

Available online at <http://www.mecs-press.net/ijmsc>

A Novel Mathematical Model for Cross Dock Open-Close Vehicle Routing Problem with Splitting

Mahdi Alinaghian ^a, Mina Rezaei Kalantari ^a, Ali Bozorgi-Amiri ^b, Nima Golghamat Raad ^a

^a *Department of Industrial and Systems Engineering, Isfahan University of Technology, Isfahan, Iran.*

^b *School of Industrial Engineering, College of Engineering, University of Tehran, Iran*

Abstract

Cross docks play an important role in goods distribution. In most of the common models, the capacity of vehicles is not completely used as they assume that each node is met only by one vehicle. Also, due to high cost of purchasing vehicles with high capacity, rental vehicles are used in collecting section. In this paper, a novel mathematical model is presented in which, each node can be possibly visited by different vehicles (splitting). Besides, in the proposed model, existence of open routes in pickup section has been supposed. Then, one meta-heuristic method based on the simulation annealing algorithm with two different approaches has been developed. For testing the performance of the proposed algorithm, the obtained results compared with the exact answers in both small and large scales. The outcomes show that the algorithm works properly.

Index Terms: Cross dock, open-close vehicle routing problem (OCVRP), vehicle routing problem with splitting delivery (VRPSD), simulated annealing algorithm (SAA).

© 2016 Published by MECS Publisher. Selection and/or peer review under responsibility of the Research Association of Modern Education and Computer Science

1. Introduction

Today, using cross docks is a novel logistic strategy for logistics companies [1]. Cross docks are used when the supply chain consists of at least two levels (for example, customer and supplier) [2]. Cross docking is a continuous process to the final destination without storing the products and materials in distribution center for a long time. Unlike the traditional methods of storage of goods, in this method, all goods are maintained only for a short period of time (maximum 24hrs). This strategy proposed by Walmart for the first time [3]. When the order is issued by the customer, the goods get loaded in vehicles and become shipped. These warehouses get used just when the demand is fixed and can be fulfilled swiftly. Using this strategy the inventory costs and the customer waiting time decreases [4]. In cross docks, the incoming goods become unloaded from vehicles and

* Corresponding author. Tel.:

