Modelling tank truck routing for city park watering problems in Surabaya

N I Arini¹, N Siswanto¹* and A Rusdiansyah¹

¹Department of Industrial and Systems Engineering, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia

E-mail: *siswanto@ie.its.ac.id

Abstract. Surabaya City Government tries to have 20% public and 10% private green open space to obey regulation of Indonesian Law No. 26/2007, so that the City Government is actively increasing the number of public green space by creating city parks. With the ever-increasing amount of green open space, maintenance activities are needed, one of which is to do routine watering every day. The problem often faced by the City Government is determining the truck routes to optimize traveling time. Many variables can affect this traveling time, among others, demand, number of vehicles, vehicle capacity, time windows, operating hours and vehicle routes. In practice, a large number of watering locations, irregular routes, and uneven service area assignments often result in overtime which impacts the next shift working hours. The water truck routing problem can be categorized as NP Hard so that the process of scheduling and determining routes use Ant Colony. The objective of this study is to develop a model that can minimize the total routing time and the number of vehicles in the process of filling and watering from the depot to the park. The scenarios developed provides savings in terms of time minimization and distribution of assignments.

1. Introduction

According to Indonesian Law No. 26/2007, the city area should have minimum 30% of green open space, consisting 20% of public and 10% of private. To carry out this regulation, the Surabaya City Government is actively focusing on the addition number of parks, while keeping maintaining the existing parks by watering them every day. One of the problems faced by the city government in watering activities is determining the number of vehicles used to fulfill the needs of watering plants at each point, where each park has a different demand. In addition to demand at several points, there are still other variables which effects this activity such as vehicle capacity, shift time limits, operational working hours and vehicle routes. The variation and irregular values of these variables often results in employee overtimes or vehicles returning late to the rayon office. These overtime and lateness result in disruption of the next work shift (afternoon shift) because the allocation of available vehicles is only 1 vehicle per district.

The watering truck routing problem is the extension of Vehicle Routing Problem (VRP). While the VRP is a problem of which the vehicle visiting each point only once, the problem needs to be split the service so that the division of tasks between vehicles is evenly distributed. So, it needs additional constraints, especially the time windows constraints. Therefore, the problems in this study can be classified as a models of Vehicle Routing Problem with Time Windows (VRPTW) by adding split service.

The approach used to solve the problem is an Ant Colony Optimization (ACO), a meta-heuristic method, by considering a number of parks that must be visited by vehicles before they are returning to the depot within a time limit shift. The use of ACO algorithm is expected to help find the best value with faster computing time than the one of exact algorithm.
2. Literature Review
In previous studies, routing research related to picking up and deliveries were carried out by [1] regarding the customer requests fulfillment for delivery and pickup of goods in one trip. Latiffianti et. al [2] develops exact algorithm for solving pickups and deliveries of workers from several points of origin and several destinations with time windows called as split delivery vehicle routing problem (SDVRP). Dayanara et. al [3] also develops an ACO model for waste collection in the city of Surabaya. In Dayanara’s problem, the waste collection can be pick up splitly within the time window by using an intermediate facility.

Another variant, called as VRP Delivery and Pickup (VRPDP), was conducted by [4] solved by using a Tabu Search approach based on variations in single demand. The problem of VRPDP basically models a set of vehicles which visit to each node are done exactly once. However, in some cases, nodes can be visited more than once (split service) if there are significant savings, distance, time, cost, and the number of routes compared to without the split service option. The extention of this problem was conducted by [5] concerning multiple node visits which can be called as Split Delivery Vehicle Routing Problems with Time Windows (SDVRPTW).

Similar research was conducted by [6] using the Branch and Cut algorithm to solve SDVRPTW problems which produce an optimal output of 5% compared to SDVRPTW with exact calculations. Other research on SDVRPTW was also conducted by [7] which compared the method of solving the problem using Ant Colony Optimization (ACO) and Hybrid Meta-heuristic Algorithm which is a combination of ACO, Genetic Algorithm (GA), and heuristics. This study tested the three meta-heuristic algorithms on several datasets used in several previous studies [8]. The test results obtained promising results for ACO, where 14 of the 21 datasets used in the study of [8] yield better values than the other two algorithms. Another meta-heuristic methods called Crow Search Algorithm is also implementated in solving complex scheduling problem as discussed in [9]. This research concluded that the Crow Search Algorithm overtake other algorithms such as the combination of Particle Swarm Optimization and Simulated Annealing (PSO + SA), hybrid Genetic algorithm (hGA) in term of computational time and optimum solution.

