Sets of construction teams

$K_i(\tau)$ : set of construction teams available at depot  $i$ ,  $i \in N_0(\tau)$ , at time instant  $\tau$ ;  $K_i(\tau) \subseteq K(\tau)$ ,  
 $K(\tau)$ : set of the total construction teams available in the entire affected area at time instant  $\tau$ .

## (5) Variables

$a_i$ : arrival time at disaster node  $i$ ,  
 $d_i$ : departure time from disaster node  $i$ ,  
 $d_{ik}$ : departure time of vehicle  $k$  from depot  $i$ ,  $i \in N_0(\tau)$ ,  
 $x_{ijk}$ : 0-1 decision variable; if construction team  $k$  leaves disaster node  $i$  toward disaster node  $j$ , then  $x_{ijk} = 1$ ; otherwise,  $x_{ijk} = 0$ .

**4.3. Model Formulation**

The damaged infrastructure repair problem is addressed as a series of multidepot VRP problems in the time-rolling horizon. Whenever network information is updated, a new VRP problem must be formulated as a 0-1 mixed integer programming problem and solved accordingly. Referring to Hsueh et al. (2008), the multidepot VRP problem at specific instant  $\tau$  can be formulated as follows.

## (1) Objective function:

$$\text{Min} \sum_{i \in N_{0cu}(\tau)} \sum_{j \in N_{0u}(\tau)} \sum_{k \in K(\tau)} c_{ij}(\tau) x_{ijk} + \sum_{i \in N_u(\tau)} W_i(\tau) \{\max(a_i - l_i, 0)\}. \quad (4.1)$$

## (2) Flow conservation constraints:

$$\sum_{j \in N_{0u}(\tau)} \sum_{k \in K(\tau)} x_{ijk} = 1 \quad \forall i \in N_{cu}(\tau), \quad (4.2)$$

$$\sum_{i \in N_{0cu}(\tau)} \sum_{k \in K(\tau)} x_{ijk} = 1 \quad \forall j \in N_u(\tau), \quad (4.3)$$

$$\sum_{i \in N_{0cu}(\tau)} x_{ihk} - \sum_{j \in N_{0cu}(\tau)} x_{hjk} = 0 \quad \forall h \in N_u(\tau), k \in K(\tau), \quad (4.4)$$

$$\sum_{j \in N_{0cu}(\tau)} x_{ij\bar{k}_i(\tau)} = 1 \quad \forall i \in N_c(\tau), \quad (4.5)$$

$$\sum_{j \in N_u(\tau)} x_{ijk} \leq 1 \quad \forall i \in N_0(\tau), k \in K_i(\tau), \quad (4.6)$$

$$x_{ijk} \in \{0, 1\} \quad \forall i \in N_{0cu}(\tau), j \in N_{0u}(\tau), i \neq j, k \in K(\tau). \quad (4.7)$$

(3) Definitional constraints:

$$a_j = d_i + c_{ij}(\tau) \quad \text{if } x_{ijk} = 1 \quad \forall i \in N_{cu}(\tau), j \in N_u(\tau), k \in K(\tau), \quad (4.8)$$

$$d_i = \begin{cases} \max(\tau, a_i + r_i(\tau)) & \text{if } \sum_{k \in K(\tau)} x_{ijk} = 0, \\ \infty & \text{if } \sum_{k \in K(\tau)} x_{ijk} = 1, \end{cases} \quad \forall i \in N_{cu}(\tau), j \in N_{cu}(\tau), \quad (4.9)$$

$$d_{ik} = \tau \quad \text{if } x_{ijk} = 1 \quad \forall i \in N_0(\tau), j \in N_u(\tau), k \in K_i(\tau). \quad (4.10)$$

Objective function (4.1) minimizes the weighted sum of the total vehicle travel time and the penalty associated with violating the time window at all disaster nodes. The penalty incurred at node  $i$  is defined by the multiplication of the violation of time window and the estimated weight  $W_i(\tau)$ . If the lower limit of the time window at disaster node  $i$  is violated, that is,  $a_i - l_i > 0$ , then a penalty occurs; otherwise, the penalty vanishes. Note that the larger the value of weight  $W_i(\tau)$ , the higher the priority given to disaster node  $i$ . When all the assigned weights are very large, the final solution will be largely determined by the “soft” time-window constraints, rather than the total travel time, in the objective function.

Constraints (4.2) through (4.7) manage construction teams at disaster nodes. Equation (4.2) requires that only one construction team can leave from a critical disaster node or unserved disaster node  $i$  once. Equation (4.3) denotes that only one construction team can arrive at unserved disaster node  $j$  once. Constraints (4.2) and (4.3) together indicate that each disaster node can be serviced by exactly one construction team. Constraint (4.4) requires that one construction team that enters disaster node  $h$  earlier must leave that node later. Equations (4.5)-(4.6) designate that each vehicle can leave the depot once at most. Constraint (4.5) requires that construction team  $\bar{k}_i$  which has arrived at or is approaching critical disaster node  $i$ , must also leave this disaster node once. Note that construction team  $\bar{k}_i$  is known at

time  $\tau$ . Constraint (4.6) denotes that each construction team  $k$  can be dispatched from depots at most once. Constraint (4.7) defines variable  $x_{ijk}$  as a 0-1 integer variable. If construction team  $k$  leaves node  $i$  toward  $j$ , then  $x_{ijk} = 1$ ; otherwise,  $x_{ijk} = 0$ .