E-mail address: alinaghian@cc.iut.ac.ir

get stored then they are transferred to be loaded in delivery section [4]. At least four objective functions have been considered for this problem consist of minimizing the number of vehicles, minimizing the total cost of transportation, minimizing the total end time and total covered distance [5]. Hansani-Goodarzi, Tavakkoli-moghadam in 2012 considered a multi-product SVRR with limited capacity of the vehicles. They suggested a MILP model for this problem and solved it with GAMS software. In their paper, all tours are considered as closed [5]. Moghadam et al. attended the routing and scheduling problem in a cross dock and considered a special time window for each customer and salesperson. The demand of customer or salesperson must be fulfilled in this time window [3]. Two of the first researchers discussed the cross dock are Sung and Song [6]. They developed an integrated network for locating the cross docks and allocating the trucks throughout the chain. Mousavi and Tavakkoli Moghaddam [4] worked on the layout and routing of the vehicles in a cross dock and proposed a mixed nonlinear integer programming model for this problem. It is mentioned in their paper that this type of problems are considered as NP-hard problems. Finally, the problem was solved with two Meta heuristic algorithms. Zhi-hua hou et al did some studies about cross docks regarding the vehicle routing in order to decrease its costs. They divided the problem into two sub-problems of pickup process and delivery process and also presented two separated objective functions [7]. Mousavi et al proposed cross dock Location Routing Problem (LRP) regarding the uncertainty of some input parameters [8]. Dondo and Cerda [9] suggested MILP model for routing the vehicles regarding the cross docks. This model can do the cross dock allocation the truck scheduling in a time period when the cost and capacity of the vehicles are different. They used Sweep heuristic-based models and solved the problem in large scales. Oh and et al. [10] used the platform allocation problem for a post center in South Korea and used a 3-phased heuristic algorithm and one genetic algorithm for solving it. Miao et al. [11] used Truck dock assignment in a cross dock regarding the operational costs inside the cross dock and solved the problem with two Meta heuristic algorithms, tabu search algorithm and genetic algorithm. Also Miao and et al. [12] devised a linear model for Truck dock assignment and solved the problem with a tabu search algorithm. Kuo [13] considered the vehicle scheduling and truck dock assignment, and solved it with a variable neighborhood searching algorithm and used simulated annealing algorithm to compare the obtained results. Konur & Golias [14] studied vehicle scheduling problem in a cross dock. In their paper, the arriving times of the vehicles to the dock are not determined. Madani and et al. [15] considered a problem with multiple cross docks and proposed a mixed nonlinear integer programming model for it. The objective was to minimize the total operational time and to maximize the outcome of the cross dock. For solving this problem, two firefly Meta heuristic algorithms and one simulated annealing algorithm are used. Another group of researchers have worked on vehicle routing in cross docks. Lee and et al. emphasized the importance of cross docks in decreasing the inventory costs and attended the vehicle routing problem. They explained that unlike most of the previous researches which were focused on strategic aspect of the cross docks, they were concentrated on the operational aspect of cost reduction in the cross docks [16]. Liao et al. [17] worked on the vehicle routing in cross docks. Their objective was to minimize the operational costs, transportation costs and to propose an optimal sequence of vehicles and to calculate the optimum number of vehicles. Morais et al. [18] considered a routing problem in cross docks and solved it. Their objective was to find the best routs for the vehicles. Agustina et al. [19] worked on the scheduling and routing of a cross dock for food and expressed that using the cross docks is useful for the goods which have short holdover time. They minimized the costs containing the inventory cost, delivery cost and earliness or tardiness cost. The open vehicle routing problem is a kind of vehicle routing problem. In this problem a rout starts from a customer and finally gets back to the depot or starts from the depot and returns back to a customer without going back to the depot. The open vehicle routing problem occurs in firms which do not have their own vehicles and ask other companies to provide the vehicles [20]. In most of the usual models, because of this assumption that every node can be visited only by one vehicle, a part of vehicle's capacity remains unused. Also, because of the need to high-capacity vehicles in pickup section, the vehicles of this part are usually rental. So assuming the open routs in pickup section is practical. In this paper, a novel mathematical model is presented in which, different vehicles can visit one node. Besides, in the proposed model, existence of open –close Tours is assumed. This means that open Tours are used in collecting section. In continue, a mathematical model is presented in section

two, third section is assigned to meta-heuristic algorithm presentation. In fourth section results are presented and finally conclusion is presented in the section five.

2. Mathematical Model

In companies which use cross-docks for their production section, the vehicles of the pickup section are typically larger than the vehicles of the delivery section and also are more costly. So, in these companies, rental vehicles are more used. Accordingly, the routs are open in this section. Also, in some distribution policies, it is possible to split the demands of customers and suppliers. In this paper, a mathematical model for a cross-dock with open-close routs and possibility of splitting the demands is proposed.

All assumptions are as follows:

- All of supplies produce one type of goods (all kinds of goods can be considered as one type of goods) and scheduling is assigned in one period.
- A supply chain consist of two levels (customers and suppliers) with one cross dock is considered.
- In each node, splitting service is allowable.
- Shortage, throughout the chain is not allowable, and total demands of the customers is equal to total deliveries of the suppliers also no inventory is stored in cress dock.
- Each vehicle has a determined capacity and amount of loaded goods must be equal to or less than its capacity.
- Vehicles in suppliers section are rental and won't start their trips from the cross dock. Also using a vehicle has a fix cost and variable cost.
- Number of vehicles in customer section is predetermined and homogenous.

