Ant Colony Optimization Solutions for Logistic Route Planning with Pick-up and Delivery

Eric Hsueh-Chan Lu*, Ya-Wen Yang and Zeal Li-Tse Su
Department of Geomatics, National Cheng Kung University
No.1, University Rd., Tainan City 701, Taiwan (R.O.C.)
*Correspondence: luhc@mail.ncku.edu.tw; v47244724@gmail.com; zeal0820@gmail.com

***Abstract*—Online shopping behaviors lead to a large number of goods need to be transported in the real world. Researches on logistics have attracted extensive attentions. One of popular topics is logistic route planning. Although various previous studies have discussed some classical routing problems, real logistic constraints are not considered such as the vehicle capacity, various logistic requirements, etc. Thus, these solutions may not be directly applied to the logistic route planning. In this paper, we propose a novel solution based on *Ant Colony Optimization* (*ACO*) to find high quality logistic routes not only meeting real logistic constraints but also taking pick-up and delivery requirements into account. To the best of our knowledge, this is the first work using *ACO* to plan the logistic routs that considers various logistic requirements, simultaneously. Through extensive experimental evaluations by a semi-real logistic dataset, the proposed *ACO*-based solution was shown to deliver excellent performance.**

Keywords—Logistics; Route Planning; Ant Colony Optimization; Data Mining.

I. Introduction

In recent years, more and more online shopping websites have sprung up. The change of online shopping behaviors lead to the logistics industry is flourishing. A large number of goods need to be delivered to the corresponding consumers anytime anywhere. *7-ELEVEN* is the leading chain store in *Taiwan*. The subsidiary *President Transnet Corp.* which is a logistic company grows rapidly. The growth rate of a year exceeds 10% in recent five years. In *Taiwan Chunghwa Post Company*, the revenue ratio of goods logistic businesses has exceeded that of traditional postal services. Hence, researches on logistics-related issues have received a lot of interests in both of the industry and academia. Among them, one of active topics is the logistic route planning. *Traveling Salesman Problem* (*TSP*) [10] is a well-known route planning problem for multiple destinations. The goal of *TSP* is to find the shortest route that visits each customer exactly once and returns to the start location. Although a number of works [9, 13] on *TSP* have been studied in the previous research, the concepts and solutions may not be directly applied to the logistic route planning problem because the total volume of delivered goods usually exceeds the vehicle capacity very much.

Vehicle Routing Problem (*VRP*) [11] is more similar to the logistic route planning problem. Given a vehicle with a limited capacity, *VRP* wants to deliver a set of goods to customers meeting the vehicle capacity constraint and minimize the routing distance. Most of previous works adopted some *Evolutionary Algorithms* (*EAs*) such as *Simulated Annealing* [11], *Genetic Algorithm* [8, 12], and *Ant Colony Optimization* [3] for solving *VRP* problem. Although *VRP* has considered the capacity constraint of a vehicle, it only focuses on delivering goods to the customers. As people get increasingly busy, to pick up goods from customers has become an emerging logistic service. *Vehicle Routing Problem with Backhauls* (*VRPB*) [4] which is an extension of *VRP* not only delivers goods to customers but also picks up goods from customers. In *VRPB*, the delivery and pick-up targets are called linehaul and backhaul customers, respectively. The goal of *VRPB* is to serve each linehaul and backhaul customer from a central warehouse and the vehicles with a limited capacity. Although a number of studies [1, 2, 4, 6] have been proposed to solve *VRPB*, most of them considered that the backhaul tasks cannot start until all of the linehaul tasks are completed. However, we consider that the routing cost should be reduced if the pick-up and delivery tasks can be handled at the same time.

