Computers and Operations Research 105 (2019) 237–247

Contents lists available at ScienceDirect

Computers and Operations Research

journal homepage: www.elsevier.com/locate/cor

Integrated optimal scheduling of repair crew and relief vehicle after disaster

Youngchul Shin[a], Sungwoo Kim[b], Ilkyeong Moon[c,*]

[a] Department of Industrial Engineering, Seoul National University, 1, Gwanak-ro, Gwanak-gu, Seoul, Korea
[b] H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology College of Engineering, North Ave NW, Atlanta, GA 30332, USA
[c] Department of Industrial Engineering and Institute for Industrial Systems Innovation, Seoul National University, 1, Gwanak-ro, Gwanak-gu, Seoul, Korea

ARTICLE INFO

Article history:
Received 5 March 2018
Revised 28 January 2019
Accepted 28 January 2019
Available online 1 February 2019

Keywords:
Repair crew
Humanitarian logistics
Relief scheduling
Mixed integer linear programming
Network repair

ABSTRACT

When catastrophes occur, certain areas could be isolated and have more hardship in being provided with the relief goods because of the destruction of roads. Additional damages will occur if isolated areas cannot be supplied relief goods quickly. Therefore, to provide adequate relief goods to the demand area in time, it is essential to recover the destroyed roads and make the isolated areas accessible. In addition, relief goods should be distributed on the basis of optimal scheduling. If the order for repairing destroyed roads and supplying relief goods to an isolated area is established from the respective viewpoints, a delay will occur compared to optimal schedule considering both of them simultaneously. Thus, this study proposes a mathematical model based on Mixed Integer Linear Programming (MILP) and an Ant Colony Optimization (ACO) algorithm providing optimal scheduling by taking into account both reconstruction and delivery. By conducting various numerical experiments, the solution from the MILP and ACO could lead to more effective results than decisions based on two points of view that are not synchronized. Some meaningful findings are also found by numerous computational experiments. This research can help to make the optimal decision for scheduling of recovery and supply that thus reduces any additional damage after a major disaster.

1. Introduction

From the past to the present, various types of disasters, such as the Haitian earthquake of 2010 and the Japanese tsunami of 2011, have brought devastation throughout the world. These disasters cause the world suffering from large human casualties and significant property damage, and they result in a huge amount of recovery cost. These circumstances suggest that research on the response to disasters is needed. Although preparing for disasters in advance is critical, because of the unpredictable and time-varying nature of disasters, it is practically impossible to completely prepare in advance. Therefore, it is necessary to prepare a reactive response that minimizes additional damages after a disaster.

When a disaster occurs, various government agencies, such as armed forces and firefighters, get involved in the disaster areas to deal with damage through their respective roles. Because of the various types of stakeholders involved, several relief activities are in operation at the same time such that integrated decision should be made within a short time after the disaster. In addition, there are many conflicting factors, such as labor costs, goods, and time windows appropriate for the distribution of aid. In rapidly changing and complex situations, the decision of the relevant manager may directly affect the additional damages in the affected areas. Despite the important role they play, the critical momentary decisions are often made on the basis of personal experiences or intuition. However, with many conflicting interests to consider simultaneously and immediately, all points of view are difficult to see, and as a result, individual judgments may reflect biases. However, by making scientific decisions based on data, managers might better incorporate all conflicting elements in making optimal decisions.

To cope with these complex disaster situations, various types of studies have been carried out in the field of industrial engineering, operations research, management science, and computer science, among others (Altay and Green 2006; Caunhye et al. 2012). Especially, to reduce further damage in the post disaster period, the research in the following areas has been conducted: evacuation planning, relief distribution, casualty transportation, and facility location problems (Hamacher and Tjandra 2001; Osman and Ram 2017; Sheu 2007; Rath and Gutjahr 2014; Yi and Kumar 2007;

* Corresponding author.
E-mail addresses: chulbee@snu.ac.kr (Y. Shin), asdf1358@gmail.com (S. Kim), ikmoon@snu.ac.kr (I. Moon).

https://doi.org/10.1016/j.cor.2019.01.015
0305-0548/

Talarico et al., 2015; Balcik and Beamon 2008; Rancourt et al., 2015). However, Holguin-Veras et al. (2012) pointed to insufficient research on responses after disasters. They claimed that the low frequency of disasters, little attention from academia, and operations of short duration are the main reasons for lack of research. Galindo and Batta (2013) also explained that research efforts dealing with recovery after disasters have not received sufficient attention compared to research focusing on the distribution of relief goods.

According to the studies cited in the previous paragraph, research dealing with post-disaster response remains insufficient. Therefore, additional research on responses after a disaster is needed, specifically those that focus on minimizing additional damage or recovery time for destroyed areas through efficient recovery operation systems. The repair crew problem deals with damage control efforts by deciding the sequence for repairing destroyed roads to reconnect supply chain networks. Especially in rural areas, where road networks are sparse and supply chains are limited, the road-network repair should be implemented immediately after man-made or natural disasters because the road destruction can contribute to a high possibility that people will be completely isolated (Kim et al., 2018). When roads are destroyed and certain areas are isolated, the areas will become difficult to receive relief goods such as emergency foods, medicines, and drinking water. Many people in isolated areas also face challenges in evacuating to safe areas. Therefore, the destroyed roads need to be repaired as quickly as possible so that isolated areas are accessible and people receive aid items. In the post-disaster state, if sufficient manpower and resources are available, all destroyed roads can be repaired at the same time. In these cases, the order of operations for repairing destroyed roads or transporting relief goods does not have much effect in terms of additional damages. However, when resources or manpower is limited, the destroyed roads must be repaired according to a specific indicator, such as priority location. The schedule for repairing roads should be made through scientifically derived criteria, such as mathematical models and heuristic algorithms, rather than the intuition of the decision makers, and this study addresses possible operations to use when manpower for road repair is lacking.

In a previous study on the repair of a destroyed network, Aksu and Ozdamar (2014) developed a mathematical model to maximize network accessibility for the evacuation of survivors and the removal of roadside debris. They also developed an algorithm to solve recovery problems in realistically sized networks within a reasonable time. Yan and Shih (2012) developed an integer network flow model with side constraints, to deal with the roadway repair problem, and an ant colony system based hybrid algorithm, to solve the large-scale data efficiently. Duque et al. (2013) proposed a problem for improving path accessibility by developing a knapsack problem based heuristic and a Variable Neighborhood Search (VNS). They also verified the performance through examples with 480 random generations. Avci and Avci (2017) developed the Greedy Randomized Adaptive Search Procedure (GRASP) to solve efficiently a traveling repairman problem with profits. Kim et al. (2017) proposed a repair crew scheduling problem by considering time-varying aspects of the disaster.

In case of research that did not address the sequence of repair after a disaster, the focus was mainly on network restoration problems with allocation of limited resources. Karlaftis et al. (2007) pointed out that the importance of resource allocation is undervalued in the existing disaster recovery research and suggested a resource allocation method for road destruction recovery in urban areas. Duque and Sörensen (2011) considered resource allocation issues for repair of disaster-affected rural road networks. They developed GRASP and VNS metaheuristic to maximize the accessibility of the socio-economic infrastructure. They also conducted a case study on the 2008 Haitian disaster. Chou et al. (2014) developed Biological-based Genetic Algorithms (BGA) dealing with resource allocation problems in disaster response. The BGA developed by Chou et al. (2014) outperformed other solution procedures in cases of the allocation problem. Zhou et al. (2017) developed the emergency resource scheduling problem for a case of multiple periods in which unsatisfied demand and risk for choosing the destroyed road were minimized and optimal roads to rescue were efficiently selected.

