Author's Accepted Manuscript

www.elsevier.com/locate/caor

Location arc routing problem with inventory constraints

Juan-Pablo Riquelme-Rodríguez, Michel Gamache, André Langevin

PII: S0305-0548(16)30144-7
DOI: http://dx.doi.org/10.1016/j.cor.2016.06.012
Reference: CAOR4024

To appear in: *Computers and Operation Research*

Received date: 9 June 2014
Revised date: 6 April 2016
Accepted date: 17 June 2016

Cite this article as: Juan-Pablo Riquelme-Rodríguez, Michel Gamache and Andr Langevin, Location arc routing problem with inventory constraints, *Computer and Operation Research,* http://dx.doi.org/10.1016/j.cor.2016.06.012

Location arc routing problem with inventory constraints

Juan-Pablo Riquelme-Rodríguez[1,4], Michel Gamache[1,2], and André Langevin[1,3]

[1]Department of Mathematics and Industrial Engineering, École Polytechnique de Montréal, Canada.
[2]Research Group in Decision Analysis (GERAD), Montréal, Canada.
[3]Interuniversity Research Center on Enterprise Networks, Logistics, and Transportation (CIRRELT), Montréal, Canada.
[4]Department of Industrial Engineering, Universidad Católica del Norte, Antofagasta, Chile.

Authors
Juan-Pablo Riquelme-Rodríguez (Corresponding author)
Department of Industrial Engineering, Universidad Católica del Norte, Antofagasta, Chile.
Department of Mathematics and Industrial Engineering, École Polytechnique de Montréal, Canada.
juan-pablo.riquelme@polymtl.ca
jriquelme@ucn.cl
École Polytechnique de Montréal Office: C-314.10 C.P. 6079, Succ. Centre-ville Montréal, Québec
Canada H3C 3A7
Current address and affiliation:
Universidad Católica del Norte
Office: Y1-323
Casilla 62, sucursal UCN
Código Postal 1240000
Antofagasta
Chile
Phone: (56) 55 235 5167

Michel Gamache
Department of Mathematics and Industrial Engineering, École Polytechnique de Montréal, Canada.
Research Group in Decision Analysis (GERAD), Montréal, Canada.
michel.gamache@polymtl.ca
École Polytechnique de Montréal
Office: A-305.29
C.P. 6079, Succ. Centre-ville
Montréal, Québec
Canada H3C 3A7
Phone: (514) 340-4711 ext. 5920
Fax: (514) 514-340-4086

André Langevin
Department of Mathematics and Industrial Engineering, École Polytechnique de Montréal, Canada.
Interuniversity Research Center on Enterprise Networks, Logistics and Transportation (CIRRELT), Montréal, Canada.
andre.langevin@polymtl.ca
École Polytechnique de Montréal
Office: A-520.31
C.P. 6079, Succ. Centre-ville
Montréal, Québec
Canada H3C 3A7
Phone: (514) 340-4711 ext. 4511
Fax: (514) 340-4463

Abstract

Dust suppression of hauling roads in open pit mines is done by periodically spraying water from a water truck. The objective of this article is to present and compare two methods for locating water depots along the road network so that penalty costs for the lack of humidity in roads and routing costs are minimized. Because the demands are located on the arcs of the network and the arcs require service more than once in a time horizon, this problem belongs to the periodic capacitated arc routing domain. We compare two methods for finding the initial depot location. We then use an exchange algorithm to modify the initial location and an adaptive large neighborhood search algorithm to modify the initial routing of vehicles. This method is the first one used for depot location in periodic arc routing problems.

Keywords: Location arc routing problem, Adaptive large neighborhood search, Periodic capacitated arc routing problem.

1 Introduction

1.1 Location arc routing problems

Arc routing problems deal with operational decisions on how to find the routes that satisfy a set of customers located on the arcs of a network while minimizing the routing cost associated with it. When the objective is to determine the best location of a set of depots so that routing costs are minimized, the problem becomes a location arc routing problem (LARP). Both decisions, locating depots and finding routes to serve customers, are done simultaneously [13].

A classification of location routing problems can be found in [15]. Although many examples refer to location problems in the context of node routing problems, i.e. the demands are located on the nodes of a network, some applications can be found in the arc routing domain. A review of the methods used in location arc routing problems can be found in [16].

The fist LARP application was presented by Levy and Bodin [13] in 1989 to find the best location for postal carriers to park their vehicles so that they can perform a tour to deliver mail. The authors introduced a location, allocation and routing (LAR) approach in which locations for the depots are selected, then the edges to be served are assigned to each depot and finally a route is built to serve those edges. Ghiani and Laporte [7] approached the Location Rural Postman Problem (LRPP) by transforming it to a Rural Postman Problem (RPP) when there are no bounds on the number of depots and by using a branch and cut method to solve it. Ghiani and Laporte [8] reviewed the common applications of LARP, such as mail delivery, garbage collection and street maintenance. The review covered common heuristics used to solve LARP problems that include the above mentioned LAR as well as the ARL, in which customers are first assigned to a vehicle route, then the route is formed and finally, depot locations are determined.

Other applications where location decisions are made in the arc routing domain include garbage collection using mobile depots [6]. Small capacity trucks move along the streets, collecting garbage and delivering their contents into the larger trucks used as temporary depots. The authors use a variable neighborhood descent to schedule meetings of both types of vehicles so that the small trucks reduce the number of returns to the main depot. A similar application was presented in [1], where one type of vehicles is used to paint street lines while a second type is used to refill at specific points in the network.

For this work, we consider an arc routing problem in which a fleet of identical vehicles with limited capacity provide service to the edges of the network. The edges need to be visited more than once in a time horizon. Because several visits are scheduled, the vehicles need to go back to the depot in order to refill and start a new route. This problem is called periodic capacitated arc routing problem (PCARP). The objective is to locate a number of depots along the network so that the refill process can be improved.

1.2 Periodic capacitated arc routing problem

The periodic capacitated arc routing problem, or PCARP, was introduced in [12] for a garbage collection problem in which the demands on the arcs were different from one period to the next one, and a solution was needed for the whole time horizon instead of for individual periods. This problem was shown to be NP-hard because it contains the capacitated arc routing problem (CARP) as a special case [12]. CARP was shown to be NP-hard in [9]. Some applications of the PCARP that have been studied in literature include the already mentioned garbage collection, road monitoring [17] and road maintenance and surveillance [19]. Due to the complexity of the problem,

heuristic and metaheuristic algorithms have been used to solve large PCARP instances, including three heuristic algorithms proposed in [2], a memetic algorithm [11], a scatter search algorithm [3] and an ant cololy optimization algorithm [10].
A special case combines the PCARP problem and inventory management (PCARP-IC). The arcs of a network act as customers that require certain quantity of material in stock. The inventory is replenished periodically by means of delivery vehicles with limited capacity. Vehicles return to a depot for refill and start a new delivery route. Applications include the dust suppression in open-pit mine roads [14] and [23] and plants watering in street medians and sidewalks. This problem was introduced as such in [23]. The authors propose a mathematical model which is capable of solving small instances. An adaptive large neighborhood search was proposed to solve larger instances of this problem in [24]. In order to accelerate the refill of vehicles, depots are strategically located along the network. To the best of our knowledge, location decisions have not been studied for a periodic capacitated arc routing problem. The only studies combining location and routing decisions for periodic applications are presented in [21] and [22]. Both refer to node routing problems.
The methodology presented in this paper is based on the mathematical model proposed in [23] and the heuristic algorithm developed in [24] for the problem presented in [23]. Both articles address the routing problem but not the depot-location problem. The contribution of this article is a solution approach to the location problem in the periodic arc routing domain.
The article is divided as follows: In Section 2, the definition of the problem of road watering in open-pit mines is presented as well as the mathematical model. The solution algorithm is presented in Section 3. Tests results are shown in Section 4. Finally, concluding remarks are presented in Section 5.