Constraints (4.8) through (4.10) define arrival and departure times. Constraint (4.8) defines the arrival time at node  $j$  by adding the departure time from node  $i$  to the link travel time between node  $i$  and node  $j$ . Constraint (4.9) defines the completion time of repair work at node  $i$ . If node  $i$  is an unserved node, then the completion time of the repair work is defined by the arrival time of the construction team plus the service time; that is,  $d_i = a_i + s_i$ . If node  $i$  is a critical node, there are two conditions to be considered. First, if disaster node  $i$  is not the last to be serviced, that is,  $\sum_{j \in N_0} \sum_{k \in K} x_{ijk} = 0$ , then the completion time for the repair work is defined by formula  $d_i = \max(\tau, a_i + s_i)$ . On the other hand, if node  $i$  is the last to be serviced, that is,  $\sum_{j \in N_0} \sum_{k \in K} x_{ijk} = 1$ , then the departure time of the construction team is set as infinitive; that is,  $d_i = \infty$ . This is equivalent to requiring the construction team stay at the current node to wait for the next assignment after the repair work is finished. Constraint (4.10) defines the departure time for construction team  $k$  to leave the depot.

## 5. Solution Procedure

As mentioned, the damaged infrastructure repair problem is addressed as a series of multidepot VRP problems with time windows. The solution procedure continuously updates real-time information such as demands and link travel times and determines when a new VRP problem is to be formed. Each VRP problem is then solved using a two-stage solution algorithm in which an initial solution is generated in the first stage and further improved upon in the second stage. In the following sections, the framework of the solution procedure is introduced, followed by the initial solution construction and improvement.

### 5.1. Unified Framework

While repairing the damaged infrastructures, relevant information is continuously updated. To take this new information into consideration, the routing schedule of the construction teams must be reoptimized. Changeable information could include the link travel time, the number of construction teams, as well as the number of disaster nodes.

- (1) New link travel times: suppose that we have two nearby damaged bridges, denoted by two disaster nodes, along a freeway. If the nearest damaged bridge has not been repaired, then the second bridge cannot be easily reached. In this situation, the link travel time between them is very large and is symbolically represented by the total network travel time under normal conditions. However, when the near damaged bridge is repaired, the farther damaged bridge becomes accessible, and repair work can be scheduled. In this situation, the link travel time is reduced from a large number to a much smaller value and hence must be updated. This operational sequence is called a take-and-conquer strategy. In other words, whenever new information is generated the network data, and hence, the optimal schedule for repairing the damaged infrastructure segments will be changed accordingly.
- (2) new construction teams: new construction teams may be formed or borrowed/leased from other administrative agencies. Since different number of construction teams may have different routing plans, the corresponding VRP problem needs to be reformulated and reoptimized.

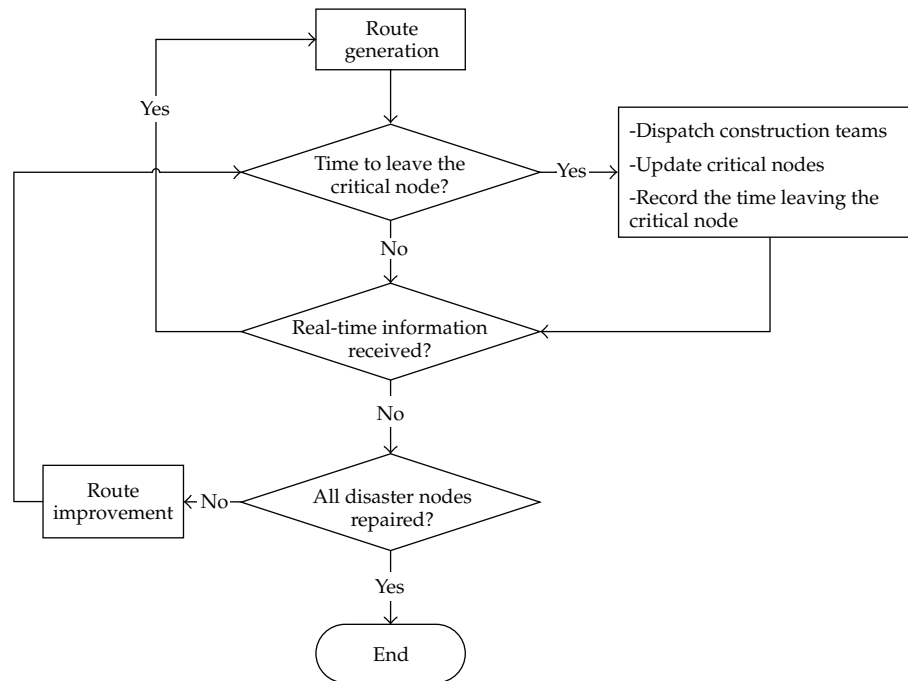


Figure 3: Unified framework for solution procedure.

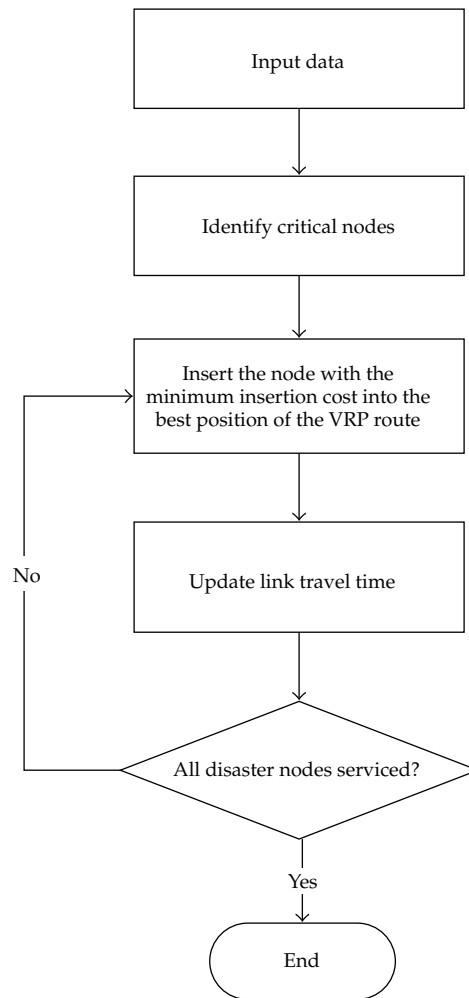
- (3) New disaster nodes: information about infrastructure damages is not perfect immediately following a natural disaster but will become more transparent over time. When new disaster nodes are taken into consideration, the repair plan must be reoptimized.