In this paper, we propose a novel solution based on *Ant Colony Optimization* (*ACO*) for solving the problem of logistic route planning with pick-up and delivery and making the total routing cost as low as possible. We first adopt *Greedy Randomizes Adaptive Search Procedure* (*GRASP*) to generate initial logistic routes which is called an initial solution for initialize the pheromone distribution of logistic graph. Then, we simulate a set of ants to search better solutions iteratively. A solution generated by an ant represents a set of logistic routes that can well deliver and pick up goods among customers. *Exploitation* and *Exploration* are two operations to decide how to assign goods to vehicles. To guarantee the qualities of generated solutions in each iteration, we adopt *NEH* algorithm [3] to refine the generated routes. Finally, the initial solution and the best solution in each iteration would update the pheromone of logistic graph. *ACO* iteration stops when the pheromone of logistic graph is stable or the number of iterations reaches user-specific value of generations. Thus, the logistic routes with pick-up and delivery can be obtained. To the best of our knowledge, this is the first work on logistic rout planning with pick-up and delivery using *ACO*-based solution. Through extensively experimental evaluations by a semi-real logistic dataset obtained from *KERRY TJ Logistics* [5], we show that our proposed *ACO*-based solution delivers an excellent performance in terms of the route quality.

The remainder of this paper is organized as follows. In Section II, we first briefly introduce the related studies on route planning. Next, we formulate the problem in Section III. In Section IV, we introduce the proposed *ACO*-based solution. In Section V, we evaluate system performance by a semi-real logistic dataset. Finally, we summarize this work and provide some possible future work in Section VI.

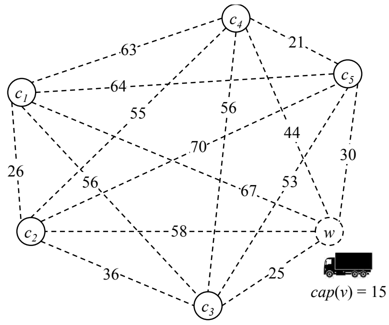
II. RELATED WORK

In 1930s, Karl Menger *et al.* first proposed *Traveling Salesman Problem* (*TSP*) [10] to model a salesman visits each customer exactly once and finally returns to the starting location. However, the *TSP* did not consider the vehicle capacity of logistics and cannot be applied to logistic route planning problem directly. *Vehicle Routing Problem* (*VRP*) is a kind of scheduling problem and more similar to the logistic route planning problem. *VRP* aims at finding a set of routes for visiting customers by the vehicles with a limited capacity. Most of previous work adopted *Evolutionary Approaches* (*EAs*) to *VRP* such as the simulated annealing, tabu search algorithms [11], *Genetic Algorithm* (*GA*) approach [12] and *Ant Colony Optimization* (*ACO*) [3] to solve *VRP*. To consider the situations of multiple depots in real logistic cases, Lim *et al.* proposed two-stage and one-stage methodologies [7] and showed the one-stage algorithm outperforms the published methods by experimental results. In [8], Li-xia proposed the hybrid *GA* based on fuzzy simulation to effectively solve the vehicle routing model. Although *VRP* considers real logistic constraints, it only focuses on how to deliver goods to customers. To pick up goods from customers becomes an emerging logistic service as people get increasingly busy.

Vehicle Routing Problem with Backhaul (*VRPB*) is an extended version of *VRP* and focuses on not only delivery to customers but also pick-up from customers. The goal of *VRPB* is to accomplish each vehicle services both delivery points (linehaul customers) and pick-up points (backhaul customers). In [4], Goetschalckx *et al.* proposed a two-phased solution based on space filling curve heuristics to derive an initial solution to the linehaul-backhaul problem. In [1], Belloso *et al.* provides the *VRP with Clustered Backhauls* (*VRPCB*) that considers the delivery customers has to be served before the first pickup customer can be visited. Battarra *et al.* modeled the *Split VRPCB* (*SVRPCB*) [6] and used an *Integer Linear Programming* formulation and *Adaptive Guidance* meta-heuristic for solving *SVRPCB*. In [2], Bortfeldt *et al.* extended *VRPCB* to an integrated routing and three-dimensional loading problem, called *VRPCB with 3D loading constraints* (*3L-VRPCB*). This paper developed two hybrid algorithms for solving *3L-VRPCB*. However, most of previous works considered that it is not allowed to start the backhaul tasks until all of the linehaul tasks are completely delivered. We think that the routing cost should decreases if the vehicle can stop by some of backhaul customers to pick up goods when delivering goods to other linehaul customers.