2 Mathematical model

2.1 Road humidity and dust retention

In open-pit mines, when vehicles travel along the roads, dust clouds are formed. The most cost-effective method for suppressing dust in these temporary roads is to periodically spray water over them [20]. Due to evaporation and traffic volume, humidity is lost and needs to be replenished. The roads in the network can be traversed in any direction. They are classified according to their priority, where the roads with higher traffic volume have a higher priority. Consider a graph $G(N, E)$ where N is the set of nodes and E is the set of edges that represent the roads. A penalty cost is assigned for having a lower level of humidity than the required one to ensure dust particle retention. Figure 1 shows the humidity level, H_{ij}^t, of edge $(i, j) \in E$. $\overline{h_{ij}}$ is the required level of humidity to ensure dust retention for edge $(i, j) \in E$. After a quantity of water, q_{ij}^{kt}, is delivered by vehicle k at time t, humidity is consumed until the next service. Figure 1a shows the shortage of humidity that occurs when H_{ij}^t is less than $\overline{h_{ij}}$. Figure 1b shows the same situation for a discretized time horizon divided in time periods of equal duration. H_{ij}^t is considered to be constant during a time period.

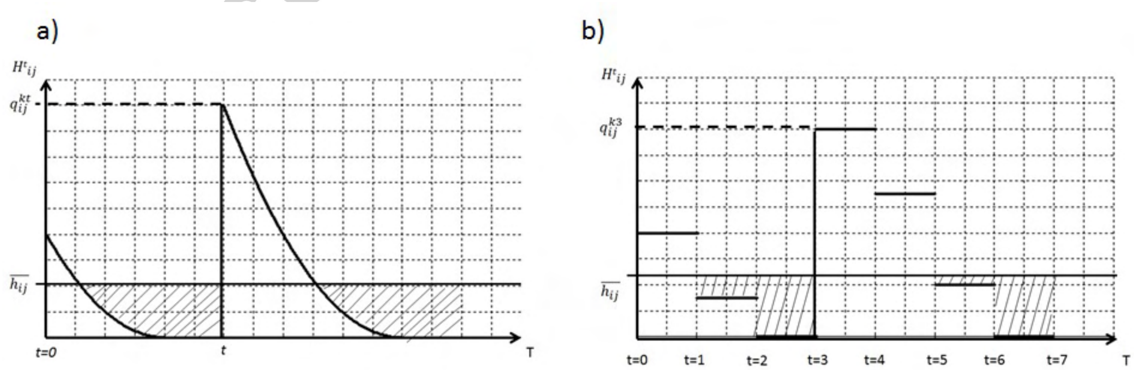

Figure 1: Humidity level of edge (i, j)

Because water trucks have a limited capacity, they reload at a depot before starting a new route. The objective is to find the location of water depots along the mine road network so that the penalty cost for lack of humidity and the routing cost are minimized. For this problem, we consider that the depots can handle any number of vehicles.

This problem combines strategic and operational decisions. The placement of depots is a long term decision, therefore, the performance of the vehicles is tested on different scenarios. A scenario is created when the values of some parameters are changed. For example, at the beginning of the time horizon the roads may have different levels of humidity. Each initial humidity level represents a scenario.

2.2 Mathematical model

The model presented in this section is based on the model presented in [24] that aims to minimize operational costs such as penalty and routing costs when one depot and a fleet of identical vehicles are considered. We include both of these costs tested under different scenarios in order to minimize long term costs such as vehicle and depot placement.

Consider a time horizon that corresponds to one working shift divided in T time periods. A time period is the amount of time it takes a water truck to cover a constant distance D at a constant speed. For example, a truck traveling at $20km/h$ can cover a distance $D = 300m$ in approximately 1 minute (54 seconds). We assume service and deadheading speeds for the water truck are the same. Even when the truck has a faster speed during deadheading, there are several factors affecting this speed such as the presence of other trucks in the same road [14] and the condition of the road [28]. The analysis of different truck speed for the model with one depot and one vehicle is presented in [23]. Consider a mixed network $G(N, E \cup A)$, where N is the set of nodes, and E is the set of edges that correspond to the roads of the mine network. A is the set of arcs that indicate the direction of the traversal of each edge. $B_1 \subseteq A$ is the set of arcs such that for each edge $(i, j) \in E$ there are two artificial arcs $(i, j), (j, i) \in B_1$. $B_2 \subseteq A$ is the set of artificial loops located at each node $i \in N$ where it is possible to place a depot. A loop $(i, i) \in B_2$ is used to simulate the refill of a vehicle. Note that $B_1 \cap B_2 = \emptyset$.

Consider the following parameters of the model:

- C is the number of scenarios that simulate different weather conditions that affect evaporation such as rain or cloudiness.
- K is the number of trucks with the same capacity Q^{max}.
- T is the number of time periods in the time horizon.
- A penalty cost P_{ij} is given to edge $(i, j) \in E$ if the humidity level is below the minimum humidity level $\overline{h_{ij}}$, required to ensure particle retention. The penalty cost is assigned to each edge based on the priority of the road it represents, being the most traversed edges the ones with the highest priority.
- hc_{ij} is the cost of watering arc $(i, j) \in B_1$ or refilling at loop $(i, i) \in B_2$.
- tr_{ij} is the cost traversing arc $(i, j) \in B_1$ without service or waiting at $(i, i) \in B_2$.
- cv is the cost of purchasing a vehicle and cd_b is the cost of establishing a depot in location $b \in B_2$.
- g_{ij} is the percentage of humidity that is lost each period because of the traffic volume in edge $(i, j) \in E$. The edges with more traffic volume have a higher loss of humidity.
- e^{tc} is the evaporation caused by the time period $t \in \{0, \ldots, T\}$ and scenario $c \in \{1, \ldots, C\}$. It increases or decreases the amount of humidity that is lost each period.
- I^c_{ij} represents the initial humidity level of edge $(i, j) \in E$ in scenario c.
- H^{max}_{ij} is the maximum humidity level allowed in edge (i, j).
- $MinDep$ and $MaxDep$ are, respectively, the minimum and maximum number of depots allowed in the network. In an open-pit mine, the number of potential location sites is limited by the fact that the mine's topologies are in perpetual evolution.
- d_{ij} is the number of time periods required to travel along arc $(i, j) \in A$.
- $\Omega \subseteq A$ is the set of arcs that require more than one time period to be traversed, i.e., $d_{ij} > 1$.

Consider the following variables of the model:

- H^{tc}_{ij} is the humidity level of edge $(i, j) \in E$ at time t of scenario c.

- $Y_{ij}^{ktc} = 1$ if vehicle $k \in \{1, \ldots, K\}$ traverses arc $(i,j) \in B_1$ without service or waits at $(i,i) \in B_2$ at the beginning of time t of scenario c, 0 otherwise.
- $X_{ij}^{ktc} = 1$ if vehicle k waters arc $(i,j) \in B_1$ or refills at $(i,i) \in B_2$ at the beginning of time t of scenario c, 0 otherwise.
- $Z_b = 1$ if arc $b \in B_2$ is chosen as a location for a depot, 0 otherwise.
- $R_k = 1$ if vehicle k is used, 0 otherwise.
- Q^{ktc} is the quantity of water in vehicle k at the beginning of time t of scenario c.
- q_{ij}^{ktc} is the quantity of water delivered to edge $(i,j) \in E$ by truck k at the beginning of time t of scenario c.
- q_b^{ktc} is the quantity of water to be refilled to vehicle k at depot $b \in B_2$, at the beginning of time t of scenario c.
- $S_{uv}^{ktc} = 1$ if vehicle k is between nodes u and v of arc $(u,v) \in \Omega$ at the beginning of time t of scenario c, 0 otherwise.
- $w_{ij}^{tc} = \max\{0, \overline{h_{ij}} - H_{ij}^{tc}\}, \forall (i,j) \in E$. It is the difference in humidity levels when $\overline{h_{ij}} \geq H_{ij}^{tc}$ as depticted in Figure 1, 0 otherwise.