The framework for the entire solution procedure is depicted in Figure 3. The repair operation begins with a routing plan based on the available information. This routing plan can be changed, if necessary, to include updated information such as new link travel times, new disaster nodes, as well as new construction teams. The new routing plan can be obtained by solving the VRP problem, which must start routes from the critical nodes. A critical node may be thought of as a virtual depot for the subnetwork consisting of the remaining unserved disaster nodes. This procedure for solving a series of multidepot VRP problems with time windows continues until no more unserved disaster nodes exist. As in the literature, the time-constraint VRP problem is solved using a two-stage algorithm, that is, initial solution generation and initial solution improvement, which will be detailed in the following sections.

## 5.2. Initial Solution Generation

To generate initial routes, the minimum cost insertion method is adopted. A route starts from the depot and chooses the next disaster node with the minimum insertion cost.

The incurred insertion cost for unserved disaster node  $k$  between two consecutive served disaster nodes  $i$  and  $j$ , in a “construction team” route, is computed as follows.



**Figure 4:** Flow chart for initial solution construction.

Insertion cost = (cost of travel from disaster node  $i$  to the newly scheduled disaster node  $k$ ) + (cost of travel from the newly scheduled disaster node  $k$  to disaster node  $j$ ) – (cost of travel between disaster node  $i$  and disaster node  $j$  on the “construction team” route).

This process for choosing the next disaster node continues until all disaster nodes have been serviced (see Figure 4). The minimum cost insertion method can also be applied with minor modifications to situations at which multiple depots exist.

The steps for generating initial routes can be described as follows.

*Step 1.* Input data.

Input link travel times, estimated repair time for each disaster node, and upper limit of time windows for each disaster node.

*Step 2.* Identify critical disaster nodes.

Identify critical disaster nodes to begin with initial route construction for the remaining unserved disaster nodes.

*Step 3.* Construct initial VRP routes using the minimum insertion cost method.

*Substep 3.1.* Select an unserved disaster node one by one for each VRP route and compute its insertion cost (including its added travel time and penalty incurred due to the violation of time windows) at different positions in the route.

*Substep 3.2.* Choose the route position with the minimum insertion cost.

*Step 4.* Insert the chosen disaster node into the best position of the route and update the link travel times for the subsequent disaster nodes accordingly.

*Step 5.* If no more unserved nodes exist and the initial routes have been constructed, then return to the unified framework; otherwise, go back to Step 3.

### **5.3. Initial Solution Improvement**

To improve the initial solution, two types of heuristics can be considered. The first type is known as local search in which nodes and/or links are exchanged within intraroutes or between interroutes. It is easy to implement; however, in some circumstances, the heuristic may be trapped in a local region. In addition, the repair sequence (or precedent-successor relationship) of disaster nodes makes the local search complicated. The second type, called metaheuristics, is capable of moving beyond local optima and hence suitable for difficult combinatorial problems such as vehicle-routing problems. A well-known metaheuristic is called ant colony optimization (ACO) which involves a collection of various ant-related algorithms. The ant system [15] was the first ACO made available to academia. Later, various modified algorithms were published, such as the elitist ant system [16], the ant colony system (ACS)[17], the max-min ant system [18], the rank-based ant system [19], the fast ant system [20], and approximate nondeterministic tree-search procedures [21].

In this paper, we adopt the ACS, to improve the initial solution, because it has been successfully employed in solving VRP problems. As an ant-related algorithm, the ACS is motivated by the behavior of ants seeking a path between their colony and a source of food. When one ant finds a good (i.e., short) path from the colony to a food source, other ants are more likely to follow that path, and positive feedback eventually leads all the ants following a single path. The idea of the ACS is to mimic this behavior with “simulated ants” walking around the graph representing the problem to solve.

The ACS consists of four major steps: initial condition, edge selection, pheromone update, and termination criterion. We describe them below.

#### *(1) Initial Condition*

Six parameters are used to set the initial values (Table 1).

#### *(2) Link Selection Rule*

To select the next link for a route to augment, the state transition rule is employed. Two rules, exploitation and exploration, can be used depending on the random number  $q$  generated.