III. PROBLEM STATEMENT

Based on a logistic environment, given a warehouse w, a set of customers $C = \{c_1, c_2, ..., c_n\}$, a vehicle v, a set of delivery goods $G^{del.}$ and pick-up goods $G^{pic.}$, the goal of this research is to plan a set of logistic routes $\{\varphi_1, \varphi_2, ..., \varphi_m\}$ such that the total routing costs $\Sigma_{\forall 1 \le i \le m}\, \delta(\varphi_i)$ as small as possible and all of the goods $g \in (G^{del.} \cup G^{pic.})$ are successfully transported. The route $\varphi = \langle w, g_1, g_2, ..., g_m, w\rangle$ must be generated by a non-overloaded vehicle and routing cost $\delta(\varphi)$ is defined as the sum of transportation



$G^{del.}$	*Start*	*Dest.*	*Vol.*
g_1	w	c_1	4
g_2	w	c_3	7
g_3	w	c_5	5

$G^{pic.}$	*Start*	*Dest.*	*Vol.*
g_4	c_1	w	5
g_5	c_2	w	8
g_6	c_4	w	2

Figure 1. An example of logistic environment.

distances as shown in (1), where the function $ED(p, q)$ indicates the *Euclidean Distance* between p and q.

$$\delta(\varphi) = ED(w, c_1) + \sum_{i=1}^{m-1} ED(c_i, c_{i+1}) + ED(c_m, w) \qquad (1)$$

Figure 1 shows a logistic environment which contains a warehouse w and 5 customers (c_1 to c_5). There are totally 6 goods need to be transported (3 delivery and 3 pick-up), where *vol.* indicates the volume of a goods. The capacity of vehicle is 15. The vehicle can first loads g_1 and g_3 at w, delivers g_3 to c_5, then delivers g_1 and picks up g_4 at c_1 at the same time, picks up g_5 from c_2, and finally return to w. The first valid route $\varphi_1 = \langle w, g_3, g_1, g_4, g_5, w\rangle$ and $\delta(\varphi_1) = 30 + 64 + 0 + 26 + 58 = 178$. After the first transportation, the remaining goods are g_2 and g_6. The vehicle can load g_2 at w, deliver g_2 to c_3, pick up g_6 from c_4, and return to w. The second valid route $\varphi_2 = \langle w, g_2, g_6, w\rangle$ and $\delta(\varphi_2) = 25 + 56 + 44 = 125$. The total routing cost is $178 + 125 = 303$.

IV. PROPOSED METHOD

Due to the logistic route planning is a NP-hard problem, how to find the approximate logistic routes within an acceptable search time is an important issue. One of well-known solutions for solving NP-hard problems is *Ant Colony Optimization* (*ACO*) which is based on iterative and probabilistic concepts. *ACO* may be suitable for logistic route planning problem if the travelling paths of ants are considered as the logistic routes. Although the *ACO*-based study [3] has been proposed to solve logistic route planning problem, it only focuses on delivering goods to the customers. As people get increasingly busy, to pick up goods from customers has become an emerging logistic service. Based on the above motivations, we propose the *ACO*-based solution for solving logistic route planning problem. We consider that there are two issues that are worthy to be taken care. 1) Unlike the simple delivery logistics, the new logistics with pick-up and delivery needs to additional consider whether the vehicle has sufficient space to pick up goods from customers. 2) The travelling behaviors of ants in *ACO* is based on the pheromone of logistic network. In [3], the pheromone is related to the total routing distance of each vehicle. However, the topology of logistic graph is ignored. We think that the pheromone integrating the edge distances of graph benefits for improving the quality of route planning.

A. System Framework

Figure 2 shows the system framework of this research. A logistic environment consists of a logistic graph G_L that represents the city road network, a warehouse that is used for goods distribution and several customers who may have pick-up or delivery requirements. The warehouse has

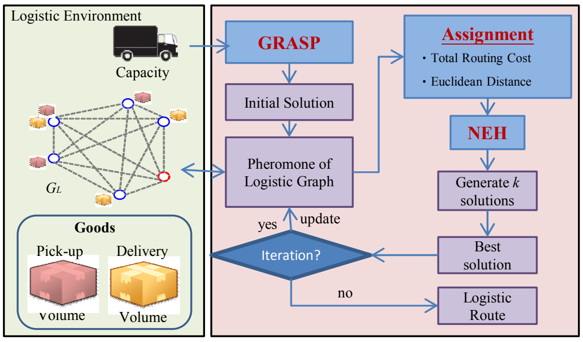


Figure 2. System Framework.