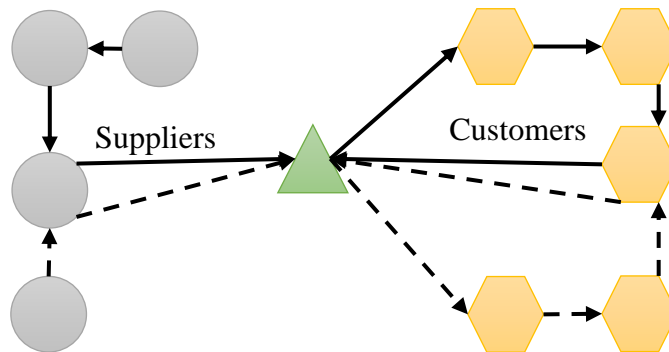


Fig.1. An Example of Proposed Problem

As can be seen in the fig.1, in suppliers and customers sections splitting is allowable, also in the suppliers section open tours are acceptable.

2.1. Sets, variables and parameters definition

Sets, variables and parameters are as follows:

K: Set of all vehicles and k is its index.

Y: Set of the pickup section vehicles.

U: Set of the vehicles of delivery section.

N: Set of the Nodes, and i, j and h are its indexes.

D: Set of delivery nodes.

P: Set of pickup nodes.

c_{ij} : Cost of traveling edge i-j.

C: Renting cost of delivery section vehicles.

N+1: Virtual node.

q_p : Production capacity of node p.

Q_d : Demand of node d.

Q: Capacity of pickup section vehicles.

L: Capacity of delivery vehicles.

re^k : Binary variable which is equal to 1 if the vehicle k is being used.

w_i^k : Binary variable which is equal to 1 if vehicle k traverse from node i to node j.

x_{ij}^k : Binary variable which is equal to 1 if vehicle k visits node i.

Z_d^k : Non-negative variable which shows the amount of goods deliver to the node d by vehicle k.

a_p^k : Non-negative variable which shows the amount of goods pickup from node p with vehicle k.

u_i^k : Non-negative variable which is used in sub tour elimination constraints.

2.2. Mathematical model

The mathematical model of the problem is as follows:

$$\text{Min}(\sum_{i \in N} \sum_{j \in N} c_{ij} \sum_{k \in K} x_{ij}^k + C \sum_{k \in Y} re^k) \quad (1)$$

$$\sum_{j \in P} x_{jl}^k = re^k \quad \forall k \in Y \quad (2)$$

$$\sum_{j \in P} x_{(n+1)j}^k = re^k \quad \forall k \in Y \quad (3)$$

$$\sum_{k \in K} \sum_{j \in N} x_{j(n+1)}^k = 0 \quad (4)$$

$$\sum_{i \in N} \sum_{j \in N} x_{ij}^k \leq M \times re^k \quad \forall k \in K \quad (5)$$

$$\sum_{j \in N} x_{1j}^k = 1 \quad \forall k \in U \quad (6)$$

$$\sum_{i \in N} x_{ih}^k = \sum_{j \in N} x_{hj}^k \quad \forall k \in K, \forall h \in N / \{1, n+1\} \quad (7)$$

$$\sum_{k \in K} \sum_{i \in N} x_{ij}^k \geq 1 \quad \forall j \in N / \{1, n+1\} \quad (8)$$

$$\sum_{k \in K} u_i^k = 0 \quad (9)$$

$$u_i^k + 1 \leq u_j^k + M \times (1 - x_{ij}^k) \quad \forall k \in K, \forall j \in N / \{1\}, \forall i \in N / \{1\} \quad (10)$$

$$\sum_{p \in P} \sum_{k \in K} x_{dp}^k = 0 \quad \forall d \in D \quad (11)$$

$$\sum_{d \in D} \sum_{k \in K} x_{pd}^k = 0 \quad \forall p \in P \quad (12)$$

$$\sum_{k \in U} z_d^k = q_d \quad \forall d \in D \quad (13)$$

$$\sum_{k \in Y} a_p^k \leq q_p \quad \forall p \in P \quad (14)$$

$$\sum_{d \in D} z_d^k \leq L \quad \forall k \in U \quad (15)$$

$$\sum_{p \in P} a_p^k \leq Q \quad \forall k \in Y \quad (16)$$

$$z_d^k \leq q_d \times w_d^k \quad \forall k \in U, \forall d \in D \quad (17)$$

$$a_p^y \leq q_p \times w_p^k \quad \forall k \in Y, \forall p \in P \quad (18)$$

$$\sum_i x_{ij}^k = w_j^k \quad \forall k \in K, \forall j \in N \quad (19)$$

$$w_j^k, x_{ij}^k, re^k \in \{0, 1\} \quad z_d^u, a_p^y, u_i^k \geq 0 \quad \forall k \in K, \forall j \in N, \forall i \in N, \forall y \in Y, \forall p \in P, \forall u \in U, \forall d \in D \quad (20)$$