The complete model is as follows:

$$\min \frac{1}{C}\left[\sum_{c=1}^{C}\sum_{(i,j)\in E}\sum_{t=0}^{T} P_{ij} w_{ij}^{tc} + \sum_{c=1}^{C}\sum_{(i,j)\in E}\sum_{k=1}^{K}\sum_{t=0}^{T} hc_{ij}(X_{ij}^{ktc} + X_{ji}^{ktc}) + tr_{ij}(Y_{ij}^{ktc} + Y_{ji}^{ktc})\right] + \sum_{k=1}^{K} cvR_k + \sum_{b\in B_2} cd_b Z_b \tag{1}$$

subject to:

$$w_{ij}^{tc} \geq \overline{h_{ij}} - H_{ij}^{tc} \qquad \forall (i,j) \in E,\ t \in \{0,\ldots,T\},\ c \in \{1,\ldots,C\} \tag{2}$$

$$H_{ij}^{(t+1)c} = (1 - (e^{tc} g_{ij}))H_{ij}^{tc} + \sum_{k=1}^{K} q_{ij}^{k(t+1)c} \qquad \forall (i,j) \in E,\ t \in \{0,\ldots,T-1\},\ c \in \{1,\ldots,C\} \tag{3}$$

$$H_{ij}^{0c} = I_{ij}^{c} \qquad \forall (i,j) \in E,\ c \in \{1,\ldots,C\} \tag{4}$$

$$q_{ij}^{ktc} \leq H_{ij}^{max}(X_{ij}^{ktc} + X_{ji}^{ktc}) \qquad \forall \{(i,j) \in E,\ t \in \{0,\ldots,T\},\ k \in \{1,\ldots,K\},\ c \in \{1,\ldots,C\} \tag{5}$$

$$H_{ij}^{tc} \leq H_{ij}^{max} \qquad \forall (i,j) \in E,\ t \in \{0,\ldots,T\},\ c \in \{1,\ldots,C\} \tag{6}$$

$$\sum_{(i,j)\in E} q_{ij}^{ktc} \leq Q^{ktc} \qquad \forall t \in \{0,\ldots,T\},\ k \in \{1,\ldots,K\},\ c \in \{1,\ldots,C\} \tag{7}$$

$$Q^{k0c} = Q^{max} \qquad \forall\, k \in \{1,\ldots,K\},\ c \in \{1,\ldots,C\} \tag{8}$$

$$Q^{ktc} \leq Q^{max} \qquad \forall t \in \{0,\ldots,T\},\ k \in \{1,\ldots,K\},\ c \in \{1,\ldots,C\} \tag{9}$$

$$Q^{k(t+1)c} = Q^{ktc} - \sum_{(i,j)\in E} q_{ij}^{ktc} - \sum_{b\in B} q_b^{ktc} \qquad \forall t \in \{0,\ldots,T-1\},\ k \in \{1,\ldots,K\},\ c \in \{1,\ldots,C\} \tag{10}$$

$$q_b^{ktc} = -Q^{max} X_b^{ktc} \qquad \forall t \in \{0,\ldots,T\},\ k \in \{1,\ldots,K\},\ b \in B_2,\ c \in \{1,\ldots,C\} \tag{11}$$

$$\sum_{c=1}^{C}\sum_{(i,j)\in E}\sum_{T=0}^{T}(X_{ij}^{ktc}+X_{ji}^{ktc})\leq R_kTC \qquad \forall k\in\{1,\ldots,K\} \tag{12}$$

$$\sum_{c=1}^{C}\sum_{t=0}^{T}\sum_{k=1}^{K}(X_{b}^{ktc}+Y_{b}^{ktc})\leq Z_bKTC \qquad \forall b\in B_2 \tag{13}$$

$$MinDep\leq\sum_{b\in B_2}Z_b\leq MaxDep \tag{14}$$

$$Z_{dep}=1 \tag{15}$$

$$\sum_{(i,j)\in E}(X_{ij}^{ktc}+Y_{ij}^{ktc}+X_{ji}^{ktc}+Y_{ji}^{ktc})\leq 1 \qquad \forall t\in\{0,\ldots,T\},\ k\in\{1,\ldots,K\},\ c\in\{1,\ldots,C\} \tag{16}$$

$$\sum_{k=1}^{K}(X_{ij}^{ktc}+X_{ji}^{ktc})\leq 1 \qquad \forall (i,j)\in E,\ t\in\{0,\ldots,T\},\ c\in\{1,\ldots,C\} \tag{17}$$

$$\sum_{(i,j)\in E|i=dep}X_{ij}^{k0c}+Y_{ijc}^{k0c}=1 \qquad \forall k\in\{1,\ldots,K\},\ c\in\{1,\ldots,C\} \tag{18}$$

$$\sum_{(i,j)\in E|j=dep}X_{ij}^{kTc}+Y_{ij}^{kTc}=1 \qquad \forall k\in\{1,\ldots,K\},\ c\in\{1,\ldots,C\} \tag{19}$$

$$X_{ij}^{ktc}+Y_{ij}^{ktc}\leq\sum_{u|(j,u)\in E}(X_{ju}^{k(t+d_{ij})c}+Y_{ju}^{k(t+d_{ij})c})+\sum_{v|(v,j)\in E}(X_{jv}^{k(t+d_{ij})c}+Y_{jv}^{k(t+d_{ij})c})$$
$$\forall (i,j)\in E,\ t\in\{0,\ldots,T\},\ k\in\{1,\ldots,K\},\ c\in\{1,\ldots,C\} \tag{20}$$

$$X_{ji}^{ktc}+Y_{ji}^{ktc}\leq\sum_{u|(i,u)\in E}(X_{iu}^{k(t+d_{ij})c}+Y_{iu}^{k(t+d_{ij})c})+\sum_{v|(v,i)\in E}(X_{iv}^{k(t+d_{ij})c}+Y_{iv}^{k(t+d_{ij})c})$$
$$\forall (i,j)\in E,\ t\in\{0,\ldots,T\},\ k\in\{1,\ldots,K\},\ c\in\{1,\ldots,C\} \tag{21}$$

$$X_{ij}^{ktc}+Y_{ij}^{ktc}+X_{ji}^{ktc}+Y_{ji}^{ktc}\leq S_{ij}^{k(t+m)c} \qquad \forall (i,j)\in\Omega,\ m\in\{0,\ldots,d_{ij}-1\},\ t\in\{0,\ldots,T-m\},\ k\in\{1,\ldots,K\},\ c\in\{1,\ldots,C\} \tag{22}$$

$$\sum_{(i,j)\in E\setminus\{(u,v)\}}(X_{ij}^{ktc}+Y_{ij}^{ktc}+X_{ji}^{ktc}+Y_{ji}^{ktc})\leq 1-S_{uv}^{ktc} \qquad \forall t\in\{0,\ldots,T\},\ (u,v)\in\Omega,\ k\in\{1,\ldots,K\},\ c\in\{1,\ldots,C\} \tag{23}$$

$$H_{ij}^{tc},\ w_{ij}^{tc}\geq 0 \qquad \forall (i,j)\in E, t\in\{0,\ldots,T\},\ c\in\{1,\ldots,C\} \tag{24}$$

$$q_{ij}^{ktc}\geq 0 \qquad \forall (i,j)\in E, t\in\{0,\ldots,T\},\ k\in\{1,\ldots,K\},\ c\in\{1,\ldots,C\} \tag{25}$$

$$q_{b}^{ktc}\leq 0 \qquad \forall t\in\{0,\ldots,T\},\ k\in\{1,\ldots,K\},\ b\in B_2,\ c\in\{1,\ldots,C\} \tag{26}$$

$$X_{ij}^{ktc},\ Y_{ij}^{ktc},\ X_{ji}^{ktc},\ Y_{ji}^{ktc}\in\{0,1\} \qquad \forall (i,j)\in A, t\in\{0,\ldots,T\},\ k\in\{1,\ldots,K\},\ c\in\{1,\ldots,C\} \tag{27}$$

$$Q^{tc}\geq 0 \qquad \forall t\in\{0,\ldots,T\},\ c\in\{1,\ldots,C\} \tag{28}$$

$$S_{uv}^{ktc}\in\{0,1\} \qquad \forall (u,v)\in\Omega, t\in\{0,\ldots,T\},\ c\in\{1,\ldots,C\},\ k\in\{1,\ldots,K\} \tag{29}$$

$$Z_b\in\{0,1\} \qquad \forall b\in B_2 \tag{30}$$

$$R_k\in\{0,1\} \qquad \forall k\in\{1,\ldots,K\} \tag{31}$$

The objective function (1) minimizes the total cost. The penalty cost for a shortage of humidity and the routing cost of watering and deadheading are the result of the average cost of all scenarios. It is added to the vehicle cost and the depot location cost.