a set of goods that need to be delivered to one of the customers and a vehicle with a limited capacity. In the proposed *ACO*-based solution, we consider that a better initial solution benefits for efficiently finding the high-quality final solution. Hence, *GRASP* method is first used to generate the initial logistic route and initialized the pheromone of logistic graph. Then, we use *ACO* to iteratively find better solutions based on the best solution in the last generation. For each iteration, all goods would be assigned to a suitable vehicle according to the graph pheromone and distance factors. Besides, all logistic routes generated by *ACO* are resorted by *NEH* optimization method. The so far best solution would be used for updating the pheromone of logistic graph. The solutions would be better after a number of pheromone updates and iterative optimizations. *ACO* stops when the pheromone of logistic graph is stable or the number of iterations reaches user-specific value of generations. The best solution in the last iteration would be considered as the final logistic route.

B. GRASP

In *ACO*, the generated solution depends on the pheromone distribution of logistic graph in each iteration. A better initial pheromone distribution would influence the efficiency for finding high-quality solutions. Hence, we adopt *Greedy Randomizes Adaptive Search Procedure* (*GRASP*) to generate initial logistic routes which is called initial solution before *ACO*. Besides, this initial solution is used for generate the initial pheromone distribution of logistic graph. Starts from the warehouse, the *Restricted Candidate List* (*RCL*) is designed for storing the k nearest not yet processed goods which is called candidate. Next, one of goods in *RCL* is randomly selected as the next logistic goods. Based on the location of goods, we search other k nearest not yet processed goods and add them into *RCL*. It is worth noting that *GRASP* needs to check whether the generated logistic route meets vehicle capacity constraint or not. Repeats the same procedure until all goods are processed. The visiting order is considered as the initial logistic routes.

Take Figure 1 as an example. Figure 3 shows the details *GRASP*. Suppose that the length of *RCL* is 3. (*a*) Based on the warehouse w, we add the 3 nearest goods are g_2, g_3 and g_6 into *RCL*. Next, suppose g_6 is randomly selected from *RCL* and considered as the next processed goods. Thus, the first generated route is $<w, g_6, w>$. This is a valid route because the volume of g_6 does not exceed the capacity of vehicle. (*b*) Then, the 3 nearest goods from g_6 are g_1, g_3 and g_4. We randomly select g_1 as the next processed goods and generate the route $<w, g_6, g_1, w>$. This is also a valid

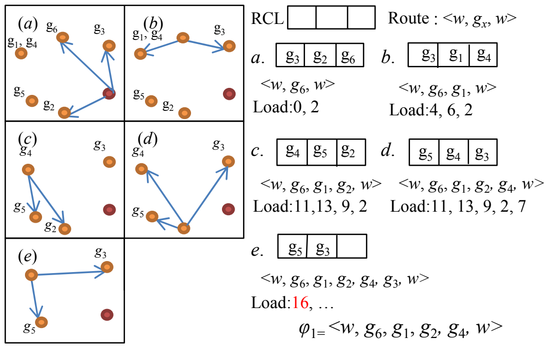


Figure 3. The initial solution generated by *GRASP*.

route since the vehicle loading at w, c_4 and c_1 respectively are 4, 6 and 2. (*c*) Repeat the same procedure, the 3 nearest goods from g_1 are g_2, g_4 and g_5, and g_2 is randomly selected to add the route. (*d*) The 3 nearest goods from g_2 are g_3, g_4 and g_5, and g_4 is randomly selected to add the route. Until now, the route is $<w, g_6, g_1, g_2, g_4, w>$. (*e*) The 3 nearest goods from g_4 are g_3 and g_5, however, the total volume would exceed the vehicle capacity if g_3 is added into the route. Hence, we obtain the first route $\varphi_1 = <w, g_6, g_1, g_2, g_4, w>$. The remaining goods are formed as the second route $\varphi_2 = <w, g_5, g_3, w>$. The two routes are called an initial solution. How to initialize the graph pheromone based on the initial solution would be described later since each iteration of *ACO* will update the pheromone. Hence, we first introduce the procedure of an iteration.