To model the open trips in pickup section, a dummy node is defined also it is assumed that all vehicles in pickup section must start their trips from dummy node. Distance between dummy node and other nodes considered to be zero. In the proposed model Eq.1 shows the objective function of the problem which consists of two terms .The first term states the total covered distance by vehicles and the second term is the total rental cost of vehicles in pickup section. Eq.2 and Eq.3 mean that, if any vehicle wants to be used in pickup section should start it trip from dummy node and finish to the cross dock. Eq.4 insure that it is impossible to return to the dummy node.Eq.5 states that only vehicles with $re^k = 1$ can be used for serving the nodes.

Eq.6 insure that all vehicles in delivery section should start and finish their tours form cross dock to cross dock. Eq.7 insure the flow conservation of trips. Eq.8 states that all nodes except number 1 and the dummy node, can be visited by more than one vehicle. Eq.9 and Eq.10 are for eliminating the sub tours. Eq.11 and Eq.12 guarantee that vehicles of pickup section cannot serve the delivery nodes and vice versa. Eq.13 and Eq.14 insure that amount of goods pick up or deliver to a node by a specific vehicle should not exceed form demand of delivery nodes or capacity of pickup nodes. Eq.15 and Eq.16 are the constraints of vehicles capacity. Eq.17, Eq.18 and Eq.19 insure that if $w_j^k = 1$ vehicle k can carry the load of node j. and finally, Eq.20 is for variables definition.

3. Solution Method

Considering the NP-hardness of proposed problem, to solve large size instances a metaheuristic algorithm based on simulated annealing algorithm is proposed. In this section proposed metaheuristic algorithm is presented.

Meta-heuristic algorithms are known tools for optimization [21]. Simulated annealing algorithm is a simple and effective meta-heuristic algorithm for optimization. The origin of simulated annealing refers to the works of Kirkpatrick in 1983 [22]. Kirkpatrick was specialist of statistical physics. He suggested a method based on

slow annealing technic for solving difficult optimization problems. Slow annealing technique is used by metallurgists for reaching a state in which, the solid substance is well formed and its energy is at the minimum level. This technique consists of heating up the substance to a very high temperature and then cooling the substance gradually. Fig.2 illustrate the simulated annealing algorithm. In simulated annealing algorithm, each point of the space is corresponding to a state of a physical system, and the function which has to get minimized is corresponding to internal energy of the system in that state. In this method, the objective is to transit the system from the initial state to the minimum energy state.