Most of the studies related to the repair crew problem, however, mainly focused on reducing the total time spent to convert inaccessible areas to accessible ones. To minimize additional damages, it is critical to consider not only the repair operations in inaccessible areas but also the scheduling of relief goods delivery to the areas in need (Yan and Shih 2009). If repair and delivery are not considered at the same time, then the operations feature inefficiencies either in the schedule for repairing the disrupted roads or in the transporting of the relief goods to the areas of demand. The delivery of relief goods depends on the schedule of the repair crew because transportation can only be possible when a destroyed road becomes available. Therefore, the repair crew must be assigned on the basis of a schedule developed from a global perspective precisely because the operations, such as transportation of relief goods, depends on the timing of the repairs. These related features of disaster response have been overlooked, and as a result, few mathematical models reflect these characteristics.

To address the lack of research in the area of simultaneous repair of roads and goods delivery, a mathematical model based on Mixed Integer Linear Programming (MILP) for Integrated Optimal Scheduling of Repair Crew and Relief Vehicle after Disaster (IOSRCRVD) is developed. According to the IOSRCRVD, destroyed roads are repaired by a *repair crew* and demand areas receive relief goods by a *relief vehicle*. The repair crew and the relief vehicle are considered as one large entity, respectively. The solution obtained by MILP provides an optimal schedule and route for the repair crew and relief vehicle. The mathematical model considering the repair crew problem developed in this research was influenced by the working paper of Duque et al. (2014). The mathematical model was developed according to the Traveling Salesman Problem (TSP) in a graph $G=(V,\ E)$. Unlike a traditional TSP, the model has the property of the Steiner TSP such that the repair crew or relief vehicle can visit each node more than once. The model also features the property of a Selective TSP such that only specific nodes, not all nodes, can be visited. Because of these model properties, the complexity of the problem is very high. Duque et al. (2016) claimed that the proposed mathematical model is intractable even for the small-scale data and did not include the model in their paper; rather, instead of presenting a mathematical model, they proposed Dynamic Programming for small- and medium-scale data and the Iterated Greedy-Randomized Constructive Procedure for the large-scale data. For the research presented herein, the model proposed by Duque et al. (2016) is extended with the added concept of the transportation of relief goods. An Ant Colony Optimization (ACO) algorithm is also developed to solve the problem efficiently.

In a previous study from Duque et al. (2016), the objective function was expressed as the product of the weights, each denoted as w_i, for each demand node and the times that the demand nodes become accessible. The weight of each demand node was considered as the number of isolated inhabitants. In this study, the objective function is developed to minimize the last completed transportation time to all the demand nodes. Unlike commercial companies seeking to maximize profit in accordance with the needs determined by local perspectives, in the case of disaster situations, it seems to critical to minimize the completion time for all aid distribution to isolated areas from the overall perspective. If the optimal solution is obtained by the mathematical model with unequal

priority given to each demand area, then the time to complete the last delivery of relief goods in isolated areas is delayed because of the biased weights. In addition, by calculating the minimum total time to respond to a disaster, the situation can be analyzed from an overall perspective. From this macroscopic view, decision makers can also determine additional details or consider alternatives. Therefore, in this study, the objective function is set to minimize the last completion time for the transportation of relief goods.

The remainder of the paper is structured as follows. Section 2 presents explanations on the MILP of the IOSRCRVD, and Section 3 explains the ACO algorithm. Section 4 describes the computational experiments and analysis. Section 5 summarizes the findings of this research.

2. Mathematical model

2.1. Problem description

This study focuses on disaster response in a rural area where the roads between each region are sparse, rather than in an urban area where most areas are densely populated and physically connected through multiple routes. Where connections between the regions are many, the disaster response can be accomplished by receiving relief goods and transferring victims via detours on undamaged roads. In this urban case, conducting disaster relief research with a resource allocation problem or resource scheduling would be more appropriate. However, even in rural areas, disasters can destroy key roads and leave some areas isolated. That is, responses to road repair and relief goods delivery are critical to mitigate the disaster aftermath in any area that may be isolated, and an integrated schedule that accounts for both road repair and relief delivery simultaneously is needed. For this study, a mathematical model is proposed to determine the order for repairing destroyed roads and transporting the relief items to demand areas after a disaster occurs in a rural area with a limited number of connecting roads. Because many areas are assumed inaccessible, the possible starting points for the repair crew and relief vehicle are also limited. Because the relief vehicle can pass only on roads left un-destroyed or after repair, the relief vehicle should start operations from a location near that of the repair crew. For these reasons, the repair crew and relief vehicle are presumed to start from the same depot.

When defining a problem dealing with multiple regions and the roads between them in terms of a mathematical model, the components can be represented by a network model. The network model in this study is defined as $G=(V, E)$ in which V is a set of nodes representing regions, and E is a set of arcs that symbolize the roads connecting the nodes. Node set V includes a V_d set of demand nodes for relief goods and V_r is a set of destroyed nodes that require repair. A depot is a location from which a repair crew and relief vehicle depart, and a dummy node is designated as the spot in which both the crew and vehicle finish the repair and relief tour. Figs. 1 and 2 show the simple example of the network considered in this study. Fig. 1 presents an example of a disaster that destroys edges between Nodes 0 and 2, 1 and 2, and 0 and 3. As a result of the disaster and destroyed edges, Nodes 2 and 3 are isolated. Fig. 2 shows that the destroyed edges are transformed into Nodes 4–6.

In this network model, a repair crew must decide the order in which to repair Nodes 4–6. This order can be determined by a mathematical model that accounts for the demand at each demand node and the travel times between each node. However, to minimize additional damage, the scheduling of the relief vehicle transporting the relief goods must be considered at the same time that the schedule for repairs is decided. Although one of the purposes of repairing destroyed roads is to move injured persons to a safe area, supplying relief goods to the required area must also be done as quickly as possible. However, if the sequences of repairing the destroyed roads and of transporting relief goods are considered separately, then the repair and relief are inefficiently completed; therefore, the two schedules must be integrally determined.

To develop the MILP that allows for simultaneously considered concerns of both repair and relief, the following assumptions are made:

(1) A repair crew goes through the repair work together. In other words, the crew is considered one large entity.
(2) Demands, travel times, and times required to repair destroyed areas are deterministic.
(3) A repair crew cannot move between nodes that are not connected by an edge.
(4) A repair crew can visit nodes and go through edges multiple times.
(5) A relief vehicle has sufficient capacity for the supply goods during a single tour.
(6) Demands for relief goods at nodes are deterministic.
(7) After the repair crew repairs a destroyed node, the relief vehicle can go through the node.
(8) A relief vehicle can visit nodes and go through edges multiple times.