C. Assignment

In this phase, we introduce how to apply *ACO* to logistic route planning problem. In each iteration, we simulate a set of ants to find solutions. A solution is generated by an ant that represents a set of logistic routes that can well deliver and pick up goods to each customer. *Exploitation* and *Exploration* are two operations to decide how to assign goods to vehicles. *Exploitation* depends on the pheromone in the logistic graph. *ACO* thinks that two goods should be processed consecutively if they are usually planned in a consecutive way. However, it is easy to fall into the local optimal problem if only *Exploitation* is used. To avoid such situation, *ACO* also designs *Exploration* to proportionally select other goods as the next target according to the pheromone and other impact factors.

1) Exploitation

Based on the locations of each vehicle and unprocessed goods, *Exploitation* would find the best pair of vehicle and goods such that the pheromone multiplied by a distance-based factor μ is maximal. As (2) shows, $pher(v_k, g_n)$ represents the pheromone between the current location of the k^{th} vehicle and the goods g_n, α and β respectively indicate the impact factors of pheromone and distance-based factor. As (3) and (4) shows, we consider two kinds of distance-based factors: 1) ACO_{TRD} that is defined as the reciprocal of *Total Routing Distance* of a vehicle μ_{TRD} [3] and 2) ACO_{ED} that is defined as the reciprocal of *Euclidean Distance* between two goods μ_{ED}. μ_{TRD} tries to balance the loading of each vehicle and μ_{ED} tends to select the nearest unprocessed goods from one of vehicles.

$$(k,n) = \arg_{\max}[pher(v_k, g_n)^{\alpha} \times \mu(k,n)^{\beta}] \qquad (2)$$

$$\mu_{TRD}(k,n) = 1/TotalRoutingDistance_k \qquad (3)$$

$$\mu_{ED}(k,n) = 1/EuclideanDistance(v_k, g_n) \quad (4)$$

2) Exploration

It is easy to fall into the local optimal problem if only *Exploitation* is used. The purpose of *Exploration* is to make *ACO* has a chance to discover various potential routes even the pheromone of a route is not very large. However, the solutions may be not good if the random selection strategy is applied. Hence, we consider that the selection probability is still based on the pheromone and the distance-based factor. When a route between two locations has much more pheromone and much better distance-based value, the route has higher probability to be selected as the part of solution. As (5) shows, the selection probability can be calculated by the normalization of each pair of the vehicle v_k and good g_n.

$$Prob(k,n) = \frac{([pher(v_k, g_n)]^{\alpha} \times \mu(k,n)^{\beta})}{\sum_{v_{k'} \in V, g_{n'} \in G}([pher(v_{k'}, g_{n'})]^{\alpha} \times \mu(k',n')^{\beta})} \quad (5)$$

D. NEH

Due to *ACO* is based on the pheromone distribution and distance-based factor, the solutions may become worse if the pheromone accumulates on the incorrect routes. To guarantee the generated solutions are not too bad in each iteration, we adopt *NEH* algorithm [3] which is a well-known heuristic algorithm on sorting problems to refine the generated routes. *NEH* can efficiently resort the solution permutation to obtain better solutions and further improve the pheromone accumulation in the logistic graph. *NEH* includes three steps:

Step 1. For a logistic route, check the order of the first two goods for minimizing the routing cost without violating the vehicle capacity constraint.

Step 2. Insert the next goods into the new route and make the routing cost as short as possible.

Step 3. Repeat the step 2 until all goods are resorted completely.

