African Journal of Business Management Vol. 5(22), pp. 9297-9303, 30 September, 2011
Available online at http://www.academicjournals.org/AJBM
DOI: 11.5897/AJBM11.942
ISSN 1993-8233 ©2011 Academic Journals

Full Length Research Paper

Transportation problem of cross-docking network with three-dimensional trucks

Hadi Charkhgard* and Ali A. Yahya Tabar

Industrial Engineering Department, Sharif University of Technology, Tehran, Iran.

Accepted 18 July, 2011

A significant part of price of each product is related to distribution process. Therefore, by controlling this process, a good improvement will be observed in not only net profit of related company but also customer satisfaction. Cross-docking network is considered as a useful method for achieving to these goals. In this paper, transportation problem of cross-docking network is used in order to minimize the total cost for transferring the loads between origins and destinations. The three-dimensional shapes are applied for both trucks and loads. This assumption helps us to find the exact capacity of each truck in basis of each product. Thus, decision makers will find easily by which combinations and amount of products each truck will fill. This problem is formulated using a mixed integer non-linear programming (MINLP) and solved by a heuristic algorithm that simulated annealing (SA) is its core. Several numerical examples are solved, and the results show that this algorithm can find efficient solutions in comparison with exact algorithm.

Key words: Cross-docking, transportation problem, mixed integer non-linear programming (MINLP), Heuristics algorithm, simulated annealing (SA).

INTRODUCTION

Customer satisfaction is an important factor that leads to implementing supply chain management in many companies. 30% of item price is because of distribution process (Apte and Viswanathan, 2000). So, improvement of material flow through efficient control of supply chain can not only decrease heavily total cost of supply chain and the item price but also increase dramatically customer satisfaction. A lot of methods have been investigated by many companies to control the material flow. Among all these strategies, cross-docking is believed as an effective method to reduce inventory and improve customer satisfaction. Cross-docking is an operation strategy at flow consolidation centers, and items are transferred from receiving dock to shipping dock without storing them. In other words, cross-docking is a continuous process to the destinations through the cross-dock without storing materials (that is, products) in a distribution center (Apte and Viswanathan , 2000).

Many studies investigated the cross-docking in supply chain in the literature. These studies can be divided into three different groups include: concept of cross-docking, inside of cross-dock and distribution planning problems. There are lots of researches about concept of cross-docking. Pros of cross-docking were described by Allen (2001) and Luton (2003). Apte and Viswanathan (2002) presented cross-docking as a method in the management of a manufacturing supply chain. Studies about inside of cross-dock are related to shape and layout of cross-dock (Bartholdi and Gue, 2000), tracking items inside the cross-dock and dock-door assignment problem. Dock-door assignment problem come into play to find the suitable way of conveying materials from receiving doors to shipping doors in order to minimize total handling cost (Tsui and Chang, 1992; Aickelin and Adewunmi, 2006; Ley and Elfayoumy, 2007). Distribution planning problems are related to distribution planning in cross-docking networks, scheduling of vehicles in receiving and shipping doors, vehicle routing in cross-docking network and finding location of docks in the network. Donaldson et al. (1999) studied a schedule-driven transportation planning in the cross-docking network. Li et al. (2004) focused on eliminating or minimizing the cost of storage

*Corresponding author. E-mail: h.charkhgard@gmail.com. Tel: +98-9389736796.

Abbreviations: MINLP, Mixed integer non-linear programming; **SA,** simulated annealing; **JIT,** just-in-time; **VRP,** vehicle routing problem; **3PL,** third-party-logistics.