**Table 1:** Parameter settings.

Parameter	Definition
$q_0$	Constant between 0 and 1, $0 \leq q_0 \leq 1$ ; we set $q_0 = 0.9$
$\rho$	Rate of pheromone evaporation in local updating rule, $0 < \rho < 1$ ; we set $\rho = 0.1$
$\alpha$	Rate of pheromone evaporation in global updating rule, $0 < \alpha < 1$ ; we set $\alpha = 0.9$
$\beta$	Parameter to control the importance of edge desirability, determined by the trial-and-error method, usually greater than 0; we set $\beta = 5$
$m$	Number of artificial ants used; we set $m = 10$ . Each ant simulates a solution consisting of a set of routes
$\tau_0$	Initial pheromone value, usually set as $\tau_0 = (NL_{nn})^{-1}$ , where $N$ denotes the number of links and $L_{nn}$ represents the route length computed by the nearest neighbor search
Iter	the maximum number of ACS iterations; we set Iter = 30

If  $q \leq q_0$ , then the link with the higher pheromone value and the shorter length will be chosen as shown in (5.1).

$$\text{Exploitation: } s = \begin{cases} \arg \max_{j \in J_k(i)} \{ [\tau_{ij}] \cdot [\eta_{ij}]^\beta \} & \text{if } q \leq q_0, \\ S & \text{otherwise.} \end{cases} \quad (5.1)$$

On the contrary, if  $q > q_0$ , then an ant will move from node  $i$  to node  $j$  with the probability computed by the following formula:

$$\text{Exploration: } S_k(i, j) = \begin{cases} \frac{\tau_{ij} \cdot [\eta_{ij}]^\beta}{\sum_{u \in J_k(i)} \tau_{iu} \cdot [\eta_{iu}]^\beta} & \text{if } j \in J_k(i), \\ 0 & \text{otherwise.} \end{cases} \quad (5.2)$$

where,  $\tau_{ij}$  is the amount of pheromone on link  $(i, j)$ ,  $\eta_{ij}$  is the desirability or visibility of link  $(i, j)$  (a priori knowledge, typically  $\eta_{ij} = 1/L_{ij}$ ),  $L_{ij}$  is the length of link  $(i, j)$ ,  $J_k(i)$  is the set of unserved nodes (indexed as  $i$ ) by ant  $k$ ,  $\beta$  represents the importance of link length relative to pheromone value ( $\beta > 0$ ),  $q$  is a random variable uniformly distributed between 0 and 1,  $S$  is the next node as determined by (5.1),  $S_k$  is the next node to visit as determined by (5.2).

The probability for link  $(i, j)$  being chosen will increase as the associated pheromone value gets higher and distance gets shorter. However, it is no guarantee that the link with the highest probability will be chosen for route expansion.

### (3) Pheromone Update

Two rules are used to update pheromone values. The local updating rule shown in (5.3) is used to update pheromones on all routes when all ants have moved to next node.

$$\tau_{ij}(t+1) = (1 - \rho)\tau_{ij}(t) + \rho \cdot \tau_0(t). \quad (5.3)$$



As shown in (5.4), the global updating rule is used to increase the pheromone value on the shortest path only when all ants have completed their routes

$$\tau_{ij}(t) = (1 - \alpha)\tau_{ij}(t) + \alpha \cdot \Delta\tau_{ij}, \quad (5.4)$$

where  $\rho$  is the rate of pheromone evaporation in the local updating rule,  $0 < \rho < 1$ ,  $\alpha$  is the rate of pheromone evaporation in the global updating rule,  $0 < \alpha < 1$ ,  $\Delta\tau_{ij}$  is the amount of pheromone deposited, typically given by

$$\Delta\tau_{ij} = \begin{cases} \frac{Q}{L_{gb}} & \text{If link } ij \text{ is on the best route,} \\ 0 & \text{Otherwise,} \end{cases} \quad (5.5)$$

where  $Q$  is constant and  $L_{gb}$  is the length of the shortest path.

#### (4) Termination Criterion

Four termination criteria are commonly used in the ACS method: the maximum number of iterations, the maximum number of consecutive iterations in which the objective value is not improved, the maximum execution time, or the acceptable objective value. In this paper, a maximum number of iterations after which the corresponding objective value cannot be improved are used as the termination criterion.

The flow chart for the initial solution improvement is depicted in Figure 5.

According to Figure 5, the algorithmic steps of the ACS can be described as follows.

*Step 1.* Set parameters. Initialize the number of ants  $m$ , the rate of pheromone evaporation  $\rho$  in the local updating rule, the rate of pheromone evaporation  $\alpha$  in the global updating rule, the parameter to represent the relative importance of link length to pheromone value  $\beta$ , and the initial pheromone value  $\tau_0$ . Set the initial solution as the temporary best solution.

*Step 2.* Dispatch all ants from the depot(s).

*Step 3.* For each ant, select the most suitable construction team and send it to the next disaster node using the link selection rule.

*Step 4.* For each ant, perform the local rule to update pheromone values.

*Step 5.* If all ants have completed route construction, continue; otherwise, go back to Step 3.

*Step 6.* Determine if any constructed solution is superior to the current best solution. If not, make no change; otherwise, replace the current best solution and perform the global rule to update pheromone values.

*Step 7.* Determine if the termination criterion is satisfied. If so, then stop; the stage of initial solution improvement has completed. Otherwise, go back to Step 2.

## 6. Computational Experiments

We generate 13 numerical examples for testing. Our assumptions are described as follows.

- (1) There are two depots, indexed as 1 and 2, respectively.
- (2) For example 1, 5 construction teams are available by the time when disaster occurs. Among which 2 construction teams are located in depot 1 while the other 3 construction teams in depot 2.
- (3) For examples 2–13, 15 construction teams are available by the time when disaster occurs. Among which 7 construction teams are located in depot 1 while the other 8 construction teams in depot 2.
- (4) There are between 10 and 70 nodes, indexed from 3 on.
- (5) The recovered travel times for all pairs of two disaster nodes are randomly generated between 0.3 and 28 hours; hence, all test problems are complete networks after repairing.
- (6) The repair time for each disaster node is randomly generated from 2 to 90 hours.
- (7) The time window for each disaster node is externally determined.
- (8) The unit penalty for the violation of gold rescue time (i.e., 72 hours) is set as 10.