We use φ_1 = <w, g_6, g_1, g_2, g_4, w> in Figure 3 as an example to explain the procedure of *NEH*. In Step 1, *NEH* generates two kinds of goods permutations for the first two goods g_6 and g_1, i.e. φ_{a1} = <w, g_6, g_1, w> and φ_{a2} = <w, g_1, g_6, w>. Due to both of φ_{a1} and φ_{a2} meet the vehicle capacity constraint and the routing costs of them are the same, we arbitrarily select φ_{a1} as the new order for the first two goods. In Step 2, there are three positions to insert the third goods g_2 into φ_{a1}, i.e., φ_{b1} = <w, g_2, g_6, g_1, w>, φ_{b2} = <w, g_6, g_2, g_1, w> and φ_{b3} = <w, g_6, g_1, g_2, w>. We select φ_{b3} as the new order since they are valid and the routing cost of φ_{b3} is minimal, i.e., $\delta(\varphi_{b1}) = 211$, $\delta(\varphi_{b2}) = 223$ and $\delta(\varphi_{b3}) = 188$. In the last step, *NEH* repeatedly inserts the next goods g_4 into φ_{b3}. For φ_{c1} = <w, g_4, g_6, g_1, g_2, w> and φ_{c2} = <w, g_6, g_4, g_1, g_2, w>, the vehicle has no enough space to pick up g_4 because g_1 and g_2 have already been loaded in the vehicle (the remaining space is only 4 which is smaller than the volume of g_4). We select φ_{c3} = <w, g_6, g_1, g_4, g_2, w> as the new order because $\delta(\varphi_{c3}) = 188$ is smaller than $\delta(\varphi_{c4}) = 302$, where φ_{c4} = <w, g_6, g_1, g_2, g_4, w>. *NEH* stops since no other goods in φ_1 need to be processed. The original route φ_1 is replaced by φ_{c3}. We observe that the routing cost decreases significantly from $\delta(\varphi_1) = 302$ to $\delta(\varphi_{c3}) = 188$ after applying *NEH* algorithm.

E. Pheromone Update

Recall to the system framework in Figure 2, the initial solution and the best solution in each iteration would update the pheromone of logistic graph. In this section, we introduce how to update the pheromone of logistic graph. Initially, all pheromone values between every pair of goods are equally set as 1.0. After generating some better solutions including the initial solution, the pheromone on each route of the solution would increases. We think that the better a solution is, the much pheromone each route increases. Besides, the pheromone on the route where no ant travels should be decreased because it may not be a good route. Hence, a control parameter named evaporation rate ρ ($0 < \rho < 1$) is designed to decay the pheromone. As (6) shows, the updated pheromone between the goods g_i and g_j can be calculated by the sum of remaining pheromone $(1 - \rho) \times pher(i, j)$ and additional increasing pheromone $\Delta pher_k(i, j)$. The remaining pheromone of each route would be evaporated by multiplying $(1 - \rho)$, where the default value of ρ is set as 0.1. $\Delta pher_k(i, j)$ is measured by the reciprocal of total routing distance of the vehicle k if the vehicle is through the route between gi and g_j. Otherwise, $\Delta pher_k(i, j)$ equals to 0.

$$pher(i,j) = (1-\rho) \times pher(i,j) + \sum_{\forall 1 \le k \le m} \Delta pher_k(i,j) \quad (6)$$

V. Experimental Evaluations

In this section, we evaluate the routing cost of *ACO*-based solution including ACO_{TRD} [3] and ACO_{ED} by a series of experiments based on a semi-real logistic dataset and various system condition settings. All the experiments are implemented in Java JDK 1.8 on an Intel i5 CPU 3.3GHz machine with 8GB of memory running Microsoft Windows 7.

A. Experiment Settings

A real logistic dataset from *KERRY TJ Logistics* [5] in *Miaoli*, *Taiwan* is collected. The dataset contains 1,192 logistic records including a record id, a customer name and a delivery address. Each address is transferred to the longitude and latitude by Google Map API. Unfortunately, this dataset does not contain the goods volume and the vehicle capacity, and only has goods delivery records. Therefore, we develop a logistic simulator based on the real dataset to generate a semi-real dataset. n_G and r_D are used to control the number of pick-up and delivery goods. We randomly select $n_G \times r_D$ delivery records (default as $150 \times 70\% = 105$) from the *KERRY TJ* dataset and generate $n_G \times (1 - r_D)$ pick-up records by the data simulator. The customer's location of a pick-up record is randomly generated within *Miaoli*. The volume of each goods is determined by normal distributions with means equal to v_{avg} (default as 5). The default vehicle capacity *cap* is set as 250. For *ACO*, the default number of ants in each iteration is set as 20, and α and β are set as 1.0. Due to the main goal of this series of experiments is to measure the quality of logistic route planning, we use total routing distance to measure the routing cost.