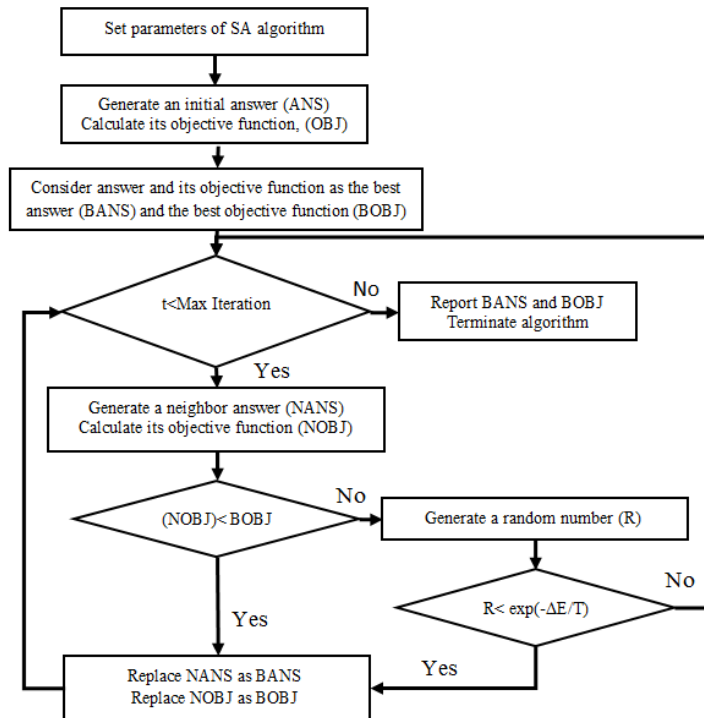


Fig.2. Flow Chart of Simulated Annealing Algorithm

For solving an optimization problem, SA algorithm starts from a feasible answer at first then, it moves toward the neighbor answers. If the neighbor answer is better than the current one, the algorithm replace it as the current answer. (Moves to it), otherwise, the algorithm chooses that answer with the probability of $\exp(-\Delta E/T)$ as the current answer. In this equation, ΔE is the difference between the objective function of the current answer and the neighbor answer.

T is the temperature parameter. In each temperature, many Interactions are done and then the temperature is reduced slightly. In first steps, the temperature is chosen too high, so that the bad answers are more likely to be chosen. By gradual reduction in temperature, in last steps there is little chance for bad answers to be accepted. Hence, the algorithm is convergent to a good answer.

3.1. Answer representation

In this paper, representation of answer is done by two vectors .the first vector consist of N cells and shows the sequence of visiting nodes. Second vector consist of K-1 cells and shows that whether splitting for vehicle k and K+1 is allowable or not. For better illustration an example are shown in fig.3. Suppose that nodes 1, 2

and 3 are suppliers and nodes 4,5 and 6 are customers, also 2 vehicles accessible in pickup section and 1 vehicle is available in delivery section. According to the fig.3 in pick up section sequence of customers are 3,2 and 1. Vehicle 1 starts its trip from node 3, and continue servicing suppliers until its capacity allows then vehicle 2 continue servicing suppliers according to the predetermined sequence of them. Also according to the fig.3 splitting between vehicle 1 and 2 is allowable, so vehicle 1 and 2 can service one customer together.

$$[3,4,5,2,6,1][1,0]$$

Fig.3. Example of Representation for Three Vehicles and Six Customers

In this paper two strategies are used for simulated annealing that are explained as follows.

Strategy 1: in this approach, in each iteration loop a predetermined number of repetitions are assigned and then the temperature gets decreased. The initial temperature is considered equal to the objective function of the first generated answer by the algorithm and in each step reduced 0.2 units. This number is chosen by doing sensitively analysis on the time and objective function.

Strategy 2: in this approach dimension of the problem are considered for declining the temperature. This means that, unlike the first approach, the temperature doesn't go down according to the specific number, but it declines according to the dimensions of the problem. By increase of problem dimension, the algorithm steps increase. In this method, the temperature calculation is as follows:

$$t_f = 0.08 \times t_0 \quad (21)$$

$$m = \frac{P+D-1}{2} \quad (22)$$

$$\beta = \frac{t_0 - t_f}{m \times t_0 \times t_f} \quad (23)$$

In these formula P and D, are suppliers and customers nodes respectively. T_f and T_0 are final temperature and initial temperature respectively. 0.08 is a coefficient obtained by trial and error method and the temperature declines B units in each step.

In this paper, in both strategies, neighborhoods are chosen randomly. Pseudocode of simulated annealing meta-heuristic algorithm as follow:

-
- Generate an Initial random Solution.
 - Calculate the objective function of initial Solution
 - Consider the initial solution as best solution
 - For iteration 1 to total iteration
 - For interior iteration 1 to total interior iteration
 - Generate one new solution
 - Calculate the objective function for the new solution
 - Compare the obtained solution with the best solution
 - If the new answer is better, choose it as the best solution and solution, Otherwise, Generate a Random number and compare it with $\exp(-\Delta E/T)$. If the random number is superior, choose it as best solution.
 - End for interior iteration
 - Decrease temperature
 - Choose the best solution.
-

4. Results

In order to assess the performance of the proposed algorithm, some random problems were created in large scale and small scale, Then the obtained solutions of the algorithm were compared with those obtained with exact method.