2.2. Mathematical formulation

According to the assumptions, a mathematical model formulated as an MILP for the IOSRCRVD is developed. Sets, parameters, and decision variables of the mathematical model are as follows:

Sets

V	Set of nodes
V_d	Set of demand nodes that need relief goods
V_r	Set of repair (destroyed) nodes that require repair

Parameters

w_i	Relief demand in node i, $\forall i \in V_d$
t_{ij}	Travel time between nodes i and j, $\forall i,\ j \in V$
s_i	Time required to repair a destroyed node, $i \in V_r$
s'_i	Time required to supply relief good at a demand node, $i \in V_d$

Decision variables

x_{ij}	Order in which the destroyed nodes are repaired by a repair crew = 1, node j is repaired immediately after node i = 0, otherwise $\forall i,\ j \in V_r \cup \{0, n+1\}$
r_{ijlm}	= 1, edge between l and m is used on the path from i to j by a repair crew = 0, otherwise $\forall i,\ j \in V_r \cup \{0\}, \forall l,\ m \in V$
a_{ijk}	= 1, node k is visited on the path from i to j by a repair crew = 0, otherwise $\forall i, j \in V_r \cup \{0\}, \forall k \in V$
y_{0ilm}	= 1, edge between l and m is used on the path from the depot to node i = 0, otherwise $\forall i \in V_d, \forall l,\ m \in V$
b_{0ik}	= 1, node k is visited on the path from the depot to node i = 0, otherwise $\forall i \in V_d, \forall k \in V$
d_i^{arr}	Time at which the repair crew arrives at destroyed node $i\ \forall i \in V_r \cup \{0\}$
d_i	Time at which the destroyed node i is repaired $\forall i \in V_r \cup \{0\}$
f_i	Time at which the demand node i is accessible $\forall i \in V_d \cup \{0\}$
z_{ij}	Order in which the demand nodes are visited by a relief vehicle = 1, node j is visited immediately after node i = 0, otherwise $\forall i,\ j \in V_d \cup \{0, n+1\}$
e_{ijlm}	= 1, edge between l and m is used on the path from i to j by relief vehicle = 0, otherwise $\forall i,\ j \in V_d \cup \{0\}, \forall l,\ m \in V$

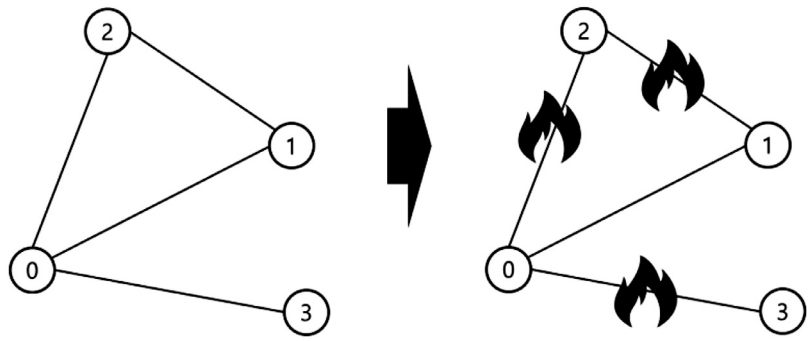

Fig. 1. Network representing destroyed roads.

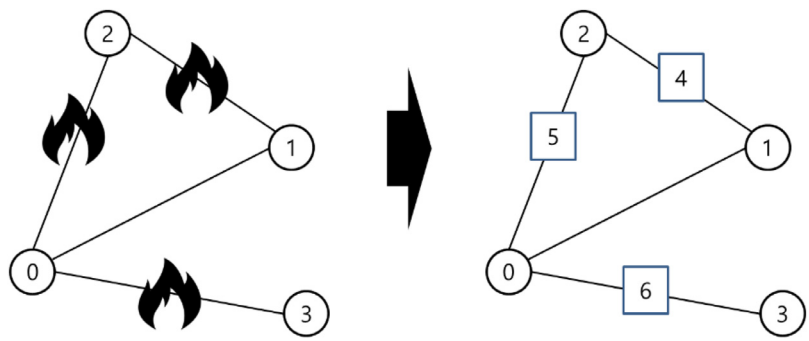

Fig. 2. Network that transforms the destroyed roads into the nodes.

c_{ijk} = 1, node k is visited on the path from i to j by relief vehicle = 0, otherwise $\forall i,\ j \in V_d \cup \{0\}, \forall k \in V$

h_i^{arr} Time at which the relief vehicle arrives demand node $i\ \forall i \in V_d \cup \{0\}$

h_i Time at which the relief goods are supplied at demand node $i\ \forall i \in V_d \cup \{0\}$

f' Last time that the relief service is conducted

The objective function and constraints of the MILP and explanations of them are as follows:

Objective function

$$minimize\ f' \tag{1}$$

Subject to

$$\sum_{i \in V_r \cup \{0,\ n+1\}} x_{ij} = 1 \quad \forall j \in V_r \cup \{0,\ n+1\} \tag{2}$$

$$\sum_{j \in V_r \cup \{0,\ n+1\}} x_{ij} = 1 \quad \forall i \in V_r \cup \{0,\ n+1\} \tag{3}$$

$$\sum_{k \in succ(i)} r_{ijik} = x_{ij} \quad \forall i,\ j \in V_r \cup \{0\} \tag{4}$$

$$\sum_{k \in pred(j)} r_{ijkj} = x_{ij} \quad \forall i,\ j \in V_r \cup \{0\} \tag{5}$$

$$\sum_{l \in succ(k)} r_{ijkl} + \sum_{l \in pred(k)} r_{ijlk} = 2a_{ijk} \quad \begin{array}{l} \forall i,\ j \in V_r \cup \{0\} \\ \forall k \in V \backslash \{i,\ j\} \end{array} \tag{6}$$

$$\sum_{k \in V_d} y_{0j0k} = 1 \quad \forall j \in V_d \tag{7}$$

$$\sum_{k \in V_d} y_{0jkj} = 1 \quad \forall j \in V_d \tag{8}$$

$$\sum_{l \in succ(k)} y_{0jkl} + \sum_{l \in pred(k)} y_{0jlk} = 2b_{0jk} \quad \begin{array}{l} \forall j \in V_d \\ \forall k \in V \backslash \{0,\ j\} \end{array} \tag{9}$$

$$d_i + \sum_{l,m \in V} r_{ijlm} t_{lm} + M\left(x_{ij} - 1\right) \le d_j^{arr} \quad \forall i,\ j \in V_r \cup \{0\} \tag{10}$$

$$d_j = d_j^{arr} + s_j \quad \forall j \in V_r \tag{11}$$

$$d_k + M\left(a_{ijk} - 1\right) \le d_j \quad \begin{array}{l} \forall i,\ j \in V_r \cup \\ \forall k \in V_r \end{array} \tag{12}$$

$$d_k + M(b_{0ik} - 1) \le f_i \quad \begin{array}{l} \forall i \in V_d \\ \forall k \in V_r \end{array} \tag{13}$$

$$\sum_{i \in V_d \cup \{0,\ n+1\}} z_{ij} = 1 \quad \forall i \in V_d \cup \{0,\ n+1\} \tag{14}$$

$$\sum_{j \in V_r \cup \{0,\ n+1\}} z_{ij} = 1 \quad \forall j \in V_d \cup \{0,\ n+1\} \tag{15}$$

$$\sum_{k \in succ(i)} e_{ijik} = z_{ij} \quad \forall i,\ j \in V_d \cup \{0\} \tag{16}$$

$$\sum_{k \in pred(j)} e_{ijkj} = z_{ij} \quad \forall i,\ j \in V_d \cup \{0\} \tag{17}$$

$$\sum_{l \in succ(k)} e_{ijkl} + \sum_{l \in pred(k)} e_{ijlk} = 2c_{ijk} \quad \begin{array}{l} \forall i,\ j \in V_r \cup \{0\} \\ \forall k \in V \backslash \{i,\ j\} \end{array} \tag{18}$$

$$h_i + \sum_{l,m \in V} e_{ijlm} t_{ij} + M\left(z_{ij} - 1\right) \le h_j^{arr} \quad \forall i,\ j \in V_d \cup \{0\} \tag{19}$$

$$h_j = h_j^{arr} + s'_j \quad \forall j \in V_d \tag{20}$$

$$h_k + M\left(c_{ijk} - 1\right) \le h_j \quad \begin{array}{l} \forall i,\ j \in V_d \cup \{0\} \\ \forall k \in V_d \end{array} \tag{21}$$