B. Impact of Various Distance-based Factors β

The parameter β is used to control the impact of distance-based factor. This experiment analyzes the routing cost of ACO_{TRD} and ACO_{ED} under various β settings. As Figure 4 shows, we observe that ACO_{ED} is

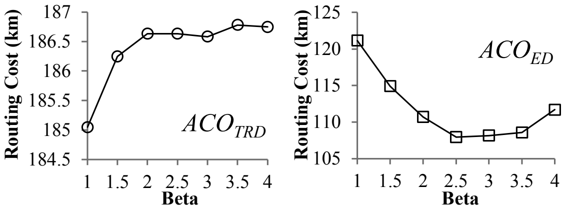


Figure 4. Impact of various distance-based factors β.

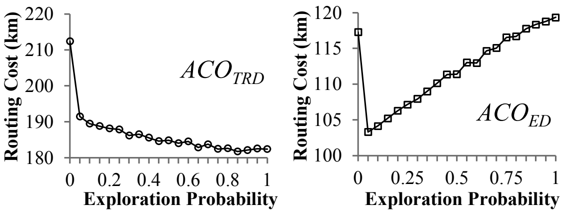


Figure 5. Impact of various exploration probabilities.

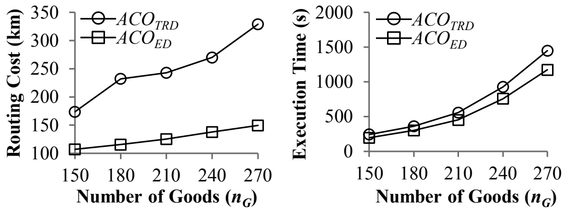


Figure 6. Impact of various number of goods.

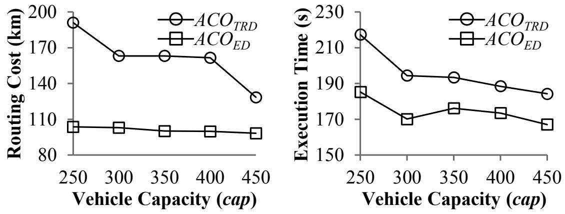


Figure 7. Impact of various vehicle capacities.

better than ACO_{TRD} in terms of routing cost. The routing cost increases with β increases for ACO_{TRD}. The result shows that excessive emphasis on total routing distance does not help for the quality of route planning. For ACO_{ED}, we observe that the routing cost is not good when β is too small or too large. The best solution is obtained when β is set as 2.5. Hence, the default value of β is set as 2.5 for the following experiments.

C. Impact of Various Exploration Probabilities

The purpose of exploration is to make *ACO* has a chance to discover various potential routes even the pheromone of a route is not very large. This experiment analyzes the routing cost of ACO_{TRD} and ACO_{ED} under various exploration probabilities. In Figure 5, the experimental results show that the best exploration probability setting is around 0.1. A too large exploration probability may lead to the exploitation behavior of ant decreases and good solutions may not be extended to the following iterations. Besides, the execution time would increase when the exploration probability increases because exploration needs to spend much time to calculate the selection probability for every unprocessed goods. Hence, the default probability of exploration is set as 0.1.

D. Impact of Various Numbers of Goods n_G

This experiment compares the routing cost and execution time of ACO_{TRD} and ACO_{ED} when n_G is varied from 150 to 270. As Figure 6 shows, we observe that the routing cost and execution time increase with n_G increases. The reason is that both of the route length and the number of possible routes increase when n_G increases. *ACO*-based solution needs to spend much time to search the logistic routes. Although the execution time of ACO_{TRD} and ACO_{ED} are similar, ACO_{ED} significantly outperforms ACO_{TRD} in terms of routing cost especially under the large number of goods. The average improvement rate of routing cost is 49%.

E. Impact of Various Vehicle Capacity cap

This experiment compares the routing cost and execution time of ACO_{TRD} and ACO_{ED} when *cap* is varied from 250 to 450. Figure 7 shows that the routing cost decreases when the vehicle capacity increases. The reason is that a larger vehicle can load much more goods in a route, and thus the number of planned routes decreases. Besides, the execution time also decreases because the number of *ACO* iterative computations can be reduced. Overall, ACO_{ED} is better than ACO_{TRD} in terms of routing cost and the average improvement rate is 37.5%.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we have proposed a solution based on *Ant Colony Optimization* (*ACO*) for finding high quality logistic routes with pick-up and delivery by considering real logistic constraints. In the *ACO*-based solution, we have first adopted *Greedy Randomizes Adaptive Search Procedure* (*GRASP*) to generate initial solution and pheromone distribution of a logistic graph. Then, we have designed the *Exploitation* and *Exploration* operations to search better solutions by a set of ants and assign goods to each vehicle. We have also adopted *NEH* algorithm to refine the generated routes. Finally, the solutions in each iteration would update the pheromone of logistic graph and minimize the routing cost as much as possible. To our best knowledge, this is the first work that facilitates logistic route planning based on *ACO* with simultaneous consideration of goods pick-up and delivery requirements.