Using the above data two experiments are conducted. The first experiment adopts our model and solution algorithm (called Chen et al.'s approach hereafter), while the second experiment chooses an outside source, that is, Hsueh et al.'s approach (2008) with necessary modifications. Hsueh et al.'s approach is different from ours in four aspects: (1) it is designed for single depot problem rather than multiple-depot problem; (2) it takes care of both delivery and pickup operations; (3) it assumes network is always connected and will not change throughout the analysis period; (4) there is no restriction on the sequence of visiting customer nodes. The obtained results from the test networks are summarized in Table 2.

Both Chen et al.'s and Hsueh et al.'s approaches can improve 12 of 13 test examples in terms of objective function value, with average improvement rates of 11.13% and 11.37%, respectively. As can be seen in Figure 6, Hsueh et al.'s approach seems to perform slightly better for smaller size networks in terms of objective values. (The detailed information for the best results is included as Table 3.) Note that for smaller size networks, each construction team can only service very few disaster nodes, usually 2~3 disaster nodes; no apparent advantage can be concluded in favor of either approach. However, Chen et al.'s approach tends to outperform for larger size networks in terms of objective value.

The advantage of Chen et al.'s approach over Hsueh et al.'s approach on Problem 12 can be further justified by varying number of construction teams. As shown in Figure 7, Chen et al.'s approach is better than Hsueh et al.'s approach in terms of objective value for the number of construction teams ranging from 4 to 15. In a separate experiment (see Table 4), we also observed that Chen et al.'s approach is better than Hsueh et al.'s approach in terms of objective value for the majority of test problems when either depot 1 or depot 2 is closed.

In Table 2, the average computational CPU times for the two approaches are 20.64 and 5.94 seconds, respectively, implying that the Chen et al.'s algorithm is computationally intensive compared to the Hsueh et al.'s method. As shown in Figure 8, both approaches indicate that the larger the network size, the higher the CPU time. However, even with the largest test network, the problem can still be solved within 3 minutes using any method. These low CPU times are not a large concern for this type of problem.

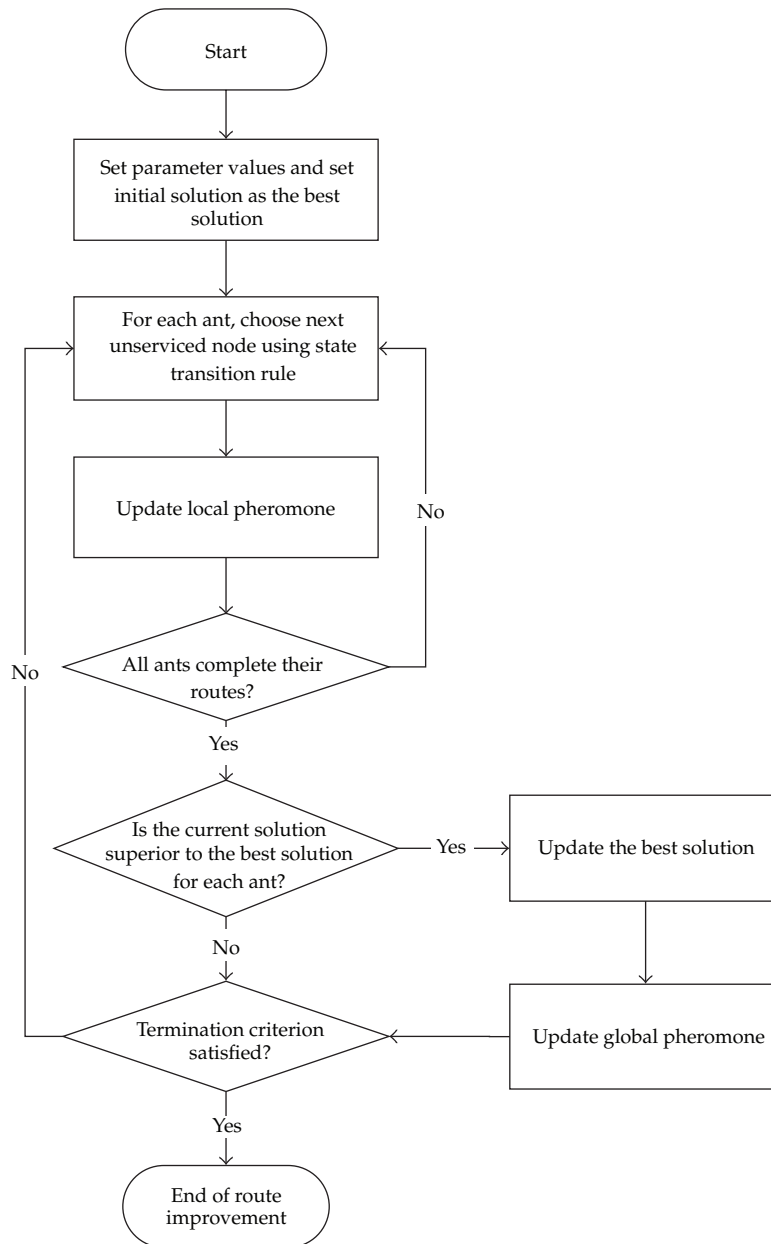


Figure 5: Flow chart for route improvement of emergency vehicle Routing.

Based on the above discussions, it is concluded that both Chen et al.'s and Hsueh et al.'s approaches are suitable for scheduling rescue operations for smaller size networks. However, for larger size networks, Chen et al.'s approach is preferred due to lower objective values. In all cases, CPU time is not a big concern.