Constraints (2) define variable w_{ij}^{tc} as the difference between the required and actual humidity levels at a time t for each scenario. If positive, this difference is penalized in the objective function. Constraints (3) estblish the humidity level for the next time period, for each scenario. Constraints (4) establish the initial humidity level of each edge for each scenario. Constraints (5) and (6) limit, respectively, the amount of water delivered and the humidity level to the maximum humidity level. Constraints (7) are vehicle capacity constraints. Constraints (8) determine the initial quantity of water available in each vehicle. Constraints (9) limit the vehicle capacity at each time period to the maximum capacity. Constraints (10) determine the quantity of water in the vehicle at each time period. Constraints (11) allow the truck refilling at each depot. Constraints (12) allow the use of trucks. Constraints (13) allow the usage of a selected depot. Constraint (14) establishes the limits on the number of depots. Equation (15) determines the main depot, $dep \in N$. This is the depot from which vehicles depart at the beginning of the time horizon using constraints (18) and to which they return at the end of it according to constraints (19). Constraints (16) limit the vehicle to either water or traverse an edge at a specific time period. Constraints (17) state that an edge can be serviced by only one vehicle at the same time period. Constraints (20) are flow conservation constraints for all arcs in the network. Constraints (22) and (23) ensure that the vehicle stays on the same edge during the number of time periods it takes to service or traverse it. Finally, constraints (24) to (31) define the variables of the model.

We added the following constraints to break the symetry in the solution tree:

$$Y_{00}^{ktc} = 1 \qquad \forall t \in \{0, \ldots, k-1\},\ k \in \{1, \ldots, K\},\ c \in \{1, \ldots, C\} \tag{32}$$

$$\sum_{(0,j) \in A} X_{0j}^{ktc} + Y_{0j}^{ktc} = 1 \qquad \forall t = k,\ k \in \{1, \ldots, K\},\ c \in \{1, \ldots, C\} \tag{33}$$

Restrictions (32) ensures that vehicles stay at the initial depot for an increasing number of periods. The first vehicle will stay at the depot until the first time period, the second vehicle will stay until the second period and so on. Equations (33) ensure vehicles leave the depot one at each period.

This model was coded in cplex OPL and solved for a network of 8 nodes and 11 edges, $K = 5$ vehicles and 10 scenarios. The scenarios included 10 different values for I_{ij}^c. The stopping criterion was 2 hours or reaching 2% relative gap. The program was able to solve this instance for less than 18 time periods before meeting the stopping criterion. These parameters resulted in a problem with 160000 restrictions and 97000 variables. For larger instances and a larger time horizon, a heuristic method is required.

3 Location and routing algorithms

Two LARP heuristics were described in [8]: The Location-Allocation-Routing (LAR) heuristic and the Allocation-Routing-Location (ARL). In the first approach, the depots are located, the edges of the network are assigned to the depots and then, a route is formed. In the second one, the edges are allocated and a route is formed for each vehicle, leaving the depot location at the end. Both approaches were developed for non-periodic problems where the resulting depot location provides a solution for one visit to the edges. In our problem, edges are visited more than once, therefore, a solution for a non-periodic problem may not be adequate. We choose a different approach taking elements from LAR and ARL in order to test the routing and location decisions after several visits to the edges.

In our approach, allocation is done independently of the depot location. The allocation step consists in assigning edges to vehicles so that each edge is served by only one vehicle in one or several opportunities during the time horizon. The depot assignment to nodes is done using the two methods described in this section. Routing is done at the end considering the results of the two previous steps.

3.1 Location algorithms

The first step of this approach is the location of depots in the nodes of the network. We developed two methods for the initial location of depots.

1. Levy and Bodin location method.

This method was proposed by Levy and Bodin [13] for a mail delivery application. In this problem, postal carriers park their vehicles at specific locations and walk to deliver mail. The objective is to find the best location to park the vehicles. The authors follow the LAR approach. The location phase is done by classifying the nodes of the network in decreasing order of their attractiveness measure and selecting the nodes using a separation criterion.

We adapt the location phase of the Levy and Bodin's LAR method to our problem as follows:

- Determine the nodes that can be used as depots. For this problem we are assuming every node is a potential location for a depot.
- Arrange the nodes in decreasing order of their *attractiveness measure* (AM). A node has a higher AM if there are more incident edges to it and the priority of the incident edges is higher. Priority is related to the edge's traffic volume, being the ones with the greatest volume, the ones with the highest priority. For our problem, priority is related to the parameter P_{ij} of the mathematical model.

$$\text{AM} = A_1\text{*Number of incident nodes} + A_2\text{*Sum of the priorities of incident nodes}$$

  where A_1 and A_2 are the weights given to each contributing factor of the attractiveness measure.

- Choose the nodes from the list using a separation criterion. Start with the first node of the list, remove it and discard the nodes whose distance to it is less than the distance of the separation criterion. Once the nodes are discarded, select, from the remaining nodes the first one and repeat the process until there are no nodes left in the AM list. The distance between nodes is calculated using Dijkstra's algorithm. The selected node is removed from the AM list and added to the depot list. The minimum distance is calculated from a percentage of the total distance the truck is able to water having full capacity, DC. If DC is closer to 1, the minimum distance required between locations is greater, resulting in less nodes selected as depots. If DC is closer to 0, the required distance is reduced, allowing more nodes to be selected as depots.

2. Minimum distance.
This method distributes depots along the nodes of the network so that the distance from the edges to the selected nodes is minimized. Consider the same graph $G(N, E)$ defined in section 2.1. Consider $X_{ij} = 1$ if edge $j \in E$ is assigned to depot $i \in N$, 0 otherwise; and $Y_i = 1$ if a depot is placed at node i, 0 otherwise. c_{ij} represents the shortest path from depot i to the nearest end of edge j. P is the number of depots obtained using Levy and Bodin's method and $|E|$ is the number of edges.

$$\min \sum_{i \in N} \sum_{j \in E} c_{ij} X_{ij} \tag{34}$$

subject to:

$$\sum_{i \in N} Y_i = P \tag{35}$$

$$\sum_{i \in N} X_{ij} = 1 \qquad \forall j \in E \tag{36}$$

$$\sum_{j \in E} X_{ij} \leq |E| Y_i \qquad \forall i \in N \tag{37}$$

$$X_{ij} \in \{0, 1\} \qquad \forall i \in N, j \in E \tag{38}$$

$$Y_i \in \{0, 1\} \qquad \forall i \in N \tag{39}$$

The objective function (34) minimizes the shortest distance between the selected depots and the edges of the network. Equation (35) establishes the number of depots. Equations (36) ensures all edges are assigned to a depot. Constraints (37) allow an edge to be assigned to node i if it was selected as depot. Constraints (38) and (39) define the variables.
The two methods result in a list of nodes that are used to locate a depot. The first node of the list is considered the *main* depot. The main depot is the place where vehicles start and where they return at the beginning and end of the time horizon.
The next two steps, allocation and routing, are based on the initial solution of the algorithm proposed by Riquelme-Rodríguez et al. [24] for the PCARP with inventory constraints.