The meta-heuristic algorithm is coded by MATLAB software. All of the runs have been done on a Corei5 system, 1.7 GHz and 6GB memory.

4.1. Results in small scales

Seven sample problems are generated in small scales and the obtained results of proposed algorithm are compared with the exact answers. Table 1 shows the results. In this table, the first column is for problem's number. The second column shows the problem's characteristics (number of suppliers- number of pickup vehicles- number of customers- number of delivery vehicles). Two next columns are for results of exact method, and finally 6 next columns are for SA with strategy 1 and SA with strategy 2.

Table 1. Results in Small Size Problems

Num	Problem	Exact method	SA1		SA2				
		answer	Time (s)	answer	Time (s)	error	answer	Time (s)	error
1	2-1-3-2	268	0.02	268	2.04	0	268	9.9	0
2	2-1-4-2	1271	0.2	1271	2.04	0	1271	12	0
3	2-2-5-3	2030	2.5	2030	3.1	0	2030	15	0
4	3-2-6-4	2759	9.6	2894	3.5	4.9	2894	15.4	4.9
5	3-3-8-5	2997	32.5	3289	4.4	9.7	3269.8	18.7	9.1
6	3-3-7-4	2996	80.0	3125	3.6	4.2	3125	17.1	4.2
7	3-3-9-5	3235	12220.2	3363	4.5	3.6	3321	19.4	2.6
Average		2222.2	192.4	2320.0	3.7	3.2	2311.2	15.3	3.0

As it is shown in the table 1, average error of proposed algorithm by the first and second strategies are %3.2 and %3.0 respectively but Strategy 1 of the algorithm has a bit better performance in run time. The Exact method takes 192 seconds in average to solve small problems. But the two strategies of proposed algorithm take around 3.2 and 15.3 seconds in average to do the same. Fig.4 shows the run time of the algorithms. In the first three sample problems, the error of both approaches is zero in comparison with the exact answer. Also, except the sample problem number 6, for rest of the problems the error of the algorithms is less than %5.

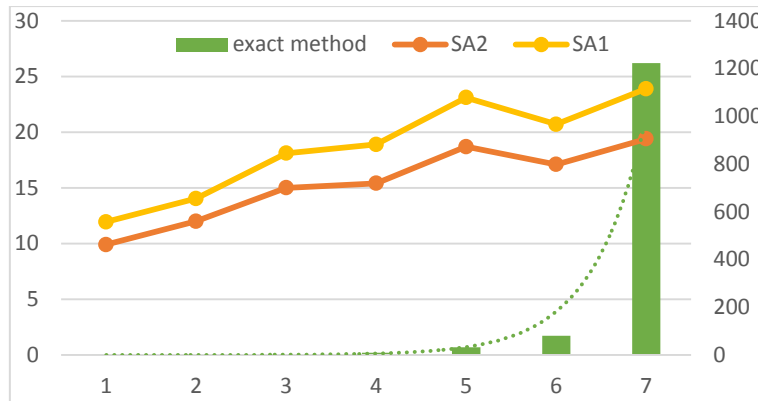


Fig.4. Run time of solution methods in small size problems

In fig.4 the right axis is for runtime of exact method, and the left axis is for run times of proposed algorithm's two strategies. As it can be seen in the fig.4, run time of exact methods increase exponentially by increasing the problem's size. But the run time of the algorithms are polynomials.

4.2. Results in large scales

In large scale test problems 8 sample problems have been solved. In this type of problems, it is impossible to reach to the optimal answer in a reasonable time. So the obtained results from the two approaches are compared with each other. The results are shown in the table 2.