3. Methodology
3.1 Data Collecting
This study utilises data from direct observation, Google Maps application as well as the historical data. The detailed data as follow:

1. The location and the length of 46 parks in east part of Surabaya.
2. The location of 40 water filling locations (rivers).
3. The number and capacity of tank trucks.
4. The operating hours when vehicles will be departed from the depot at 06.00 AM and must return back before 03.00 PM.
5. Various travel time data taken from Google Maps, which includes data from depot to park, between parks, park to each river, depot to river and vice versa.
6. Loading and unloading service time are assumed to be constant for 15 minutes and 20 minutes, respectively for each 5000 Liter.
7. Route and assignment of tank trucks
3.2 Data Processing

As seen in Figure 1, the truck departs from the depot with empty tank and fills the water in the nearest river (pickup node). Then, the vehicle goes to the nearest park for watering. For park that have demand less than or equal to truck capacity, trucks only need once to do watering (unloading). However, when the demand for a park is more than the truck capacity, the park can be visited several times until all demands are met, either with the same or different trucks. For example, as seen in Figure 1, vehicle 1 initially fills with water in river 2 and unload it in park 1. Because the demand for park 1 exceeds the capacity of truck 1, then truck 1 returns to refill water in the nearest river, in this case at river 3. After all, 1 park demand is fulfilled, then truck 1 continues the journey to the next watering location. This means that in this case, park 1 is served twice by truck 1.

3.2.1 Ant Colony Optimization for Watering Routes for City Parks

There are two main stages for developing ACO model to solve this problem, namely the construction of Traveling Salesman Problem (TSP) routes and improvement routes by dividing into several sub-routes.

1. Construction of a TSP route using the Ant Colony Optimization approach

   The process of TSP construction routes with the Ant Colony Optimization algorithm is carried out in several stages as seen in Figure 2.
Start

Data Input
Demand for each parks; Travel time (between depot and each park, between each park, between each park and each river, between depot and each river); Operational time (filling and watering); Number of vehicle;

Parameter Input
Maximum of iteration, Number of Ants Group, Initial Pheromone, Pheromone Index, Trail Index, Saving Index, Evaporation Ratio, Number of Candidate, Time Windows

Set ant group
Set vehicle
Set route starts from depot
Adjust travel time matrix
Adjust capacity
Filling (Pickup)

Yes
Take pheromone data, trail visibility, saving index
Calculate probability
Select the candidate node using the roulette wheel

Yes
Time windows checking, is it still enough?

No
Adjust remaining demand
Adjust total travel time

Loop VRPPD

All the parks have been visited?

Yes
Calculate the total travel time of the ant group

No
All ant groups have been run

No
Update pheromone
Maximal iteration?

Yes
VRPPD Route with Minimum Travel Time

End

Figure 2. Flowchart of developing ant colony optimization algorithm.
a. Initialization
   o Data:
     a) Travel time from depot to the park.
     b) Travel time from depot or park to the river.
     c) Travel time between two parks.
     d) Loading and unloading time.
   o Parameter:
     a) Early pheromones ($\tau_{ij}^1$)
        The pheromone value will always be updated in every algorithm iteration, starting
        from the first iteration until the maximum iteration is determined or has reached
        optimal results.
     b) Visibility Index ($h$)
        The visibility value between nodes is the inverse of the distance ($1/distance$) of each
        node.
     c) Evaporation ratio ($\rho$)
        The evaporation ratio is the rate of pheromone evaporation which has value $0 < \rho < 1$.
     d) Number of ant groups ($N$)
        The ACO algorithm is a population-based metaheuristic, so the search for solutions
        directly based on certain groups. The specified number of ant groups will be run in
        one iteration. There is no definite provision regarding the number of ant groups to
        produce an optimal value. So, it takes several trials to get the best results.
     e) Pheromone level index ($\alpha$) and visibility index ($\beta$)
        The $\alpha$ value is the pheromone weight $\tau$ and $\beta$ is the weight that controls the visibility
        of the pheromone level. To simplify the calculation, the values of the two indices are
        equalized to 1.
     f) Number of candidates
        To facilitate the selection of the next node for each vehicle, selected candidate nodes
        are created. For candidate nodes that have been selected by one vehicle, they will not
        be selected by another vehicle.
     g) Maximum iteration
        The maximum number of iterations is fixed as long as the algorithm runs. For the
        optimal value of a maximum iteration, several trials need to be done to get the best
        results.