3.2 Allocation

The network is partitioned in K sets of edges, where K is the number of vehicles. This procedure was described in [24] and is based on the *cluster first-route second* algorithm used in [19] for the PCARP with irregular services. It can be summarized as follows:

1. Select K edges called *seeds* as far away from each other as possible. The shortest path between edges e_1 and e_2, $SP_{e_1e_2}$, is calculated using Dijkstra's algorithm. The first seed s_1, is the one with the highest SP_{0s_1}, where 0 represents the main depot selected using any of the location methods. From a set of selected seeds $\{s_1, \ldots, s_h\}$, when $h < K$, select edge e as s_{h+1} so that $SP_{0e}(\prod_{i=1}^{h} SP_{s_ie})$ is maximum.

2. Assign the rest of the edges of the network by minimizing the sum of the lenghts of shortest paths from the edges to the seeds selected in step 1. For our problem, edges were assigned to their seeds by solving a standard assignment problem where the sum of the length of the edges assigned to each seed has to be at least a fraction of the length of all the edges in the network.

3.3 Routing

A route is calculated for each one of the vehicles by means of a constructive algorithm, first proposed in [24]. Each vehicle can serve only the edges assigned to it in the allocation phase, but it can traverse any edge. A route with a starting and ending depot is called a *run*. The main depot represents the start of the first run and the end of the last one. Any vehicle can refill at any depot, including the main depot, before starting a new run. For each run, the total quantity of water used is calculated as well as the time needed to traverse the arcs on that run. Our routing procedure can be sumarized for each one of the vehicles as follows:

1. Start with a list L_1 of available edges, i.e., edges whose humidity level is below the required level and need service. After an edge is serviced in the first run, it will not be available for service (removed from L_1) for subsequent runs until H_{ij}^{kt} is reduced enough so that it is available again.

2. From L_1, select the edge, (i, j), with the highest priority and add it to a list of selected edges. Denote L_2 the list of edges from L_1 that are selected to be serviced.

3. Assign a quantity of water to be delivered equal to $\alpha D d_{ij}$, where α is the water rate in number of liters per meter and D is the distance, in meters, a truck travels in one time period at a constant speed.

4. Order the available neighbors, i.e., adjacent edges to the edges in L_2 that need service, in increasing order of the length of their shortest paths to the departing depot and select the first edge to add to L_2. If no neighbor is available, select the closest available edge to the edges in L_2.

5. Repeat step 4 until the residual amount of water in the truck is less than $\alpha D d_{ij}, \forall (i, j) \notin L_1$ or $L_1 = \emptyset$.

6. Order the edges in L_2 in increasing order of their shortest paths to the departing depot.

7. Connect the first edge to the departing depot calculating the shortest path between them. Connect the rest of the edges in L_2 using the shortest paths between them. The direction in which the edges are traversed is important. An edge (i, j) that is traversed either in the direction of the arc (i, j) or (j, i) is removed from L_2. The process is repeated until $L_2 = \emptyset$.

8. Find the closest depot to the last arc (i, j) in the sequence and find the shortest path to connect them. The set of arcs starting and ending in a depot is called a *run*.

9. Create as many runs as possible within the time horizon, repeating steps 1-8. If the time to complete a run ends after the end of the time horizon, reorganize the path to end the run at the main depot at the end of the time horizon.

3.4 Adaptive large neighborhood search

The improvement process was done by modifying the initial solution in two stages. The change in the initial depot location was done using the algorithm described in Section 3.5. The change in the allocation and routing of vehicles was changed using the adaptive large neighborhood search heuristic (ALNS) as described in this section.
The ALNS algorithm was introduced in [25]. This algorithm has been used in the arc routing domain for the synchronized arc routing problem [26] in which snow plowing operations are performed by coordinating a set of vehicles, and for a road marking application [27]. Applications that require the combination of routing and inventory decisions in the node routing domain that use the ALNS can be found in [5] and [4]. For these examples, the customers are located in the nodes of the network instead of the arcs.
The ALNS is an iterative process in which a set of destroy/repair operators, $\{O_1, \ldots, O_n\}$, modify the initial solution.

The process is divided in segments. For our problem, a segment consists of 250 iterations. In each iteration, an operator O_i is chosen using a *roulette-wheel* mechanism, i.e. it is chosen with probability $\rho_i/\sum_{i=1}^{n}\rho_i$, where ρ_i is the weight of O_i. The weight is defined as $\rho_i = SC_i/U_i$ where SC_i is the score of O_i and U_i is the number of times it was used during one segment. When a segment starts, SC_i is set to 0 for all operators and the weights are updated so the probability of selecting each operator changes according to past performance. The operators with better performance have a higher probability of being selected. At the beginning of the procedure, the weights are the same, $\rho_i = 1/n$. After segment j, the weights for segment $j+1$ are calculated using $\rho_{i,j+1} = \rho_{ij}(1-r) + r\frac{SC_i}{U_i}$, where $r \in [0, 1]$ is called *reaction factor*. SC_i is determined by three scores: σ_1 is awarded if the use of O_i results in a better solution overall; σ_2 is awarded if the use of O_i results in a better solution than the incumbent one, but not better than the best solution and it has not been explored before; and σ_3 is awarded if O_i results in a bad solution, accepted with probability $e^{-(f'-f)/\tau}$, being f and f' the objective vlaues of the current and new solution respectively, while the temperature factor τ, starts with τ_0 and decreases with each iteration $\tau = \tau \cdot c$, where $0 < c < 1$. τ_0 is calculated from the initial solution so that a solution that is μ% worse than the current solution is accepted with 50% probability [25].
We use seven destroy/repair operators from a previous application [24]. The first three operators, O_1, O_2, and O_3, have the purpose of diversification. They randomly exchange the edges that were previoulsy assigned to K lists after the allocation phase, called *allocation lists*. The rest of the operators are intensification operators. They perform changes in the sequences created after the routing phase. They are summarized as follows:

- O_1 randomly selects one edge from an allocation list and exchanges it with another randomly selected edge from a different allocation list.
- O_2 exchanges two adjacent, randomly selected edges from two different allocation lists.
- O_3 exchanges a random number of edges from two allocation lists.
- O_4 The service status of a random number of edges is changed from *serviced* to *not serviced* and vice versa.
- O_5 randomly exchanges two sequences of arcs (serviced or not) created after the routing phase. Both sequences are performed by the same vehicle. Because the inserted sequences may not be adjacent to the existing arcs, the shortest path between them is calculated.
- O_6 is similar to O_5 except that the two sequences exchanged are performed by two different vehicles. If the total water used from the resulting inserted run in O_5 or O_6 excedes the capacity of the vehicle, thus becoming infeasible, edges are removed from the route starting from the last one until the solution regains feasibility. The removed edges will not be served in this run.
- O_7 randomly changes the amount of water to be delivered.

All operators destroy the existing solution and repair it in the same iteration. All repaired solutions are feasible. If an operator results in an unfeasible solution, it is discarded.
Because of the inventory constraints in the mathematical model, i.e., constraints (3) and (10), that show the level of humidity and vehicle capacity in period $t+1$ from the existing levels in period t, every time an operator performs any change in the solution, the rest of the solution needs to be re-calculated for the rest of the time horizon. For example, if an edge previously not served is served due to operator O_4, it will not be available for the next run, thus the list of edges that can be serviced in the second run is different.

3.5 Modification of the inital depot location

The ALNS algorithm changes the allocation and routing of the vehicles in the solution. In order to change the location of depots, we developed an algorithm that exchanges the nodes based on the usage of depots after the routing was calculated. The usage is calculated by counting the number of visits to each of the depots by all the vehicles during a time horizon. Six exchange operations J_1 to J_6 are described as follows:

- J_1. The least visited depot is removed and the next node in the attractiveness list, i.e., the most attractive, is selected as the location of the depot.
- J_2. The least visited depot is removed and the last node in the attractiveness list, i.e., the least attractive, is selected as the location of the depot.