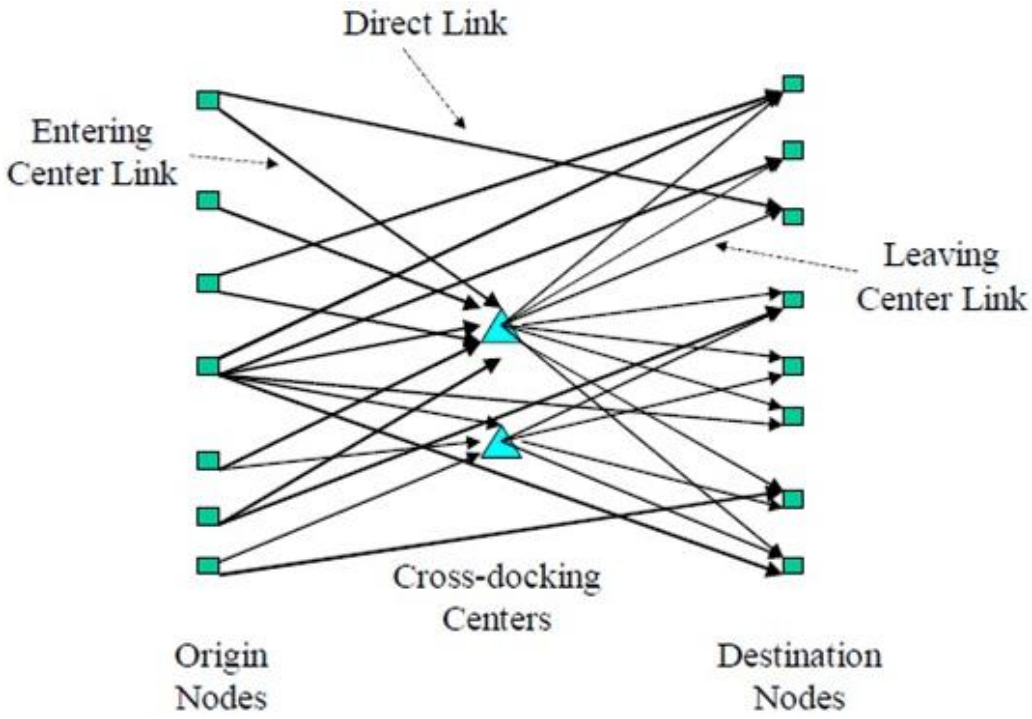

Figure 1. Cross-docking network (Donaldson et al., 1999).

or order picking activities. They used just-in-time (JIT) concepts and formulated the problem as a machine scheduling problem. Chen et al. (2006) consider the network of cross-docks instead of one cross-dock.) Lee et al. (2006) investigated the joint of scheduling and routing problems in the cross-docking network. Liao et al. (2010) considered the joint of cross-docking and vehicle routing problem (VRP) problems at the same time. They are also some papers about various heuristics for optimization such as Pardalos and Resende (2002) and Pardalos and Romeijn (2002).

In this paper, we consider the distribution planning problem of cross-docking network, and the objective is finding to load and route the trucks in the cross-docking network with minimum cost of transportation. This problem was introduced by Donaldson et al. (1999) and then Rami et al. (2010) extended it. They applied pure-integer programming model containing both non-negative integer and binary variables, and their model could find how many trucks is required in each link (origin to destination, origin to cross-dock and cross-dock to destination) and how flow should routed. The binary integer programming model is NP-hard, so they used ant colony optimization based heuristic for solving the model. This paper is adopted from Rami et al. (2010). But the authors assume the three-dimensional shape for trucks and products, so the model is more difficult and realistic. This problem is formulated mixed integer non-linear programming (MINLP), and heuristic algorithm base on simulated annealing (SA) is applied for this model.

Problem description

The cross-docking network includes I suppliers (origins), J customers (destinations) and K cross-dock facilities. The loads can be transferred from each of origins to each of destinations either directly or indirectly through the cross-docking facilities (Figure 1). The loads sent to cross-dock facilities can be consolidated by considering their destination. Thus, the objective is making decision about dispatching each of the loads directly or indirectly, and finding the best consolidation plans in order to minimize the total transportation cost.

Rami et al. (2010) considered several assumptions for their model. In this paper, some of them have been relaxed and the mathematical model is adapted to new condition. We assume as follows:

1. Number of the loads should transfer from origin i to destination j can be more than the capacity of each truck.
2. Trucks are always available when needed. It can be possible by using third-party-logistics (3PL) providers (Rami et al., 2010).
3. Loads and trucks considered as a cube. This assumption is rational because loads' package and trucks are usually like a cube.
4. Trucks are same. Thus, there is no difference between trucks' shape.
5. The loads to be sent from origin i to destination j are same, so their related cubes are completely similar to each other. But the loads to be sent from different origins

to different destinations may be different.
6. The shipping cost is related to both traveled distance and traveled time (Rami et al., 2010).