b. Probability and selection of the next node

   Calculation of the probability to select a segment using $P = \frac{\tau_{ij}^2 h_{ij}^\alpha}{\sum \tau_{ij}^2 h_{ij}^\alpha}$. For each segment, a
   cumulative probability range is related to the segment selection. Then certain values are chosen
   based on random numbers in the range (0,1). Specific segments will be selected based on
   random numbers generated. After each ant occupies each of the specified points, the ant will
   start to travel from the first point of each as a point of origin and head for one of the other points
   as a destination. Then from the second point of each, the ant will continue its journey by
   selecting one of the points that have not been visited as the next destination point. Points that
   have been served will be stored in a taboo list so that they are not visited again. The ant journey
   continues until all points are visited one by one.

c. Calculation of travel time

   The total travel time is calculated after an ant has completed the entire route and returned to the
   depot, starting from the depot to the river, river to park, park to park, and park to depot. The
   total travel time value is stored in memory as optimal local (lk).
d. Adjust pheromone
   After all groups are run, adjust the pheromone values $\tau_{n+1} = (1 - \rho)\tau_n + \rho\tau_0$

e. Iterate
   For each iteration that is done, the taboo list is first emptied to be refilled in the order of new points in the next iteration. Then iterates for stages (b) to (d) until it reaches a maximum iteration or has reached convergence. From the overall iteration will be obtained the optimal global value of the minimum total travel time.

2. Dividing VRP Sub-Routes
   The following is an algorithm carried out in finding route recommendations for watering city parks in Surabaya. The results of the Ant Colony Optimization TSP will be used as a basis for making VRP sub-routes that have been adapted to existing constraints.

a. Initialization
   - Data used as input on this VRP sub-route are:
     1. The optimum TSP route from the previous ACO TSP
     2. Travel time from the depot to the park
     3. Travel time from park to park
     4. Travel time from the park to the river
     5. Travel time from the depot to the river
     6. Water withdrawal operational time
     7. Watering operational time
     8. Number of trucks
     9. Demand of park
     10. Truck capacity
     11. End time shift
   - Initialize the truck to be used and the park visited

b. Truck Selection
   In this case, there are nine units of trucks with 5000-liter tank capacity. From a total of nine trucks, there will be a selection of candidates for the number of parks that can be visited first. For each park that has been selected as a candidate by one truck, it cannot be chosen by another truck. The process of selecting park candidates is to divide the park into four sections. Out of 46 parks, there are 12 park candidates. The purpose of selecting candidates is so that the directions of the vehicles are spread.

c. Selection of starting point
   The choice of a starting point for each scenario always starts from the depot. For every truck at the starting point, it has a capacity = 0. Each truck will return to the starting point after all the parks have been visited and the deadline for operating hours has finished.

d. Park selection
   In this study, 46 parks have to be visited. Each park has different demands depending on the size of the park. There is no time limit for a visit to the park or in other words the park is free to visit at any time. Park selection relates to the smallest travel time from the last position of the truck by considering the capacity and the time remaining.

e. Selection of watering point
   In the selection of watering points, considering the travel time from the starting point with the watering point and saving weight to determine which point will be visited next. For the next selection of visit points, initially, the closest potential points are determined based on travel time, then a list of saving values for each potential point candidate is made, from the largest to the smallest. The point that has the shortest time and the highest saving value will be chosen for
the next visit. This point selection follows the route produced by TSP ACO. Then see the travel time and capacity, is it possible to visit the point. If the truck's capacity does not meet the demand at the watering point, a visit to the nearest water filling point is made. But if the time does not meet, the truck will return to the depot. In this process, it is possible to have a split service.

f. Selection of water uptake
A visit to the water collection point is a soft constraint, so this visit is carried out if the truck capacity = 0. There is no specific process in selecting the water filling point because it only considers the smallest travel time from the last truck location and the next watering point.