- J_3. The depot that was visited the most is removed and the next node in the attractiveness list is selected as the location of the depot.
- J_4. The depot that was visited the most is removed and the location is exchanged for the last node in the attractiveness list.
- J_5. The least visited depot is removed and the location node is exchanged for a random node that has not been selected.
- J_6. The most visited depot is removed and the location node is exchanged for a random node that has not been selected.

To avoid cycles in operations J_1 to J_4, the discarded node is placed at the end of the attractiveness list for J_1 and J_3, and at the beginning of the list for J_2 and J_4, so that all nodes can be selected if the operations are repeated at least $|N|$ times.
All nodes are considered for the exchange except the main depot, located at node 0, that appears in all solutions. After the location is changed, the new route is calculated and the value of the objective function noted. Each procedure J_i is repeated until a stopping criterion is reached. The node combination with the depot location that resulted in the best value of the objective function is selected as the new location. These procedures mantain the same number of depots.

4 Test results

4.1 Parameters

We performed the tests on the *mine* instances from [24]. The road networks are taken from 5 real mines, the number of vehicles used is either 3 or 5, labeled *3V* or *5V*.
Location parameters used to compute the attractiveness measure, A_1 and A_2, are set to 0.5 and 0.5.
The ALNS parameters were tuned by changing the value of one of them and choosing the one that delivered the best objective value. Keeping the first parameter fixed, the second one was tested for several values. This process was reapeated for all the parameters resulting in the values for $(r, \mu, \sigma_1, \sigma_2, \sigma_3) = (0.1, 0.8, 25, 10, 5)$.
Network parameters are take from [24]. The time horizon is $T = 480$ time periods. The traversing cost is the same as the edge distance, i.e., $r_{ij} = d_{ij}$, while the watering cost is set to $c_{ij} = r_{ij} + 2$. In order to give priority to penalty costs $P_{ij} = \max c_{ij} \cdot priority_{ij}$, where the priority of edge (i, j) is an integer from 1 to 10. The required level of humidity is set to $\overline{h_{ij}} = \alpha D d_{ij}$. I^{tc}_{ij} and H^{max}_{ij} are set to $0.1\overline{h_{ij}}$ and $1.1\overline{h_{ij}}$ respectively. b_{ij} is set to 3% for low priority edges and 7% to high priority ones. $e_t = e_{t-1} + 0.1$ if $\lceil 0.55T \rceil \leq t < \lceil 0.75T \rceil$; $e_t = e_{t-1} - 0.1$ if $\lceil 0.75T \rceil \leq t < \lceil 0.95T \rceil$; and $e_t = 1$ otherwise. These parameters are used to approximate the evaporation rate during daytime, which is increased on the second half of the time horizon as shown in the results of hourly evaporation found in [18].
Table 1 shows the characteristics of each network. $|N|$ represents the number of nodes and $|E|$ the number of edges. All algorithms were coded in Python and executed on a 4.0 GHz Intel Core i7-4790K CPU.

Network	Number of vehicles	$\|N\|$	$\|E\|$
mine1 3V	3	21	22
mine1 5V	5	21	22
mine2 3V	3	22	27
mine2 5V	5	22	27
mine3 3V	3	49	53
mine3 5V	5	49	53
mine4 3V	3	51	60
mine4 5V	5	51	60
mine5 3V	3	30	35
mine5 5V	5	30	35

Table 1: General information on the *mine* instances.

4.2 Routing costs with different number of depots

This test is a comparison of the routing costs for locating one or several depots. We compared the two methods for establishing the initial location of depots. The cost corresponds to the average cost obtained after testing 18 different scenarios. We tested six different initial humidity levels, I_{ij}^{c}: 10%, 20%, 30%, 40%, 50% and 60% of H_{ij}^{max}. These six scenarios were combined with three evaporation scenarios: The first scenario is an approximation of the evaporation behaviour described in [18] where the evaporation rate is higher during afternoon hours; the second scenario has no evaporation, and the loss of humidity is due to traffic volume only; the third scenario is a situation where humidity is briefly increased by weather events such as rain.

As explained in Section 3.1, parameter DC is used to determine the required distance between depots. The closer DC is to 1, the longer the distance between depots and the less nodes are selected as depots in the network. Three different values of DC were tested, 0.9, 0.6, and 0.3, resulting in a different number of depots. Column 1 of Table 2 shows the network from *mine1* to *mine5* using 3, and 5 vehicles. Column 2 shows the average cost of the 18 scenarios when the only depot is the main one at node 0. Columns 3, 6 and 9 show the resulting number of depots that correspond to the different values of parameter DC. Columns 4, 7, and 10 show the average cost of the 18 scenarios using the location algorithm of Levy and Bodin. Columns 5, 8 and 11 show the average cost of the 18 scenarios using the location algorithm of minimum distance.

Table 2 results show a decrease in the routing costs as the number of depots increase. This result was expected

Network	One depot at node 0	$DC = 0.9$			$DC = 0.6$			$DC = 0.3$		
		No. of depots	cost LB	cost MD	No. of depots	cost LB	cost MD	No. of depots	cost LB	cost MD
mine1 3V	2014.40	1	2014.40	2014.40	4	2001.84	**1986.16**	6	1972.40	**1954.83**
mine1 5V	1889.25	1	1889.25	1889.25	4	**1640.88**	1714.52	6	**1568.54**	1723.43
mine2 3V	5182.99	3	5218.69	**5190.36**	6	5096.41	**5093.13**	11	**5074.33**	5074.42
mine2 5V	4978.52	3	4942.01	**4823.47**	6	4824.69	**4781.46**	11	**4670.38**	4709.56
mine3 3V	9145.62	5	9082.44	**9042.47**	7	9021.61	**8989.59**	15	9062.56	**8959.55**
mine3 5V	8935.03	5	8817.63	**8761.94**	7	**8764.35**	8768.32	15	8726.72	**8679.99**
mine4 3V	6544.99	4	6481.12	**6422.85**	6	6472.15	**6412.50**	10	6436.93	**6392.52**
mine4 5V	6313.80	4	6253.63	**6139.78**	6	6215.15	**6134.68**	10	6149.23	**6089.48**
mine5 3V	3901.29	3	3817.30	**3799.84**	5	3817.43	**3708.06**	7	3730.41	**3705.61**
mine5 5V	3619.95	3	3555.78	**3503.90**	5	3548.02	**3502.47**	7	**3433.68**	3473.76

Table 2: Cost comparison using different number of depots for both location methods and 18 scenarios.

since the purpose of adding depots to the network is to reduce the routing costs. Comparing both location methods, the results suggest that the minimum distance method performs better than Levy and Bodins method for all values of DC compared. It is also evident that Levy and Bodin's is relatively better for many depots than for a few.

Because the location algorithms are performed independently from the allocation and routing, it is the same location for each network either with three or five vehicles. It can be seen from Table 2 that in every network, the cost decreases as the number of vehicles increases from 3 to 5. This is an expected result as the objective of increasing the long term investment in vehicles is to reduce the routing costs.

To ilustrate an example of the location results, Figure 2 shows the *mine2* network. The nodes where the depots are located are shown in black. The main depot is labeled *D*. The first column shows the depot location using the minimum distance method, while the second column shows the location using the Levy and Bodin's method.

Both initial location methods seek to separate the depots by means of spacing constraints. The bottom row of Figure 2 shows the physical separation of the nodes selected as depots in the network using the minimum distance (MD) method. However, the same separation using the Levy and Bodin's method (LB) results in two depots on the left branch of the depot and none on the right one. This is explained because LB method rewards the importance of the incident edges to the nodes, thus the nodes are selected based on their importance first and the distance between them, second. Prioritizing the physical separation of the nodes over the incident edges, as done by the MD method, may be the cause this method performs better when there are few depots in the network. As the node separation decreases to 0.6 on the middle row and 0.3 in the top row, the distance between nodes becomes less relevant, making the LB method perform better.