We now give a MINLP formulation of the model described, using the following notations:

Inputs:

I: the set of origin nodes
J: the set of destination nodes
K: the set of cross-cocking facilities
C_{ij}: the cost of truck from location i to location j
C_{ik}: the cost of truck from location i to cross-dock center k
C_{kj}: the cost of truck from cross-dock center k to location j
X: length of each truck
Y: width of each truck
Z: height of each truck
x_{ij}: length of each of the loads at origin i to destination j
y_{ij}: width of each of the loads at origin i to destination j
z_{ij}: height of each of the loads at origin i to destination j
V_{ij}: the number of the loads that can put to the truck at origin i to destination j, and it can be calculated from equation (1)
$S1_{ij}$: the flow at origin i to destination j
$S2_{ij}$: the complete flow (the flows that can fill the truck) at origin i to destination j, and it can be calculated from equation (2)
N_{ij}: the number of complete trucks that will send from origin i to destination j either directly or indirectly, and it can be calculated from equation (3)
Q_{ij}: the incomplete flow (the flows that cannot fill the truck) at origin i to destination j, and it can be calculated from equation (4)

$$V_{ij} = \left\lfloor \frac{X}{x_{ij}} \right\rfloor . \left\lfloor \frac{Y}{y_{ij}} \right\rfloor . \left\lfloor \frac{Z}{z_{ij}} \right\rfloor \quad \forall i \in \{1, \dots, I\},\ \forall j \in \{1, \dots, J\} \tag{1}$$

$$S2_{ij} = V_{ij} \quad \forall i \in \{1, \dots, I\},\ \forall j \in \{1, \dots, J\} \tag{2}$$

$$N_{ij} = \left\lfloor \frac{S1_{ij}}{V_{ij}} \right\rfloor \quad \forall i \in \{1, \dots, I\},\ \forall j \in \{1, \dots, J\} \tag{3}$$

$$S1_{ij} = N_{ij} . S2_{ij} + Q_{ij} \quad \forall i \in \{1, \dots, I\},\ \forall j \in \{1, \dots, J\} \tag{4}$$

Decision variables:

$Z1_{ij}^{k}$: Binary decision variable, equal to 1 if complete flow from i to j sent through cross-docking facility k, and 0 otherwise
$Z2_{ij}^{k}$: Binary decision variable, equal to 1 if incomplete flow from i to j sent through cross-docking facility k, and 0 otherwise
O_{ik}: non-negative integer variable representing the number of trucks on link ik, from origin i to cross-dock k
D_{kj}: non-negative integer variable representing the number of trucks on link kj, from cross-dock k to destination j
$R1_{ij}$: binary decision variable, equal to 1 if complete trucks sent directly from i to j, and otherwise 0
$R2_{ij}$: binary decision variable, equal to 1 if incomplete trucks sent directly from i to j, and otherwise 0
$T1_{ikm}$: for incomplete flows at node i, at most J trucks will send to cross-dock k. So in this binary variable, m represents the truck number in origin i. This variable is equal to 1 if truck m is applied in link ik, from origin i to cross-dock k
$T2_{kjm}$: for incomplete flows at node k, at most J trucks will send to destination k. So in this binary variable, m represents the truck number in origin i. This variable is equal to 1 if truck m is applied in link kj, from cross-dock k to destination j
$U1_{ijm}$: binary variable, equal to 1 if incomplete flow from i to j is assigned to truck number m in the link of ik, otherwise 0
$U2_{ijm}$: binary variable, equal to 1 if incomplete flow from i to j is assigned to truck number m in the link of kj, otherwise 0

Approximate model:

In this part, an approximate model is shown. This model is so easier than exact model, and its result is compared with exact model in section 4.