4. Result and Discussion
The algorithm is tested using Intel® core™ i5-2450m CPU @ 2.50GHZ 4 GB RAM. The parameters used in this testing as follow: the evaporation value \( \rho \) = (0.3; 0.6; 0.9), the number of ants = (5; 10) and the maximum iteration of (10; 100; 500). Running models are carried out by 10 replications and the best objective function value is selected from these ten replications. Table 1 shows that the minimum, average, and standard deviation values of ten replications for these various parameters.

| \( \rho \) | M   | I   | Min  | Mean  | SD   | \( \rho \) | m  | i   | Min  | Mean  | SD   |
|----|----|----|------|-------|------|----|----|----|------|-------|------|
| 0.3 | 5  | 10 | 3890.0 | 3911.0 | 9.5  | 0.6 | 5  | 10  | 3727.1 | 3740.0 | 11.5 |
|     | 9  |    | 471.5  | 66.3   | 358  | 421.3 | 37.4 |
| 50  | 3634.9 | 3647.6 | 12.2  | 50  | 3640.0 | 3647.4 | 6.6  |
|     | 8  |    | 1661.6 | 120.8  | 1843 | 1889.6 | 49.3 |
| 100 | 3639.6 | 3644.6 | 3.8   | 100 | 3595.7 | 3626.5 | 14.3 |
|     | 8  |    | 4255.6 | 110.6  | 3200.0 | 4362.7 | 146  |
| 0.3 | 10 | 10 | 3657.8 | 3688.2 | 16.8 | 0.6 | 10 | 10  | 3681.1 | 3696.7 | 10.8 |
|     | 8  |    | 1033.9 | 5.2    | 625.0 | 676.4 | 66.4 |
| 50  | 3611.9 | 3630.4 | 14.5  | 50  | 3611.5 | 3621.3 | 6.9  |
|     | 8  |    | 3469.2 | 123.1  | 3060.0 | 3400.6 | 236  |
| 100 | 3618.1 | 3621.5 | 2.5   | 100 | 3623.5 | 3624.6 | 1.0  |
|     | 8  |    | 5852.1 | 369.4  | 6405.0 | 6466.4 | 45.4 |
| 0.9 | 5  | 10 | 3635.5 | 3659.4 | 12.8 | 0.9 | 10 | 10  | 3727.1 | 3740.0 | 11.5 |
|     | 8  |    | 2045.4 | 696.6  | 358  | 421.3 | 37.4 |
| 50  | 3635.5 | 3650.5 | 13.4  | 50  | 3640.0 | 3647.4 | 6.6  |
|     | 8  |    | 1492.5 | 17.3   | 1843 | 1889.6 | 49.3 |
| 100 | 3620.3 | 3640.0 | 16.0  | 100 | 3595.7 | 3626.5 | 14.3 |
|     | 8  |    | 3206.4 | 177.1  | 3200.0 | 3362.7 | 146  |

Based on the results on Table 1, it can be concluded that the greater the number of ant groups and the number of iterations, the higher the required average constant computing time. While the greater the value of evaporation does not constantly affect the required computing time. The objective function of the problem, total travel time, it can be achieved by the parameter \( \rho = 0.6 \) and the number
of ant groups $m = 5$ and the iteration is 50, with a minimum travel time of 3611.5 minutes and an average travel time of 3621.3 minutes.

From above trials and considerations, a new experiment is conducted which uses the evaporation coefficient parameter $\rho = 0.6$, the number of ant groups $m = 5$ and the number iteration $i = 50$ with ten replications. The result of this experiment can be seen in Table 2.

| No | Objective Function | Number of Vehicle | Time   |
|----|-------------------|-------------------|--------|
| 1  | 3576.1            | 8                 | 3851   |
| 2  | 3564.2            | 8                 | 3802   |
| 3  | 3560.5            | 8                 | 3822   |
| 4  | 3558.6            | 8                 | 3840   |
| 5  | 3562.3            | 8                 | 3848   |

| No | Objective Function | Number of Vehicle | Time   |
|----|-------------------|-------------------|--------|
| 6  | 3566.9            | 8                 | 3835   |
| 7  | 3560.7            | 8                 | 3839   |
| 8  | 3559.1            | 8                 | 3860   |
| 9  | 3559.5            | 8                 | 3857   |
| 10 | 3558.9            | 8                 | 3844   |

Objective function (in minutes), number of vehicle (in units), and computational time (in seconds).