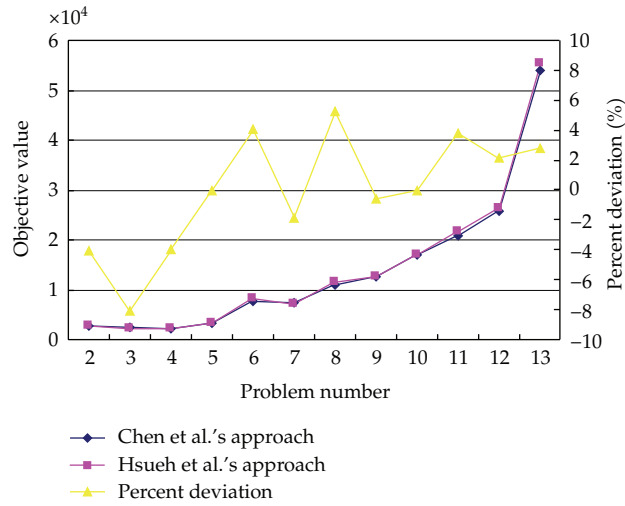


Figure 6: Objective values for the 13 test examples.

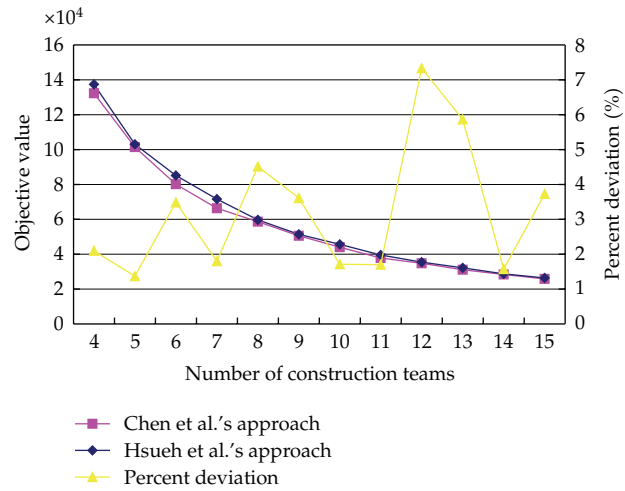


Figure 7: Objective value versus the number of construction teams for Problem 12.

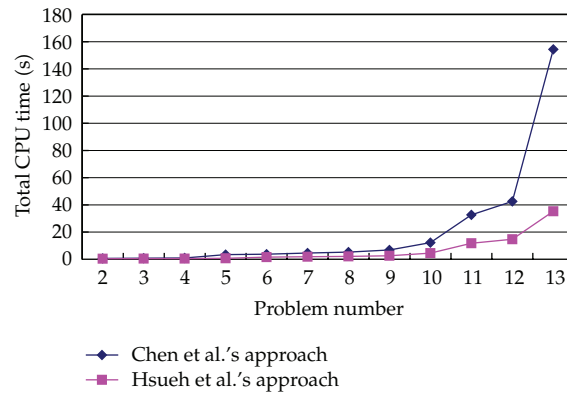


Figure 8: CPU times for the 13 test examples.

**Table 2:** The results for the test networks.

Problem no.	Nodes	Algorithm	Initial solution	Improved solution	Improved rate	CPU time (sec)		
						Initial solution	Improved solution	Total
1	10	Chen et al.	260	234	11.11	0.13	0.37	0.50
		Hsueh et al.	262	234	11.97	0.22	0.81	1.03
2	17	Chen et al.	2861	2861	0.00	0.19	0.44	0.63
		Hsueh et al.	2745	2745	0.00	0.05	0.30	0.35
3	19	Chen et al.	2570	2512	2.32	0.22	0.52	0.74
		Hsueh et al.	2570	2308	11.36	0.08	0.36	0.44
4	20	Chen et al.	2401	2324	3.31	0.28	0.61	0.89
		Hsueh et al.	2431	2231	8.96	0.08	0.41	0.49
5	23	Chen et al.	3480	3371	3.23	0.53	1.07	4.40
		Hsueh et al.	3535	3371	4.86	0.14	0.56	0.70
6	30	Chen et al.	9862	7785	26.67	1.28	2.46	3.74
		Hsueh et al.	10106	8104	24.70	0.36	1.16	1.52
7	31	Chen et al.	9070	7366	23.13	1.71	2.87	4.58
		Hsueh et al.	8376	7227	15.90	0.38	1.50	1.88
8	34	Chen et al.	12628	10840	16.50	1.76	3.52	5.28
		Hsueh et al.	11463	11408	0.48	0.38	1.68	2.06
9	37	Chen et al.	13894	12607	10.21	2.47	4.33	6.80
		Hsueh et al.	14894	12602	18.19	0.64	1.92	2.56
10	40	Chen et al.	20047	17033	17.70	5.08	7.11	12.19
		Hsueh et al.	19535	17033	14.69	1.27	3.25	4.52
11	48	Chen et al.	24154	20882	15.66	14.02	18.64	32.66
		Hsueh et al.	24608	21667	13.58	3.55	8.19	11.74
12	50	Chen et al.	28336	25787	9.89	18.45	24.09	42.54
		Hsueh et al.	28514	26339	8.26	4.56	10.13	14.69
13	70	Chen et al.	56668	53959	5.0	70.47	83.85	154.32
		Hsueh et al.	63719	55468	14.9	10.72	24.57	35.29
Average		Chen et al.	14325	12889	11.13	8.97	11.67	20.64
		Hsueh et al.	14828	13134	11.37	1.73	4.22	5.94

Remark: CPU time is recorded for Microsoft Windows XP, INTEL Pentium 4 3.0 GHz, 1 GB RAM.