$$h_j^{arr} \ge f_j + t_{kj} + M * (c_{ijk} - 1) \quad \forall i, j \in V_d \cup \{0\},\ \forall k \in V \tag{22}$$

$$d_j \le h_i + t_{ij} + M * \left(1 - e_{ikij}\right) \quad \forall i, k \in V_d \cup \{0\},\ \forall j \in V_r \cup \{0\} \tag{23}$$

$$h_j \le f' \quad \forall k \in V_d \cup \{0\} \tag{24}$$

$$x_{ij} \in \{0,\ 1\} \quad \forall i,\ j \in V_r \cup \{0\} \cup \{n+1\} \tag{25}$$

$$r_{ijlm} \in \{0,\ 1\} \quad \begin{array}{l} \forall i,\ j \in V_r \cup \{0\} \\ \forall l, m \in V \end{array} \tag{26}$$

$$a_{ijk} \in \{0,\ 1\} \quad \begin{array}{l} \forall i,\ j \in V_r \cup \{0\} \\ \forall k \in V \end{array} \tag{27}$$

$$y_{0ilm} \in \{0,\ 1\} \quad \begin{array}{l} \forall i \in V_d \\ \forall l, m \in V \end{array} \tag{28}$$

$$b_{0ik} \in \{0,\ 1\} \quad \begin{array}{l} \forall i \in V_d \\ \forall k \in V \end{array} \tag{29}$$

$$z_{ij} \in \{0,\ 1\} \quad \forall i,\ j \in V_d \cup \{0\} \cup \{n+1\} \tag{30}$$

$$e_{ijlm} \in \{0,\ 1\} \quad \begin{array}{l} \forall i,\ j \in V_d \cup \{0\} \\ \forall l, m \in V \end{array} \tag{31}$$

$$c_{ijk} \in \{0,\ 1\} \quad \begin{array}{l} \forall i,\ j \in V_d \cup \{0\} \\ \forall k \in V \end{array} \tag{32}$$

$$h_j \ge 0 \quad \forall j \in V_d \tag{33}$$

$$h_j^{arr} \ge 0 \quad \forall j \in V_d \tag{34}$$

$$d_j \ge 0 \quad \forall j \in V_r \tag{35}$$

$$d_j^{arr} \ge 0 \quad \forall j \in V_r \tag{36}$$

$$f_j \ge 0 \quad \forall j \in V_r \tag{37}$$

$$f' \ge 0 \tag{38}$$

Objective function (1) minimizes the last transportation time of the entire demand nodes. Constraints (2) and (3) ensure that each destroyed node is repaired exactly once by a repair crew. Constraint (4) enforces exactly one edge for the repair crew leaving i on the path from i to j. Constraint (5) enforces exactly one edge for the repair crew entering j on the path from i to j. Constraint (6) ensures that the path from i to j traveled by a repair crew is connected. Constraints (7) and (8) help track the path that connects the depot to every demand node. Constraint (9) ensures that the path from the depot to demand node j is connected. Constraint (10) means that d_j^{arr} is the cumulative sum of the travel times from i to j for a repair crew's first arrival time to destroyed node j. In Constraint (11), d_j is the sum of the repair crew's first arrival time to the destroyed node j and the repair time for destroyed node j. Constraint (12) means that if destroyed node k is on the path from destroyed node i to j, node k is repaired before ode j is repaired. In Constraint (13), f_j represents the accessible time for visiting demand node i. Constraints (14) and (15) ensure that each demand node is visited exactly once by a relief vehicle. Constraint (16) enforces exactly one edge for the relief vehicle leaving i on the path from i to j. Constraint (17) enforces exactly one edge for the relief vehicle entering j on the path from i to j. Constraint (18) ensures that the path from i to j traveled by a relief vehicle is connected. Constraint (19) shows that h_j^{arr} is the cumulative sum of the travel times from i to j. Constraint (20) indicates that h_j is the sum of h_j^{arr} and the relief service time. In Constraint (21), h_j is the time at which demand node j receives the relief goods from the relief vehicle. Constraints (22) and (23) ensures the time window for relief vehicle that repair of the destroyed node must precede the path of relief vehicle. Constraint (22) guarantees that the arrival time of the relief vehicle to demand node j is greater than the sum of the times that demand node j is accessible and travel time from adjacent nodes that caused isolation. The time when the demand node becomes accessible is denoted f_j. In other words, f_j is related to the time that the nearest destroyed node is repaired. Hence, the relief vehicle can arrive at the demand node after the nearest destroyed node has been repaired such that the travel time from the nearest destroyed node is considered in the overall delivery time. Constraint (23) guarantees that after the destroyed node is repaired, the relief vehicle can pass through the node. If the relief vehicle completes the supply at node i and moves to the next demand node, k, by passing through node$j$ serving as an intermediate pathway, then node j must be repaired before the relief vehicle passes. In Constraint (24), f' is the last time of the relief service time. Constraints (25) – (32) indicate that each variable is binary. Constraints (33)–(38) indicate each variable has a nonnegative condition.

2.3. Numerical example

To verify the solution of the MILP, a numerical experiment solving the small-scale data was conducted. Fig. 3 shows the network data for the numerical experiment (Kim et al., 2017). Node ss refers to the depot from which a repair crew and a relief vehicle depart. Nodes 1, 4, and 6 represent isolated areas that need relief goods. Nodes 2, 3, 5, 7, 9, and 10 denote destroyed areas that require repair. The numbers written on the edges between the nodes refer to travel times. The optimal solution provided by the MILP that took into account both the restoration order of the destroyed roads and the supply order of relief goods simultaneously yielded a better solution than the MILP that took into account each perspective distinctly. The MILP that addresses each party of interest solves the problem in a two-stage approach. In the first stage, according to Duque et al. (2016), the repair sequence by the repair crew for the destroyed nodes is determined. In the second stage, after the repair times of the destroyed nodes are set as time windows for the relief vehicle, the order for the transportation of relief goods to the demand nodes is determined. Table 1 shows the results of the experiment based on the data presented in Fig. 3.

As can be seen in Table 1, the IOSRCRVD required 46 periods to transport relief goods to all the demand nodes. In contrast, the two-stage approach needed 58 periods to complete the deliveries. Although the IOSRCRVD showed a longer time to repair all the destroyed nodes, the last transportation time of the relief goods to the all demand nodes was less than 20.19% that of the two-stage model. As can be observed in the above example, if only the repair crew is considered, all demand nodes become accessible quickly; however, the completion time for delivery of relief goods is delayed. The main reason for making isolated regions accessible as quickly as possible is to provide relief goods to people in an isolated region. If the schedule of the repair crew is determined without consideration of the routing of the relief goods, a longer completion time for the transportation of relief goods than found for the integrated decision would result. In other words, the repair crew problem should be considered with the transportation of re-