Table 2. Results in Large Sized Problems

Num	Problem	SA1			SA2		
		answer	Time (s)	Error	answer	Time (s)	Error
1	4-4-10-6	3263	4.5	0.01	3221	19.4	0
2	4-4-11-6	4275.4	4.5	0.03	4149.8	23.1	0
3	4-4-12-6	4594.4	4.7	0.01	4562	24.4	0
4	10-5-20-9	4720	5.0	0.04	4553.6	24.0	0
5	12-5-25-9	5863.7	8.2	0.04	5645.1	41.2	0
6	14-6-30-12	7773.2	11.4	0.04	7484.4	54.0	0
7	16-6-35-12	8530	11.5	0.02	8379.4	57.3	0
8	18-6-40-15	10484.6	14.7	0.03	10181	70.5	0
Average		6675.5	8.1	0.03	6522.0	39.2	0

As it can be seen in table 2 the quality of second strategy of the proposed algorithm is better than the first one.

The error of the first approach is %3 in average. Fig.5 shows improvement of the solutions in second approach. In large scaled samples, the second approach gives better answers than the first approach in all cases.

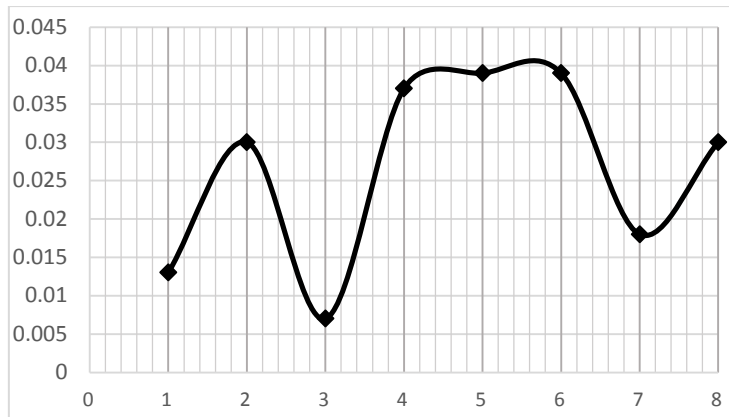


Fig.5. Percentage of improvement obtained by the second approach

The largest improvement in the second strategy is %3.9 which happens in problem number 5 and 6. In the term of run time, the first approach has a small run time in average but it is not its advantage because of its average error. It seems that this strategy rapidly converges to the local optimal solution.

5. Results and Conclusions

In this paper, a new mathematical model is proposed for cross dock problem with open – close tours and splitting. Then, according to the NP-Hardness of proposed problem, a meta-heuristic algorithm based on simulated annealing is suggested with two approaches in order to solve the model. The obtained solutions of small scale problems shows 3.2% and 3% error for two approaches of the suggested algorithm. According to the run time, the exact method takes 192 seconds in average to solve a problem while the two approaches of the algorithm take 3.2 and 15 seconds to do the same. In large scales, for all of the sample problems, the second approach has a better performance than the first one in quality criterion. In these scales, solutions of the second approach are almost %3 better than the first approach. According to the run time in large scale problems, the obtained run times of the first and second approaches were 8.1 and 39.1 seconds respectively. The results confirm a better performance of the second approach in view of quality.

References

- [1] Van Belle, J., Valckenaers, P., & Cattrysse, D. (2012). Cross-docking: State of the art. *Omega*, 40(6), 827-846.
- [2] Santos, F. A., Mateus, G. R., & Da Cunha, A. S. (2013). The pickup and delivery problem with cross-docking. *Computers & Operations Research*, 40(4), 1085-1093.
- [3] Moghadam, S. S., Ghomi, S. F., & Karimi, B. (2014). Vehicle routing scheduling problem with cross docking and split deliveries. *Computers & Chemical Engineering*, 69, 98-107.
- [4] Mousavi, S. M., & Tavakkoli-Moghaddam, R. (2013). A hybrid simulated annealing algorithm for location and routing scheduling problems with cross-docking in the supply chain. *Journal of Manufacturing Systems*, 32(2), 335-347.
- [5] Hasani-Goodarzi, A., & Tavakkoli-Moghaddam, R. (2012). Capacitated vehicle routing problem for multi-product cross-docking with split deliveries and pickups. *Procedia-Social and Behavioral Sciences*, 62, 1360-1365.