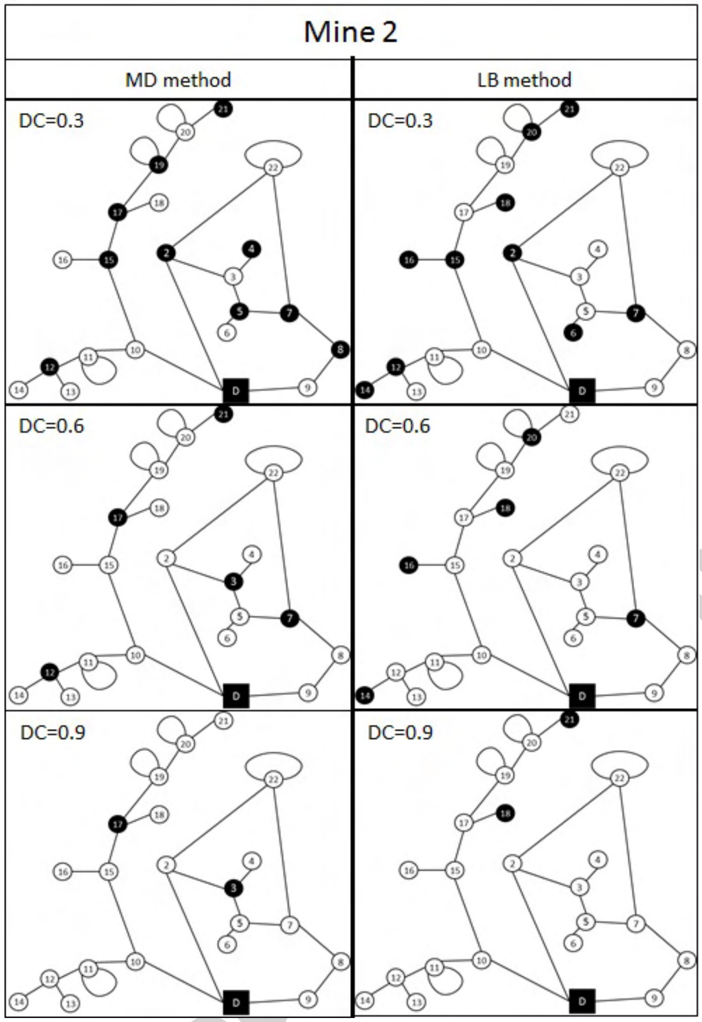

Figure 2: The network *mine2* depot location used for calculating the routing costs

4.3 Improvement of the initial solution

The improvement process was done in two stages. The change in depot location was performed using operations J_1 through J_6 described in Section 3.5. The stopping criterion is $3|N|$ iterations for each operation, where $|N|$ is the number of nodes. After each iteration is performed, the new route is calculated and the node combination that results in the minimum cost route is selected. The process is repeated for the six operations and the depot location of minimum cost among the operations is selected for the next stage.

In the next stage, the allocation and routing of vehicles is changed using the operators of the ALNS algorithm described in Section 3.4 using, as an initial solution, the solution obtained from the previous stage. Table 3 shows the result after the improvement was performed when the initial location of depots was obtained using Levy and Bodin's method. Table 4 shows the improvement of the initial solution obtained using minimum distance method. The first column on both, Table 3 and Table 4, shows the network with 3 and 5 vehicles. The second column shows the cost of the initial solution. We used the number and location of depots using a distance factor $DC = 0.6$. We tested one scenario with 60% initial humidity and the highest evaporation rate. Column three of Table 3 and Table 4 shows the improved solution after performing the change in depot location. The percentage of improvement with respect to the initial solution is shown in column four. Column five of Table 3 and Table 4 shows the improved solution after applying the ALNS algorithm to the solution obtained from location change. Column six of Table 3 and Table 4 show the percentage of improvement of the ALNS solution with respect the solution obtained after the location change. The percentage of improvement after both, the initial location change and the ALNS is shown in column seven of Table 3 and Table 4.

Network	Initial Solution LB a	Location change: Improved Solution b	Location change: Percentage Improvement $(a-b)/a$	ALNS change: Improved Solution c	ALNS change: Percentage Improvement $(b-c)/b$	total Improvement $(a-c)/a$
mine1 3V	2026.29	1941.51	4.18%	1490.63	23.22%	26.44%
mine1 5V	1806.93	1734.63	4.00%	1092.42	37.02%	39.54%
mine2 3V	5100.32	5007.39	1.82%	4505.25	10.03%	11.67%
mine2 5V	4858.78	4718.32	2.89%	4165.05	11.73%	14.28%
mine3 3V	8973.03	8915.00	0.65%	8879.68	0.40%	1.04%
mine3 5V	8728.88	8682.10	0.54%	8673.33	0.10%	0.64%
mine4 3V	6451.06	6355.51	1.48%	6324.42	0.49%	1.96%
mine4 5V	6224.98	6090.40	2.16%	5829.32	4.29%	6.36%
mine5 3V	3822.22	3694.87	3.33%	3321.27	10.11%	13.11%
mine5 5V	3570.17	3443.82	3.54%	3089.46	10.29%	13.46%

Table 3: Solution improvement after location change and allocation-routing change for the Levy-Bodin initial location of depots.

Network	Initial Solution MD a	Location change: Improved Solution b	Location change: Percentage Improvement $(a-b)/a$	ALNS change: Improved Solution c	ALNS change: Percentage Improvement $(b-c)/b$	total Improvement $(a-c)/a$
mine1 3V	2004.18	1941.51	3.13%	1521.50	21.63%	24.08%
mine1 5V	1794.14	1734.63	3.32%	1204.92	30.54%	32.84%
mine2 3V	5058.19	5024.13	0.67%	4754.51	5.37%	6.00%
mine2 5V	4774.03	4717.54	1.18%	4020.63	14.77%	15.78%
mine3 3V	8982.15	8927.77	0.61%	8890.73	0.41%	1.02%
mine3 5V	8769.14	8692.05	0.88%	8692.05	0.00%	0.88%
mine4 3V	6389.12	6359.82	0.46%	6195.97	2.58%	3.02%
mine4 5V	6141.72	6089.55	0.85%	6019.60	1.15%	1.99%
mine5 3V	3710.18	3694.87	0.41%	3364.95	8.93%	9.30%
mine5 5V	3517.32	3436.12	2.31%	2994.52	12.85%	14.86%

Table 4: Solution improvement after location change and allocation-routing change for the Minimum Distance initial location of depots.

In both, Table 3 and Table 4, small networks such as *mine1 3V, 5V*, *mine2 3V, 5V* and *mine5 3V, 5V* have the greatest percentage of improvment. Furthermore, the use of the ALNS adds a greater percentage of improvement to the already modified solution after the change in depot location was performed. The contrary can be observed for instances *mine3 3V, 5V* of Table 3 as well as instance *mine4 3V* of Table 3 and instance *mine4 5V* ot Table 4 where the change in depot location has a greater influence in the improvement of the solution, while the ALNS adds very little to the overall improvement. A special case can be seen in Table 4, instance *mine3 5V* shows no improvement from the ALNS algorithm. The percentage of improvement stays at 0.88%. This behavior suggests that the higher the complexity of the network, the more likely it is to obtain a significant improvement with a relatively simple algorithm such as the one used for depot re location.
Figure 3 ilustrates an example of the initial and improved depot location for instance *mine2 3V*. It is important to notice that both initial location methods aim to separate the depots following distance constraints. Both improved solutions are based on performance of the routing algorithm and how often the selected depots are visited after the allocation and routing rather than the physical separation of depots. This can be shown in the pair of adjacent depots of both improved solutions: Nodes 20 and 21 for the MD method and nodes 15 and 17 for the LB method.