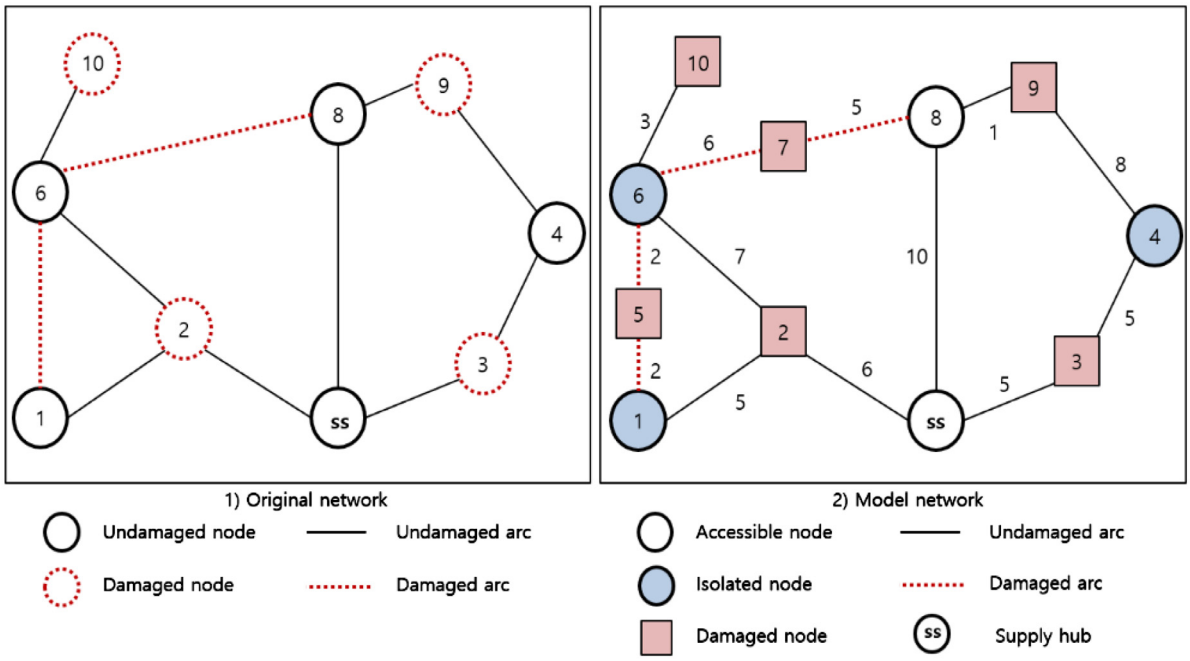

Fig. 3. Numerical example of a repair crew and relief vehicle.

Table 1
Comparing the two-stage approach MILP and the IOSRCRVD.

	Sequence of repair crew repairing destroyed nodes and the vehicle transporting relief goods (Final period for repairing the destroyed nodes and delivering the relief goods to demand nodes)	
	Two-stage approach MILP	IOSRCRVD
Repair crew	0(0)-2(12)-3(28)-9(48) -7(56)-5(67)-10(76)	0(0)-3(10)-2(27)-5(37) -7(59)-9(72)-10(103)
Relief vehicle	0(0)-6(21)-1(35)-4(58)	0(0)-4(17)-1(40)-6(46)

lief goods simultaneously to complete the transportation of relief within the shortest time.

3. Ant colony optimization algorithm

The MILP for the IOSRCRVD developed in this paper provides optimal solutions. However, because of the high complexity of the problem, it is intractable for large-scale data. To overcome this limitation, an ant colony optimization (ACO) algorithm is developed. The ACO algorithm is one of the meta-heuristic algorithms that is based on a process of the natural world. It was first proposed by Dorigo and Gambardella (1997). It features a probabilistic-based solution procedure mainly used to find the optimal path or tour in a network problem. According to Dorigo and Gambardella (1997a,b) and Dorigo and Di Caro (1999), the ACO algorithm outperformed other metaheuristics for TSP cases. In addition, according to many survey papers in the literature, such as those from Mohan and Baskaran (2012) and Neto and Godinho Filho (2013), the ACO algorithm has shown good results in solving a TSP and related variants. Because the IOSRCRVD has characteristics similar to a TSP, the ACO algorithm was applied to solve the problem effectively.

The ACO algorithm is based on the collective actions of ants finding a path to food. Usually, the first ant that finds food returns to the group and leaves a pheromone on the path. Another ant finds the path to the sustenance by following the pheromone of precedent. The second ant also leaves pheromone like the first ant, but the path changes slightly because of an intermediate pheromone. As these processes happen iteratively by a large group of ants, various pathways are accumulated. However, most of the pheromone evaporates over time and eventually only the efficient pathway remains. These processes are encoded into the ACO algorithm to find a near-optimal path or tour in the optimization problem. The algorithm has been widely used to find solutions efficiently in various fields including TSP and vehicle routing problem (VRP). Donati et al. (2008) developed the ACO algorithm to solve the time-dependent VRP. They also conducted the case study in the background of Padua logistic district, the Veneto region of Italy. The proposed ACO algorithm supported by local search procedures was performed to obtain solutions within reasonable computation times. Gajpal and Abad (2009) also studied the VRP with pickup and delivery by the ACO algorithm. They showed remarkable performances through computational experiments by adding a local search scheme to the algorithm. Bontoux and Feillet (2008) applied the ACO algorithm to the generalized TSP, namely traveling purchaser problem. The algorithm was evaluated on 140 instances and showed outstanding performances. Balseiro et al. (2011) combined the ACO algorithm with an insertion heuristic to avoid infeasible solutions at the final stage of the algorithm. The proposed algorithm was competitive to the benchmark problems. In summary, the ACO algorithm has been applied to various problems of finding a tour in a network problem. The algorithm was appropriately modified to fit with their problems. In this study, we modified the ACO algorithm to solve the IOSRCRVD.

The proposed ACO algorithm consists of two steps. In the beginning, the procedure of the algorithm is implemented with the *construction* of a feasible solution. An ant tends to select the next node with high probability, as determined by the roulette wheel method, and repeats the process until completing a tour. In the next step, *pheromone updating* is conducted by modifying the prob-

ability generated by the roulette wheel, which affects the selection of the next node for the repair ants. These steps of the algorithm are repeated to construct and improve feasible solutions.

3.1. Construction of a solution

In the ACO algorithm, two types of ants, repair crew and relief vehicle, were operated in the following network: $G=(V,\ E)$. We denote the terms *repair ant* and *vehicle ant* to represent the ants corresponding to the repair crew and the relief vehicle, respectively. Each ant aims to discover a feasible route from the perspective of repair and delivery, respectively. Because the route of a repair ant affects that of a vehicle ant in a time window, the vehicle ant finds the route to the demand nodes after the repair ant identifies the route to the destroyed nodes. These procedures are repeated to obtain the best routes after the choices were made by both types of ants rather than after each ant had made a choice or series of choices independently. Although the construction of a feasible route in each iteration focuses on the route of each ant, the optimal route necessarily takes into account the perspectives of both ants so that the transportation time for getting relief goods to the demand nodes is minimized, as determined by the time needed for the vehicle ant route to reach the last demand node.

Before searching the routes by the repair ant in the algorithm, the given network, $G=(V,\ E)$, is reduced to a sub-network to establish an efficient computation. The repair ant effectively decides the repairing sequence of the destroyed nodes without considering all the nodes in the network. In other words, the repair ant only takes into account the set $\{0\}\cup V_r$. However, the use of this sub-network could result in an infeasible path being searched. To avoid these futile search processes, a feasibility check process should be added by using Dijkstra's algorithm to determine whether a path between nodes traveled by the repair ant exists in the sub-network. For the computational experiments, if there are no edge-connecting nodes, the distance between nodes is set to M. This approach can effectively reduce computation time compared to that needed for searching the entire network $G=(V,\ E)$. Therefore, we consider a smaller network, $G^r=(\{0\}\cup V_r,E_r)$, for the repair ant.