- [6] Sung, C. S., & Song, S. H. (2003). Integrated service network design for a cross-docking supply chain network. *Journal of the Operational Research Society*, 54(12), 1283-1295.
- [7] Hu, Z. H., Zhao, Y., & Choi, T. M. (2013). Vehicle routing problem for fashion supply chains with cross-docking. *Mathematical problems in Engineering*, 2013.
- [8] Mousavi, S. M., Vahdani, B., Tavakkoli-Moghaddam, R., & Hashemi, H. (2014). Location of cross-docking centers and vehicle routing scheduling under uncertainty: a fuzzy possibilistic-stochastic programming model. *Applied Mathematical Modelling*, 38(7), 2249-2264.
- [9] Dondo, R., & Cerdá J. (2015). The heterogeneous vehicle routing and truck scheduling problem in a multi-door cross-dock system. *Computers & Chemical Engineering*, 76, 42-62.
- [10] Oh, Y., Hwang, H., Cha, C. N., & Lee, S. (2006). A dock-door assignment problem for the Korean mail distribution center. *Computers & Industrial Engineering*, 51(2), 288-296.
- [11] Miao, Z., Lim, A., & Ma, H. (2009). Truck dock assignment problem with operational time constraint within crossdocks. *European journal of operational research*, 192(1), 105-115.
- [12] Miao, Z., Cai, S., & Xu, D. (2014). Applying an adaptive tabu search algorithm to optimize truck-dock assignment in the crossdock management system. *Expert Systems with Applications*, 41(1), 16-22.
- [13] Kuo, Y. (2013). Optimizing truck sequencing and truck dock assignment in a cross docking system. *Expert Systems with Applications*, 40(14), 5532-5541.
- [14] Konur, D., & Golias, M. M. (2013). Cost-stable truck scheduling at a cross-dock facility with unknown truck arrivals: A meta-heuristic approach. *Transportation Research Part E: Logistics and Transportation Review*, 49(1), 71-91.
- [15] Madani-Isfahani, M., Tavakkoli-Moghaddam, R., & Naderi, B. (2014). Multiple cross-docks scheduling using two meta-heuristic algorithms. *Computers & Industrial Engineering*, 74, 129-138.
- [16] Lee, Y. H., Jung, J. W., & Lee, K. M. (2006). Vehicle routing scheduling for cross-docking in the supply chain. *Computers & Industrial Engineering*, 51(2), 247-256.
- [17] Liao, C. J., Lin, Y., & Shih, S. C. (2010). Vehicle routing with cross-docking in the supply chain. *Expert Systems with Applications*, 37(10), 6868-6873.
- [18] Morais, V. W., Mateus, G. R., & Noronha, T. F. (2014). Iterated local search heuristics for the vehicle routing problem with cross-docking. *Expert Systems with Applications*, 41(16), 7495-7506.
- [19] Agustina, D., Lee, C. K. M., & Piplani, R. (2014). Vehicle scheduling and routing at a cross docking center for food supply chains. *International Journal of Production Economics*, 152, 29-41.
- [20] Vincent, F. Y., Jewpanya, P., & Redi, A. P. (2016). Open Vehicle Routing Problem with Cross-Docking. *Computers & Industrial Engineering*.
- [21] Moshizi, M. M., & Bardsiri, A. K. (2015). The Application of Meta-Heuristic Algorithms in Automatic Software Test Case Generation. *IJMISC-International Journal of Mathematical Sciences and Computing (IJMISC)*, 1(3), 1.
- [22] Kirkpatrick, S. (1984). Optimization by simulated annealing: Quantitative studies. *Journal of statistical physics*, 34(5-6), 975-986.

Authors' Profiles



Mahdi Alinaghian is an Assistant Professor at Isfahan university of technology. He received PhD in Industrial Engineering, from Iran university of technology. His research interests include Discrete Event simulation, uncertain programming, supply chain management, Meta heuristic Algorithms. His published research articles appear in, EXPERT SYST APPL, NSPEC, J INTELL MANUF, IJAMT, and etc.