The assignment of the test results using the ACO model does not violate existing constraints in terms of capacity or time window with all demands in each park can be served all. The total number of vehicles available at the city is currently 9 units, while the recommendation of this study only requires eight units of vehicles. So, the demands of each park will be able to be served all of them. The city also still have a spare tank truck that will stay at the depot as a backup vehicle in the event of an increase in the number of demands or when there is a vehicle which has break down condition. The back up vehicle is very important due to every day the park must always be watered. Otherwise, the survival of plants as a green space in the city of Surabaya will be disrupted. This optimization result with the 3558.6 minutes total traveling time is better than the one of existing condition, with the 3866 minutes.

5. Conclusion
The development of ACO model for been applied for finding the truck routing in watering parks in the city of Surabaya. The model tries to determine the routes of the vehicles and the number of vehicles needed. The parameters used in the ACO model test for watering parks in the city of Surabaya were obtained from a trial of several parameters with 10 replications in each combination. The results of the selected parameters are parameters that have a small standard deviation can provide the minimum objective function value. The recommended route generated by the ACO algorithm model with the parameter evaporation value 0.6, the number of ants 5 and a maximum number iteration of 50 for watering parks in the city of Surabaya produces a goal function value of 3558.6 minutes by using 8 units of vehicles. This assignment does not violate existing contraints in terms of vehicle’s capacity or time window with all demands in each park can be served all.

References
[1] Desaulniers G, Desrosiers J, Erdmann A, Solomon MM, Soumis F. VRP with Pickup and Delivery. The vehicle routing problem. 2002 Jan 1;9:225-42. doi.org/10.1137/1.9780898718515.ch9.
[2] Latiffianti E, Siswanto N, Firmandani RA. Split delivery vehicle routing problem with time windows: a case study. IOP Conference Series: Materials Science and Engineering. IOP Publishing; 2018 Apr;337:012012. doi.org/10.1088/1757-899X/337/1/012012.
[3] Dayanara DH, Arvitrida NI, Siswanto N. Vehicle Routing Problem with Split Service, Time Window and Intermediate Facility for Municipal Solid Waste Collection in Surabaya City with Ant Colony Optimization Algorithm. *IOP Conference Series: Materials Science and Engineering*. IOP Publishing; 2019 Sep 6;598:012020. doi.org/10.1088/1757-899X/598/1/012020.

[4] Wassan N, Nagy G. Vehicle Routing Problem with Deliveries and Pickups: Modelling Issues and Meta-heuristics Solution Approaches. *International Journal of Transportation*. NADIA; 2014 Apr 30;2(1):95–110. doi.org/10.14257/ijt.2014.2.1.06

[5] Toth P, Vigo D, editors. Problems, Methods, and Applications, Second Edition. *Vehicle Routing*. Society for Industrial and Applied Mathematics; MOS-SIAM Series on Optimization; 18. 2014. 463 p. ISBN 9781611973587.

[6] Bianchessi N, Irnich S. Branch-and-Cut for the Split Delivery Vehicle Routing Problem with Time Windows. *Transp. Sci*. Institute for Operations Research and the Management Sciences (INFORMS); 2019 Mar;53(2):442–62. doi.org/10.1287/trsc.2018.0825.

[7] Rajappa GP, Wilck JH, Bell JE. An Ant Colony Optimization and Hybrid Metaheuristics Algorithm to Solve the Split Delivery Vehicle Routing Problem. *International Journal of Applied Industrial Engineering*. IGI Global; 2016 Jan;3(1):55–73. doi.org/10.4018/ijaie.2016010104.

[8] Derigs U, Li B, Vogel U. Local search-based metaheuristics for the split delivery vehicle routing problem. *J Oper Res Soc*. Informa UK Limited; 2010 Sep;61(9):1356–64. doi.org/10.1057/jors.2009.100.

[9] Adhi A, Santosa B, Siswanto N. A meta-heuristic method for solving scheduling problem: crow search algorithm. *IOP Conference Series: Materials Science and Engineering*. IOP Publishing; 2018 Apr;337:012003. doi.org/10.1088/1757-899X/337/1/012003