The amounts of pheromone given off by the repair and vehicle ants between nodes i and j are denoted by τ_{ij}^r and τ_{ij}^v, respectively. The heuristic reciprocal of the time value between nodes i and j is defined as h_{ij}. The parameter α controls the effect of the pheromone, and β controls the heuristic reciprocal. According to these parameters, the probability that the repair and vehicle ants select a next node j from a node i, denoted as p_{ij}^{rk} and p_{ij}^{vk}, respectively, are calculated as follows:

$$p_{ij}^{rk} = \left(\tau_{ij}^r\right)^\alpha \left(h_{ij}\right)^\beta / \sum_{l\in success} \left(\tau_{il}^r\right)^\alpha (h_{il})^\beta$$

$$p_{ij}^{vk} = \left(\tau_{ij}^v\right)^\alpha \left(h_{ij}\right)^\beta / \sum_{l\in success} \left(\tau_{il}^v\right)^\alpha (h_{il})^\beta$$

The procedure for the selection of the next node is executed according to the roulette wheel method. A random number is generated and compared with the cumulative value of p_{ij}^k. If p_{ij}^k is larger than the random number, then the ant moves to the next node j through edge $(i,\ j)$. By iterating the algorithm for a predetermined number of ants, feasible solutions for repair and vehicle ants are constructed and the best solution is obtained by this iteration.

3.2. Pheromone updating

After constructing the feasible solution for the repair and vehicle ants, the pheromone is updated with evaporation rate, ρ and best solution, L_k. Pheromone affects the next route selection for the ants by increasing the probability that a better feasible solution will be found. The ACO algorithm, influenced by the behavior of the ants such that they spray more pheromones on the better routes, is encoded such that a higher probability is given to the better route.

Table 2
Parameters of the ACO algorithm.

Parameter	α	β	ρ	Q	pheromone
Value	0.003	0.02	5000	1000	0.2

Table 3
Information from the data set.

Network size	Small			Middle			Large		
Number of nodes	10	11	12	13	15	17	19	21	23
Number of V_r	4	5	5	5	6	7	8	9	9
Number of V_d	4	4	5	6	7	8	9	10	12

The residual pheromones τ_{ij}^{rk} and τ_{ij}^{vk} between nodes i and j are determined for the repair and vehicle ants, respectively. The pheromone levels are updated according to the best route obtained from the construction phase. To alleviate a biased result of probability, a process for reducing the amount of pheromone by the parameter ρ is incorporated. Both τ_{ij}^{rk} and τ_{ij}^{vk} are updated through the same parameter, ρ, as follows:

$$\tau_{ij}^{rk} \leftarrow (1-\rho)\cdot\tau_{ij}^{rk}$$

$$\tau_{ij}^{vk} \leftarrow (1-\rho)\cdot\tau_{ij}^{vk}$$

The pheromone level is also affected, as follows, by the best feasible solutions, which are determined during the solution construction phase as follows:

$$\tau_{ij}^{rk} \leftarrow \tau_{ij}^{rk} \sum_k Q/L_k \text{ if repair ant } k \text{ moves from node } i \text{ to } j$$

$$\tau_{ij}^{vk} \leftarrow \tau_{ij}^{vk} \sum_k Q/L_k \text{ if vehicle ant } k \text{ moves from node } i \text{ to } j$$

Where Q is a given parameter and L_k is the last transportation time of relief goods in the entire demand nodes. The algorithm is run by repeating the procedures of solution construction and pheromone update until it meets a predetermined condition for termination.

3.3. Pseudo-code of the algorithm

The pseudocode of the ACO algorithm developed for this study was slightly modified from the traditional procedures used to create an algorithm and is described in Algorithm 1.

As shown in the algorithm, network $G=(V,\ E)$, the set and parameters for constructing the IOSRCRVD were initialized the same way as they were in the mathematical model. In the GetFeasibleNodeSet phase, the repair ant searches for destroyed nodes using the Dijkstra's algorithm to determine whether the destroyed node can be reached from the depot through a feasible path. Among the feasible paths, the ant determines the next node with a roulette wheel method. After it finds the next destroyed node, the algorithm updates the CurrentDestroyedNode. Moreover, the TotalDestroyedRoute and Total_RepairTime are calculated by adding the relevant parameters (such as $d_j = d_i^{arr} + t_{ij} + s_j$). Consequently, the Vr_Index is increased by one when the next node is found. By conducting iterations until all of the destroyed nodes are identified by the repair ant, a tour for repairing the destroyed nodes is updated and the repair completion time of each node, d_i, is calculated.

Table 4
Results of computational experiments solved by the two-stage approach, IOSRCRVD, and ACO.

Data Size (Number of ants and iterations)	Number of Nodes	Objective value			Optimality GAP		Computation time		
		Two-stage	IOSRCRVD	ACO	Two-stage	ACO	Two-stage	IOSRCRVD	ACO
Small	10	51	49	49	4.04%	0.00%	3	1.1	2.2
(20, 50)	11	63	51	51	15.69%	0.00%	3	5.6	2.4
	12	65	59	59	10.17%	0.00%	10	11.4	2.7
Middle	13	90	77	77	16.88%	0.00%	11.1	189.5	4.8
(50, 150)	15	96	89 (UB)	86	–	–	2561.4	3600+	11.7
	17	–	129 (UB)	125	–	–	3600+	3600+	14.6
Large	19	–	141 (UB)	131	–	–	3600+	3600+	49.7
(80, 300)	21	–	200 (UB)	150	–	–	3600+	3600+	54.7
	23	–	256 (UB)	177	–	–	3600+	3600+	79.4

Table 5
Computational experiments conducted by varying the number of V_r and V_d nodes.

Number of nodes	Number of V_r nodes	Number of V_d nodes	Objective value			Computation time		
			Two-stage	IOSRCRVD	ACO	Two-stage	IOSRCRVD	ACO
13	4	7	87	84	85	50.6	49.8	9.9
	5	6	90	77	77	11.1	189.5	4.8
	6	5	88	77	77	12.5	14.3	9.0
	7	4	84	76	76	18.3	141.6	8.9
15	5	8	93 (UB)	93(UB)	90	3600+	3600+	12.3
	6	7	96	89 (UB)	86	2561.4	3600+	11.7
	7	6	87	84 (UB)	79	339.7	3600+	11.8
	8	5	82 (UB)	70 (UB)	70	3600+	3600+	10.8
17	6	9	124 (UB)	116 (UB)	122	3600+	3600+	15.9
	7	8	–	129 (UB)	125	3600+	3600+	14.6
	8	7	101 (UB)	103 (UB)	97	3600+	3600+	13.4
	9	6	–	84 (UB)	81	3600+	3600+	13.3

Table 6
Computational experiments conducted by varying the number of edges.

Number of nodes	Number of edges	Objective value			Computation time		
		Two-stage	IOSRCRVD	ACO	Two-stage	IOSRCRVD	ACO
13	14	90	77	77	11	189.5	4.8
	17	61	59	61	49.3	135.1	4.1
	20	52	50	50	48.0	727.9	10.2
15	18	96	89 (UB)	86	2561	3600+	11.7
	21	–	89 (UB)	80	3600+	3600+	11.4
	24	–	68 (UB)	64	3600+	3600+	12.1
17	21	–	129 (UB)	125	3600+	3600+	14.6
	24	–	119 (UB)	111	3600+	3600+	14.7
	27	107 (UB)	104 (UB)	101	3600+	3600+	14.3

After repairing all the nodes, the time windows for a vehicle ant are updated in the UpdateTimewindows phase. To ensure that the vehicle ant does not ignore the destroyed nodes until the nodes are repaired, the cumulative travel time of each node, h_i^{arr}, is updated with d_i, if $h_i^{arr} \leq d_i$. Thus, d_i in V_r can be updated to the time windows for the vehicle ant. In the same manner, after the demand node set used for the vehicle ant is searched with Dijkstra's algorithm, the CurrentDemandNode, TotalDemandRoute, Total_TransportationTime, and Vd_Index are also updated in turn. As a result, each tour for the repair and vehicle ants is completed.