To evaluate the performance of the proposed *ACO*-based solution, we have conducted a series of experiments based on a semi-real logistic dataset from *KERRY TJ Logistics*. The experimental results show that the proposed *ACO*-based solution performs well in terms of route quality and planning efficiency. For the future work, we plan to not only consider other real logistic constraints such as transportation time and traffic conditions but also design more sophisticated solutions to further improve the quality of logistic route.

ACKNOWLEDGMENT

This research was supported by Ministry of Science and Technology, Taiwan, R.O.C. under grant no. MOST 104-2221-E-006-205 -; and Ministry of Education, Taiwan, R.O.C. The Aim for the Top University Project to the National Cheng Kung University (NCKU).

REFERENCES

[1] J. Belloso, A. A. Juan, J. Faulin and A. Serrano, "Using Multi-Start Biased Randomization of Heuristics to Solve The Vehicle Routing Problem with Clustered Backhauls," *Lecture Notes in Management Science*, vol. 7, pp. 15-20, 2015.

[2] A. Bortfeldta, T. Hahnb, D. Mannela and L. Monch, "Hybrid Algorithms for The Vehicle Routing Problem with Clustered Backhauls and 3D Loading Constraints," *European Journal of Operational Research*, vol. 243, no. 1, pp. 82-96, 2015.

[3] R.-M. Chen, F.-R. Hsieh and D.-S. Wu, "Heuristics based Ant Colony Optimization for Vehicle Routing Problem," in *ICIEA*, pp. 1039-1043, 2012.

[4] M. Goetschalckx and C. J. Blecha, "The Vehicle Routing Problem with Backhauls," *European Journal of Operational Research*, vol. 42, no. 1, pp. 39-51, 1989.

[5] KERRY TJ Logistics, http://www.kerrytj.com/en/default.aspx.

[6] M. Lai, M. Battarra, M. D. Francesco and P. Zuddas, "An Adaptive Guidance Meta-Heuristic for The Vehicle Routing Problem with Splits and Clustered Backhauls," *Journal of the Operational Research Society*, vol. 66, pp. 1222-1235, 2015.

[7] A. Lim and F. Wang, "Multi-Depot Vehicle Routing Problem A One-Stage Approach," *IEEE Transactions on Automation Science and Engineering*, vol. 2, no. 4, pp.397-402, 2005.

[8] R. Li-xia, "Algorithm Study of Multiple-Depot Vehicle Routing Problem based on Fuzzy Simulation," in *ICIMA*, pp.184-187, 2009.

[9] J. Lysgaard, "A New Sparse Transformation and Exact Algorithm for The Clustered Traveling Salesman Problem," in *VEROLOG*, 2015.

[10] K. Menger, "Das Botenproblem," *Ergebnisse Eines Mathematischen Kolloquiums*, 2, pp. 11-12, 1932.

[11] I. H. Osman, "Metastrategy Simulated Annealing and Tabu Search Algorithms for the Vehicle Routing Problem," *Journal Annals of Operations Research*, vol. 41, no. 1-4, 1993.

[12] B. Tunjongsirigul and P. Pongchairerks, "A Genetic Algorithm for a Vehicle Routing Problem on a Real Application of Bakery Delivery," in *ICECT*, pp.214-217, 2010.

[13] L.-N. Xing, Y.-W. Chen, K.-W. Yang, F. Hou, X.-S. Shen and H.-P. Cai, "A Hybrid Approach Combining An Improved Genetic Algorithm and Optimization Strategies for The Asymmetric Traveling Salesman Problem," *Engineering Applications of Artificial Intelligence*, vol. 21, no. 8, pp. 1370-1380, 2008.