## 7. Conclusion

### 7.1. Conclusion

This research addresses the damaged infrastructure repair problem as a series of multidepot VRP problem with time windows and presents 13 hypothetical test examples. In addition to link travel times, other real-time information such as new disaster nodes as well as added construction teams can also be considered in the unified framework.

**Table 3:** The best results for the test networks.

Problem no.	Nodes	Algorithm	Total travel time	Violation of time windows	Objective	Teams no.
1	10	Chen et al.	24	21	234	5
		Hsueh et al.	24	21	234	5
2	17	Chen et al.	543	232	2861	15
		Hsueh et al.	465	228	2745	15
3	19	Chen et al.	391	212	2512	15
		Hsueh et al.	379	193	2308	15
4	20	Chen et al.	419	191	2324	15
		Hsueh et al.	404	183	2231	15
5	23	Chen et al.	492	288	3371	15
		Hsueh et al.	492	288	3371	15
6	30	Chen et al.	523	726	7785	15
		Hsueh et al.	552	755	8104	15
7	31	Chen et al.	568	680	7366	15
		Hsueh et al.	554	667	7227	15
8	34	Chen et al.	725	1012	10840	15
		Hsueh et al.	772	1064	11408	15
9	37	Chen et al.	850	1183	12607	15
		Hsueh et al.	840	1176	12602	15
10	40	Chen et al.	765	1627	17033	15
		Hsueh et al.	765	1627	17033	15
11	48	Chen et al.	783	2010	20882	15
		Hsueh et al.	781	2089	21667	15
12	50	Chen et al.	729	2506	25787	15
		Hsueh et al.	724	2562	26339	15
13	70	Chen et al.	842	5312	53959	15
		Hsueh et al.	848	5462	55468	15
Average		Chen et al.	589	1231	12895	14
		Hsueh et al.	585	1255	13134	14

Remark: CPU time is recorded for Microsoft Windows XP, INTEL Pentium 4 3.0 GHz, 1 GB RAM.

The time-rolling horizon approach is appropriate for dealing with the damaged infrastructure repair problem, especially when some disaster nodes have precedent-subsequent sequential relationships. As precedent disaster nodes are repaired, subsequent disaster nodes become reachable and can be repaired.

To deal with this type of problem, we propose a heuristic, that is, Chen et al.'s, approach and include an outside source, that is, Hsueh et al.'s (2008) approach (with necessary modifications) for comparison. The former embeds ACS which can efficiently improve the initial solution generated by the minimum cost insertion method, with an average improvement rate of 11.13% on 13 numerical examples.

The experiments show that both Chen et al.'s and Hsueh et al.'s approaches are suitable for scheduling rescue operations. However, Chen et al.'s approach is slightly preferred due to lower objective values. In all cases, CPU time is not a big concern.

**Table 4:** The results for the test networks with a single open depot.

Problem no.	Nodes	Algorithm	Initial solution	Improved solution	Improved rate	CPU time (sec)		
						Initial solution	Improved solution	Total
1	10	Chen et al.	1449	1229	17.9	0.37	0.87	1.25
		Hsueh et al.	1531	1229	24.6	0.25	0.87	1.12
2	17	Chen et al.	5998	5022	19.4	0.75	1.75	2.50
		Hsueh et al.	6291	5022	25.3	0.31	1.28	1.59
3	19	Chen et al.	7153	5608	27.6	0.84	1.84	2.68
		Hsueh et al.	7721	5972	29.3	0.47	2.03	2.50
4	20	Chen et al.	8211	6445	27.4	0.94	2.18	3.12
		Hsueh et al.	8611	6445	33.6	0.37	1.44	1.81
5	23	Chen et al.	10469	8155	28.4	1.18	2.53	3.71
		Hsueh et al.	10304	8155	26.3	0.56	1.84	2.40
6	30	Chen et al.	24035	19704	22.0	2.25	4.02	6.27
		Hsueh et al.	24099	19704	22.3	0.90	2.71	3.62
7	31	Chen et al.	22848	19492	17.2	2.68	6.02	8.70
		Hsueh et al.	21749	19555	11.2	1.09	3.28	4.37
8	34	Chen et al.	29105	26149	11.3	2.93	5.40	8.33
		Hsueh et al.	30470	26380	15.5	0.87	3.06	3.93
9	37	Chen et al.	33887	28717	18.0	4.27	6.71	10.98
		Hsueh et al.	33934	28750	18.0	0.94	4.15	5.08
10	40	Chen et al.	42466	38280	10.9	6.68	9.89	16.57
		Hsueh et al.	42839	38292	11.9	1.44	5.71	7.15
11	48	Chen et al.	48422	47785	1.3	13.82	20.87	34.69
		Hsueh et al.	58266	51758	12.6	2.43	9.33	11.76
12	50	Chen et al.	58442	57703	1.3	17.81	24.65	42.46
		Hsueh et al.	71856	64206	11.9	2.62	9.92	12.54
13	70	Chen et al.	122340	120602	1.4	64.10	76.92	141.02
		Hsueh et al.	141277	126829	11.4	5.34	17.25	22.59
Average		Chen et al.	31910	29607	15.7	9.13	12.59	21.71
		Hsueh et al.	35304	30946	19.5	1.35	4.84	6.19

Remark: CPU time is recorded for Microsoft Windows XP, INTEL Pentium 4 3.0 GHz, 1 GB RAM.

## 7.2. Suggestions

Some important issues are listed as follows.