In the final phase of the solution construction, MinObj and MinSol are updated by comparing MinObj and Total_TransporationTime. After the iteration is executed until the last element of Ant_Index is concluded, the UpdatePheromone phase can proceed. The update is processed by the predetermined evaporation ratio, α, to prevent the solution from converging to the local optimum, and $\sum_k Q/L_k$ is used to increase the probability of choosing the route with the better objective function for the next iteration. The procedure for determining the pheromone is similar to that described by Dorigo and Gambardella (1997). Through this process, a near-optimal tour is obtained when the algorithm is iterated until the final iteration number, set as a parameter, is reached.

4. Computational experiments

In this research, numerous computational experiments were conducted by the two-stage approach, IOSRCRVD, and ACO algorithm. At first, experiments were carried out by varying the number of nodes for each of the three models. The two-stage approach and the IOSRCRVD were coded by FICO XPRESS-IVE version 7.9 and the ACO algorithm was coded by Java 1.8.071 language with an Intel® Core™ i5-3470 CPU @ 3.20 GHz. The parameter setting is described in Table 2. Tables 3 and 4 show the information from the data set and the computational results obtained by the three models, respectively.

As shown in Table 4, all three models solved the problem within a short time on small-sized data. Both the IOSRCRVD and the ACO algorithm reached optimal solutions. As mentioned in the previous section, however, the two-stage approach does not take

Algorithm 1 ACO algorithm for the IOSRCRVD.

```
Initialization
G=(V, E); Sets; Parameters;
α; ρ; Q; pheromone; MinSol; MinObj;
TotalRoute; Total_RepairTime; Total_TransportationTime; Timewindows; GetFeasibleNodeSet;
  iter=0;
  While{iter < Num_Iter}
  {
      Ant_Index=0;
      While{Ant_Index < Num_Ant}
      {
          Vr_Index=0;
            Vd_Index=0;
          CurrentNode=0;
          While{Vr_Index < Num_Vr}
          {
                GetFeasibleNodeSet(CurrentDestroyedNode);
                CurrentDestroyedNode=GetNextDestroyedNode();
                TotalDestroyedRoute += GetDestroyedRoute();
                Total_RepairTime += UpdateRepairTime();
                Vr_Index +=1;
        }
          While{Vd_Index < Num_Vd}
        {
                UpdateTimewindows;
                  GetFeasibleNodeSet(CurrentDemandNode);
                CurrentDemandNode=GetNextDemandNode();
                TotalDemandRoute += GetDemandRoute();
                Total_TransportationTime += UpdateTransportationTime ();
                VdIndex +=1;
        }
        If {MinObj > Total_TransportationTime}

            MinObj=Total_TransportaionTime
            MinSol=TotalDestryoedRoute+TotalDemandRoute
          Ant_Index+=1;
      }
      pheromone=UpdatePheromone;
      iter += 1;
  }
```

into account the recovery of a repair node and transport of relief goods at the same time, so it did not provide better solutions than the other two models. In the case of the middle-scale data, neither the two-stage approach model nor the IOSRCRVD solved the entire problem in 3600 s, but the ACO provided the solutions within 3600 s. In addition, the ACO provided better solutions than the upper bounds (UB) of the IOSRCRVD, which were the best solutions obtained with this model up to 3600 s. For large-scale data, both the two-stage approach and the IOSRCRVD failed to solve the problems within 3600 s. However, the IOSRCRVD could provide the upper bound in 3600 s because it solved both aspects simultaneously. In the case of the two-stage approach, the problem of the repair crew is solved first and the output data are used as the input data for the relief vehicle problem. In other words, if the first step does not provide a solution within 3600 s, the second step cannot be started. Therefore, this model could not provide even upper bounds for the large-scale data. In the case of the ACO algorithm, it provides the solutions better than upper bounds of the IOSRCRVD within a reasonable time. The solutions the ACO found in around 60 s were much better than the solutions IOSRCRVD found in 3600 s. As the size of data increased, the gap between the solutions of the ACO and the IOSRCRVD grew. According to this tendency, the discrepancy in performance is expected to appear for problems with large-scale data.

As discussed previously, the performance of the ACO algorithm was inferred as better than the other two models. When the IOSRCRVD could not find the optimal solution within a reasonable time for large data, the ACO algorithm can serve as an alternative to the IOSRCRVD. In the previous computational experiments, the number of V_r and V_d nodes in one network did not differ significantly. Additional experiments were carried out to figure out the effect of the changes in network configurations that depending on the number of V_r and V_d nodes. A description of these experiments and the results are summarized in Table 5.

As presented in Table 5, for the case in which the number of nodes is 13, the IOSRCRVD provided optimal solutions for all data. The computation time did not show any tendency when the number of V_r and V_d nodes was varied. Although some deviation in computation times was found, the process ended within a relatively stable time. In the case of the ACO algorithm, although the optimal solution for the first row data was not obtained, the overall performances were quite good. It also took less time than the IOSRCRVD. The two-stage approach solved all the data within a short time but provided a worse solution than the other two models. For the experiments with 15 nodes, the results were not identical to the case of 13 nodes. The IOSRCRVD could not solve the problems. In the case of two-stage approach, because it solves the problem by decomposing two problems, some problems can be solved more efficiently than the IOSRCRVD did. However, in all cases, it did not provide better solutions than the IOSRCRVD did. In the data set of 17 nodes, an unexpected result was derived in the third-row experiment. The solution of the two-stage approach was better than that of the IOSRCRVD within 3600 s. On the basis of the other experiments, it was expected that the IOSRCRVD would find a better solution when it had more time. Within the limit of 3600 s, however, it could not find a better solution. The ACO algorithm showed that it can provide a better solution than the other two models in short computation time. More experiments were conducted to investigate whether any pattern depends on the spar-

sity of the network. The information about the experiments and the results are presented in Table 6.

Table 6 shows that as the sparsity of the network decreased, the solutions that all three models provided, the complete time of transporting the relief goods decreased. As the number of movable paths increased, the options for choosing a shorter distance for a repair crew increased. For relief vehicles, the options of selecting a short travel time also increased and making a detour through the destroy nodes was possible. The computation time increased for all three models as the number of edges increased. As the number of edges in networks increased, the solution space also increased. It seemed that the computation time increased as the number of solutions to the search increased. For experiments involving 15 and 17 nodes, similar to previous experiments, the two-stage and IOSR-CRVD could not provide the optimal solutions within 3600 s; however, the ACO provided good solutions within a short time.

5. Conclusions

This study makes two major contributions: it was extended to address previously overlooked aspects of simultaneously planned repair and relief and provided an efficient solution procedure. In terms of an extension, the solution approach that Duque et al. (2016) developed only took into account the accessible time of the demand area. In this research, the transportation of relief goods was included. Most previous research dealing with the repair crew problem focused on the scheduling and routing of the repair crew. However, because of the significance of timely supplying of relief goods to demand areas, schedules and routes of the relief vehicle need to be considered as well. Our computational experiments provided compelling evidence that creating the schedule for the repair crew and relief vehicle from the standpoint of one over the other creates delays in transporting the relief goods, but that these delays are not found when a schedule is created that simultaneously integrates repair and relief efforts.

Therefore, the integrated scheduling that considers both crisis recovery aspects is required, and this study is helpful for advancing such decision making after disasters. In the case of the solution procedure, an ACO algorithm is developed to solve the problem rapidly. From the numerical experiments with the MILP, it was found that as the size of the data increased, the computation time was increased significantly. In the case of disasters, decision making based on data within a short time is critical. Therefore, to overcome the limitation of timeliness, the ACO algorithm is developed. Although the MILP could not handle large data sets, the ACO algorithm could solve the large problems efficiently.