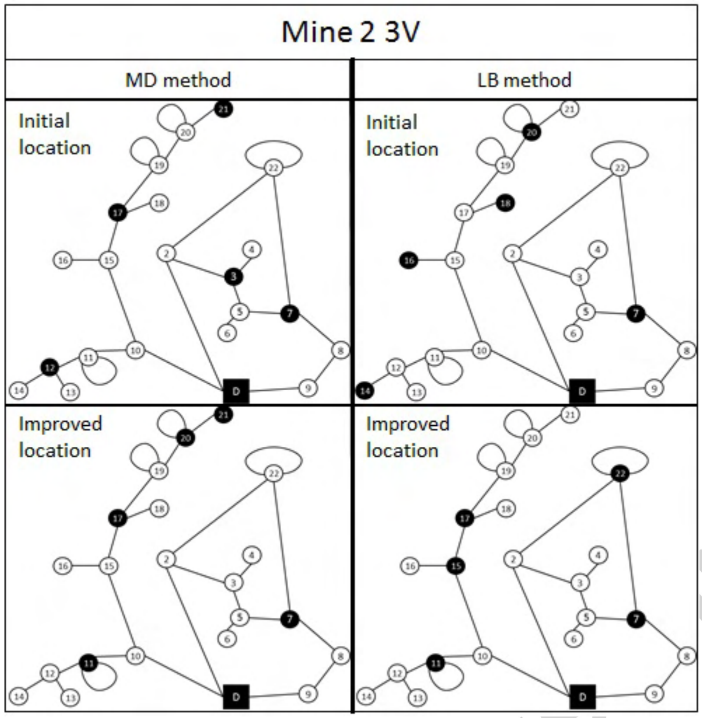

Figure 3: The network *mine2 3V* inital and improved depot location.

5 Conclusion

A mathematical model was developed for the location of water depots in open pit mine networks to improve the road watering operations and increase dust suppression. Two location algorithms were developed for this problem, the Levy and Bodin algorithm and the Minimum Distance algorithm. A vehicle allocation and a routing algorithm were developed based on the location of depots along the network. The initial solution obtained using these algorithms was then improved using an algorithm to change the initial location and an ALNS algorithm to change the initial allocation and routing.

The initial depot location shows better results in terms of routing costs when the Minimum Distance method is used. The Levy and Bodin initial location algorithm performs better with many depots than it does with few depots in terms of routing costs.

The improvement results suggest that if the network has few elements, 20 to 30 nodes and 20 to 35 edges, the use of the ALNS adds a greater percentage of improvement to the solution obtained after the change in depot location. Regarding he improvement in larger networks, more than 49 nodes and 50 edges, a complex algorithm such as the ALNS used to change the allocation and routing of vehicles, has very little influence in the improvement of the value of the objective function after the initial depot location change.

Future work is related to including more elements in the mathematical programming such as vehicle speed and the existence of mining vehicles that can affect the performance of water trucks.

Acknowledgements

This project was partially funded by the National Council of Sicence and Technology (CONACYT) of the Government of Mexico.

The authors would like to thank the reviewers' comments that helped improved the quality of this paper.

References

[1] A. Amaya, A. Langevin, and M. Trépanier. The capacitated arc routing problem with refill points. *Operations Research Letters*, 35(1):45–53, 2007.

[2] F. Chu, N. Labadi, and C. Prins. Heuristics for the periodic capacitated arc routing problem. *Journal of Intelligent Manufacturing*, 16(2):243–251, 2005.

[3] F. Chu, N. Labadi, and C. Prins. A scatter search for the periodic capacitated arc routing problem. *European Journal of Operational Research*, 169(2):586–605, 2006.

[4] L.C. Coelho, J-F. Cordeau, and G. Laporte. Consistency in multi-vehicle inventory-routing. *Transportation Research Part C: Emerging Technologies*, 24(0):270–287, 2012.

[5] L.C. Coelho, J-F. Cordeau, and G. Laporte. The inventory-routing problem with transshipment. *Computers and Operations Research*, 39(11):2537–2548, 2012.

[6] A. Del Pia and C. Filippi. A variable neighborhood descent algorithm for a real waste collection problem with mobile depots. *International Transactions in Operational Research*, 13(2):125–141, 2006.

[7] G. Ghiani and G. Laporte. Eulerian location problems. *Networks*, 34(4):291–302, 1999.

[8] G. Ghiani and G. Laporte. Location-arc routing problems. *OPSEARCH*, 38(2):151–159, 2001.

[9] B.L. Golden and R.T. Wong. Capacitated arc routing problems. *Networks*, 11(3):305–315, 1981.

[10] A. Kansou and A. Yassine. Ant colony system for the periodic capacitated arc routing problem. In *The International Network Optimization Conference*, 2009.

[11] P. Lacomme, C. Prins, and W. Ramdane-Chrif. Evolutionary algorithms for periodic arc routing problems. *European Journal of Operational Research*, 165(2):535–553, 2005.

[12] P. Lacomme, C. Prins, and Ramdane-Chérif W. Evolutionary algorithms for multiperiod arc routing problems. In *Proc. of the 9th Int. Conf. on Information Processing and Management of Uncertainty in Knowledge-Based systems (IPMU 2002)*, pages 845–852, 2002.

[13] L. Levy and L. Bodin. The arc oriented location routing problem. *INFOR.*, 27(1):74–94, 1989.

[14] J-Q. Li, P. Mirchandani, and P. Knights. Water truck routing and location of refilling stations in open pit mines. In *Proceedings of 2008 Australian Mining Technology Conference*, pages 141–156, Sunshine Coast, Australia, 2008. The Australian Institute of Mining and Metallurgy.

[15] R-B. Lopes, C. Ferreira, B-S. Santos, and S. Barreto. A taxonomical analysis, current methods and objectives on location-routing problems. *International Transactions in Operational Research*, 20(6):795–822, 2013.

[16] R-B. Lopes, F. Plastria, C. Ferreira, and B-S. Santos. Location-arc routing problem: Heuristic approaches and test instances. *Computers and Operations Research*, 43(0):309–317, 2014.

[17] F. Marzolf, M. Trépanier, and A. Langevin. Road network monitoring: algorithms and a case study. *Computers and Operations Research*, 33(12):3494–3507, 2006.

[18] J.M. Molina-Martínez, V. Martínez-Alvarez, M.M. González-Real, and A. Baille. A simulation model for predicting hourly pan evaporation from meteorological data. *Journal of Hydrology*, 318(14):250–261, 2006.

[19] I-M. Monroy, C-A. Amaya, and A. Langevin. The periodic capacitated arc routing problem with irregular services. *Discrete Applied Mathematics*, 161(4-5):691–701, 2013.

[20] R. Neulicht and J. Shular. Emission factor documentation for ap-42. section 13.2.2. unpaved roads. Technical report, U.S. Environmental Protection Agency, 1998.

[21] C. Prodhon. A hybrid evolutionary algorithm for the periodic location-routing problem. *European Journal of Operational Research*, 210(2):204–212, 2011.

[22] C. Prodhon and C. Prins. *A Memetic Algorithm with Population Management (MA—PM) for the Periodic Location-Routing Problem*, volume 5296 of *Lecture Notes in Computer Science*, book section 4, pages 43–57. Springer Berlin Heidelberg, 2008.

[23] J-P. Riquelme-Rodríguez, M. Gamache, and A. Langevin. Periodic capacitated arc-routing problem with inventory constraints. *Journal of the Operational Research Society*, 65:1840–1852, 2013.

[24] J-P. Riquelme-Rodríguez, M. Gamache, and A. Langevin. Adaptive large neighborhood search for the periodic capacitated arc routing problem with inventory constraints. Technical report, HEC Montreal, G-2014-01, 2014.

[25] S. Ropke and D. Pisinger. An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science*, 40(4):455–472, 2006.

[26] M-A. Salazar-Aguilar, A. Langevin, and G. Laporte. Synchronized arc routing for snow plowing operations. *Computers and Operations Research*, 39(7):1432–1440, 2012.

[27] M-A. Salazar-Aguilar, A. Langevin, and G. Laporte. The synchronized arc and node routing problem: Application to road marking. *Computers and Operations Research*, 40(7):1708–1715, 2013.

[28] R-J. Thompson and A-T. Visser. The functional design of surface mine haul roads. *The Journal of the South African institute of mining and metalurgy*, 100(3):169–180, 2000.