- (1) Because each construction team is equipped with different machines and tools that may not meet the requirements of all disaster nodes, a heterogeneous VRP problem may be more appropriate.
- (2) For short-term planning, the location of depots is fixed. However, the overall efficiency of the damaged infrastructure repair problem can be increased if the location is allowed to change.

- (3) If spatial information can be displayed on a computer screen, then the decision maker can make more reasonable decisions about the routing schedule of construction teams. For real-world applications, a screen-based decision support system using the best proposed method along with the geographic information system (GIS) is worth further exploring. This is because the geographical characteristics of disaster nodes as well as the damage conditions associated with these disaster nodes can be elaborated.

## Acknowledgment

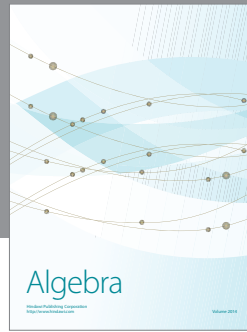
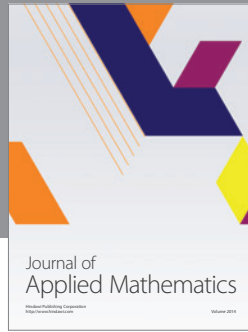
The authors would like to thank the National Science Council, Taiwan, for financially supporting part of this work.

## References

- [1] Central Weather Bureau Ministry of Transportation and Communications, (Taiwan), "Scale of seismic intensity," 2000, <http://www.cwb.gov.tw/> .
- [2] Directorate-General of Budget, Accounting and Statistics, Executive Yuan, (Taiwan), 1999, <http://www.dgbas.gov.tw/> .
- [3] FEMA, *Guide for All-Hazard Emergency Operation Planning-State and Local Guide (SIG-101)*, 1996.
- [4] F. Fiedrich, F. Gehbauer, and U. Rickers, "Optimized resource allocation for emergency response after earthquake disasters," *Safety Science*, vol. 35, no. 1–3, pp. 41–57, 2000.
- [5] L. W. Chang, *Emergency scheduling for repairing damaged infrastructures after disasters*, M.S. thesis, Graduate Institute of Transportation Science, Tamkang University, Taipei, Taiwan, 2001.
- [6] G. Barbarosoglu, L. Özdamar, and A. Çevik, "An interactive approach for hierarchical analysis of helicopter logistics in disaster relief operations," *European Journal of Operational Research*, vol. 140, no. 1, pp. 118–133, 2002.
- [7] C. H. Lee, *Using the genetic algorithm to solve the scheduling problem for repairing the post earthquake road networks*, M.S. thesis, Department of Transportation Management, National Cheng-Kung University, Tainan, Taiwan, 2003.
- [8] L. Özdamar, E. Ekinci, and B. Küçükyazici, "Emergency logistics planning in natural disasters," *Annals of Operations Research*, vol. 129, no. 1–4, pp. 217–245, 2004.
- [9] H. K. Liao, *A study on network reconstruction and relief logistics*, M.S. thesis, National Central University, Jungli, Taiwan, 2005.
- [10] S. J. Pettit and A. K. C. Beresford, "Emergency relief logistics: an evaluation of military, non-military and composite response models," *International Journal of Logistics: Research and Applications*, vol. 8, no. 4, pp. 313–332, 2005.
- [11] E. Jennings, A. K. C. Beresford, and R. Banomyong, "Emergency relief logistics: a disaster response model," in *Proceedings of the Logistics Research Network Conference*, pp. 121–128, 2002.
- [12] C. F. Hsueh, H. K. Chen, and H. W. Chou, "Vehicle routing for rescue operation in natural disasters," in *Vehicle Routing Problem*, T. Caric and H. Gold, Eds., pp. 71–84, I-Tech Education and Publishing KG, Vienna, Austria, 2008.
- [13] F. S. Wang, *Scheduling for repairing damaged networks under imprecise information in earthquakes*, M.S. thesis, Department of Civil Engineering, National Central University, Chung-Li, Taiwan, 2008.
- [14] H. K. Chen, C. F. Hsueh, and M. S. Chang, "The real-time time-dependent vehicle routing problem," *Transportation Research Part E*, vol. 42, no. 5, pp. 383–408, 2006.
- [15] M. Dorigo, V. Maniezzo, and A. Colorni, "The ant system: an autocatalytic optimizing process," *Tech. Rep.*, Politecnico di Milano, Milano, Italy, 1991.
- [16] M. Dorigo, V. Maniezzo, and A. Colorni, "The ant system: optimization by a colony of cooperating agents," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 26, no. 1, pp. 29–41, 1996.
- [17] M. Dorigo and L. M. Gambardella, "Ant colony system: a cooperative learning approach to the traveling salesman problem," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 53–66, 1997.



- [18] T. Stützle and H. Hoos, "The MAX-MIN ant system and local search for the traveling salesman problem," in *Proceedings of the IEEE International Conference on Evolutionary Computation (ICEC '97)*, pp. 309–314, April 1997.
- [19] B. Bullnheimer, R. F. Hartl, and C. Strauss, "A new rank-based version of the ant system: a computational study," Tech. Rep. POM 3, Institute of Management Science, University of Vienna, Vienna, Austria, 1997.
- [20] E. D. Taillard and L. M. Gambardella, "Adaptive memories for the quadratic assignment problem," Tech. Rep., IDSIA, 1997.
- [21] V. Maniezzo, "Exact and approximate nondeterministic tree-search procedures for the quadratic assignment problem," Tech. Rep. CSR 1, C.L. in Scienze dell'Informazione, Università di Bologna, Sede di Cesena, Cesena, Italy, 1998.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