This study pioneers concepts in related fields by developing a mathematical model considering a repair crew and a relief vehicle simultaneously. It can be used to support managers in making scientific decisions based on data, rather than their intuition, when an emergency, such as a natural disaster, occurs and a decision on sequencing a road repair is needed. However, some limitations are worth noting. Most importantly, this research is in an early stage, but it is unique in considering both repair and relief aspects simultaneously; therefore, it has considerable potential for extension. Among these extensions, an exploration into uncertainties, driven by various factors, such as an amount of demand and supply goods, would be worthwhile in a study of disasters. Specifically, researchers might consider the uncertainties and develop a mathematical model that best represents reality. Also, only one repair crew and one relief vehicle were addressed in this study. A study on the model with multiple repair crews and relief vehicles would appropriately extend the study. Furthermore, the computational experiments were conducted only with random data sets. Data from additional experiments or real case studies might be useful in additional quantitative analyses.

Acknowledgment

The authors are grateful for the valuable comments from the associate editor and anonymous reviewers. This research was supported by the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning [Grant no. 2017R1A2B2007812].

References

Aksu, D.T., Ozdamar, L., 2014. A mathematical model for post-disaster road restoration: enabling accessibility and evacuation. Transport. Res. Part E 61, 56–67.
Altay, N., Green, W.G., III, 2006. OR/MS research in disaster operations management. Eur. J. Oper. Res. 175 (1), 475–493.
Avci, M., Avci, M.G., 2017. A GRASP with iterated local search for the traveling repairman problem with profits. Comput. Ind. Eng. 113, 323–332.
Balcik, B., Beamon, B.M., 2008. Facility location in humanitarian relief. Int. J. Logist. 11 (2), 101–121.
Balseiro, S.R., Loiseau, I., Ramonet, J., 2011. An ant colony algorithm hybridized with insertion heuristics for the time dependent vehicle routing problem with time windows. Comput. Oper. Res. 38 (6), 954–966.
Bontoux, B., Feillet, D., 2008. Ant colony optimization for the traveling purchaser problem. Comput. Oper. Res. 35 (2), 628–637.
Caunhye, A.M., Nie, X., Pokharel, S., 2012. Optimization models in emergency logistics: a literature review. Socio-Econ. Plann. Sci. 46 (1), 4–13.
Chou, J.S., Tsai, C.F., Chen, Z.Y., Sun, M.H., 2014. Biological-based genetic algorithms for optimized disaster response resource allocation. Comput. Ind. Eng. 74, 52–67.
Donati, A.V., Montemanni, R., Casagrande, N., Rizzoli, A.E., Gambardella, L.M., 2008. Time dependent vehicle routing problem with a multi ant colony system. Eur. J. Oper. Res. 185 (3), 1174–1191.
Dorigo, M., Di Caro, G., 1999. Ant colony optimization: a new meta-heuristic. In: Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on, Vol. 2. IEEE, pp. 1470–1477.
Dorigo, M., Gambardella, L.M., 1997a. Ant colonies for the travelling salesman problem. Biosystems 43 (2), 73–81.
Dorigo, M., Gambardella, L.M., 1997b. Ant colony system: a cooperative learning approach to the traveling salesman problem. IEEE Trans. Evol. Comput. 1 (1), 53–66.
Duque, P.M., Coene, S., Goos, P., Sörensen, K., Spieksma, F., 2013. The accessibility arc upgrading problem. Eur. J. Oper. Res. 224 (3), 458–465.
Duque, P.M., Dolinskaya, I.S., Sörensen, K., 2016. Network repair crew scheduling and routing for emergency relief distribution problem. Eur. J. Oper. Res. 248 (1), 272–285.
Duque, P.M., Dolinskaya, I.S., Sörensen, K., 2014. Network Repair Crew Scheduling and Routing for emergency Relief Distribution Problem. In: Northwestern University, Department of Industrial Engineering and Management Sciences working paper 14-03.
Duque, P.M., Sörensen, K., 2011. A GRASP metaheuristic to improve accessibility after a disaster. OR Spectrum 33 (3), 525–542.
Gajpal, Y., Abad, P., 2009. An ant colony system (ACS) for vehicle routing problem with simultaneous delivery and pickup. Comput. Oper. Res. 36 (12), 3215–3223.
Galindo, G., Batta, R., 2013. Review of recent developments in OR/MS research in disaster operations management. Eur. J. Oper. Res. 230 (2), 201–211.
Hamacher, H.W., Tjandra, S.A., 2001. Mathematical Modelling of Evacuation Problems: A state of Art. Fraunhofer-Institut fur Techno-und Wirtschaftsmathematik, Fraunhofer (ITWM), Kaiserslautern.
Holguín-Veras, J., Jaller, M., Van Wassenhove, L.N., Pérez, N., Wachtendorf, T., 2012. On the unique features of post-disaster humanitarian logistics. J. Oper. Manag. 30 (7–8), 494–506.
Karlaftis, M.G., Kepaptsoglou, K.L., Lambropoulos, S., 2007. Fund allocation for transportation network recovery following natural disasters. J. Urban Plann. Dev. 133 (1), 82–89.
Kim, S., Park, Y., Lee, K., Moon, I., 2017. Repair crew scheduling considering variable disaster aspects. In: IFIP International Conference on Advances in Production Management Systems. Springer, Cham, pp. 57–63.
Kim, S., Shin, Y., Lee, G.M., Moon, I., 2018. Network repair crew scheduling for short--term disasters. Appl. Math. Model. 64, 510–523.
Mohan, B.C., Baskaran, R., 2012. A survey: Ant Colony Optimization based recent research and implementation on several engineering domain. Expert Syst. Appl. 39 (4), 4618–4627.
Neto, R.T., Godinho Filho, M., 2013. Literature review regarding Ant Colony Optimization applied to scheduling problems: Guidelines for implementation and directions for future research. Eng. Appl. Artif. Intel. 26 (1), 150–161.
Osman, M.S., Ram, B., 2017. Routing and scheduling on evacuation path networks using centralized hybrid approach. Comput. Oper. Res. 88, 332–339.
Rancourt, M.È., Cordeau, J.F., Laporte, G., Watkins, B., 2015. Tactical network planning for food aid distribution in Kenya. Comput. Oper. Res. 56, 68–83.
Rath, S., Gutjahr, W.J., 2014. A math-heuristic for the warehouse location–routing problem in disaster relief. Comput. Oper. Res. 42, 25–39.
Sheu, J.B., 2007. An emergency logistics distribution approach for quick response to urgent relief demand in disasters. Transport. Res. Part E 43 (6), 687–709.
Talarico, L., Meisel, F., Sörensen, K., 2015. Ambulance routing for disaster response with patient groups. Comput. Oper. Res. 56, 120–133.

Yan, S., Shih, Y.L., 2009. Optimal scheduling of emergency roadway repair and subsequent relief distribution. Comput. Oper. Res. 36 (6), 2049–2065.
Yan, S., Shih, Y.L., 2012. An ant colony system-based hybrid algorithm for an emergency roadway repair time-space network flow problem. Transportmetrica 8 (5), 361–386.
Yi, W., Kumar, A., 2007. Ant colony optimization for disaster relief operations. Transport. Res. Part E 43 (6), 660–672.
Zhou, X., Shi, Y., Deng, X., Deng, Y., 2017. D-DEMATEL: a new method to identify critical success factors in emergency management. Saf. Sci. 91, 93–104.