$$\text{Min} \quad \sum_i \sum_j N_{ij} R1_{ij} C_{ij} + \sum_i \sum_j R2_{ij} C_{ij} + \sum_i \sum_k O_{ik} C_{ik} + \sum_k \sum_j D_{kj} C_{kj} + \sum_i \sum_j \sum_k Z1_{ij}^{k} N_{ij} (C_{ik} + C_{kj}) \tag{5}$$

$$\text{St: } R1_{ij} + \sum_k Z1_{ij}^{k} = \min\{1, S2_{ij}\} \quad \forall i \in \{1, \dots, I\},\ \forall j \in \{1, \dots, J\} \tag{6}$$

$$R2_{ij} + \sum_k Z2_{2ij}^{k} = \min\{1, Q_{ij}\} \quad \forall i \in \{1, \dots, I\},\ \forall j \in \{1, \dots, J\} \tag{7}$$

$$\sum_j \left(Q_{ij}/V_{ij}\right) Z2_{ij}^k \le O_{ik} \quad \forall i \in \{1,\dots,I\},\ \forall k \in \{1,\dots,K\} \tag{8}$$

$$\sum_i \left(Q_{ij}/V_{ij}\right) Z2_{ij}^k \le D_{kj} \quad \forall j \in \{1,\dots,J\},\ \forall k \in \{1,\dots,K\} \tag{9}$$

$$Z1_{ij}^k, Z2_{ij}^k, R1_{ij}, R2_{ij} \in \{0,1\} \quad \forall i \in \{1,\dots,I\},\ \forall j \in \{1,\dots,J\}, \forall k \in \{1,\dots,K\} \tag{10}$$

$$O_{ik}, D_{kj} \in Z^+ \quad \forall i \in \{1,\dots,I\},\ \forall j \in \{1,\dots,J\}, \forall k \in \{1,\dots,K\} \tag{11}$$

The objective function minimizes the total transportation cost, and it considers both direct and indirect transportation costs. Constraint (6) ensures all complete flows should be satisfied either directly or through cross-docking centers. Moreover, if there is no complete flow between i and j, then there will be no allocation for direct or indirect dispatches. Constraint (7) ensures all incomplete flows should be satisfied either directly or through cross-docking centers. Moreover, if there is no incomplete flow between i and j, then there will be no allocation for direct or indirect dispatches. Constraints (8) and (9) calculate the number of trucks for sending incomplete flows through the cross-dock centers.

Exact model:

Q_{ij}/V_{ij} provides an overview of percentage of the truck that have been filled. By considering each of the constraints (8) and (9), it is clear that, these two constraint cannot calculate the exact number of trucks, and the real s will be greater or equal than the estimations. For example, if we have different flows, and each of them fills 60% of the truck lonely, then the mentioned constraints will calculate the number of required truck 2 (.6+.6+.6 < 2). Thus, it is completely wrong, because the exact number is three. Although approximate model is not a good model but because of simplicity, it can be regarded as a weak lower bound for the results of exact model. By substituting the following constraints instead of constraints (8) and (9), the exact model will make:

$$\sum_j \left(Q_{ij}/V_{ij}\right) Z2_{ij}^k\, U1_{ijm} \le T1_{ikm}$$
$$\forall i \in \{1,\dots,I\}, \quad \forall k \in \{1,\dots,K\}, \quad \forall m \in \{1,\dots,J\} \tag{12}$$

$$\sum_i \left(Q_{ij}/V_{ij}\right) Z2_{ij}^k\, U2_{ijm} \le T2_{kjm}$$
$$\forall i \in \{1,\dots,I\}, \quad \forall k \in \{1,\dots,K\}, \quad \forall m \in \{1,\dots,J\} \tag{13}$$

$$\sum_k Z2_{ij}^k = \sum_m U1_{ijm} \quad \forall i \in \{1,\dots,I\},\ \forall j \in \{1,\dots,J\} \tag{14}$$

$$\sum_k Z2_{ij}^k = \sum_m U2_{ijm} \quad \forall i \in \{1,\dots,I\},\ \forall j \in \{1,\dots,J\} \tag{15}$$

$$\sum_m T1_{ikm} = O_{ik} \quad \forall i \in \{1,\dots,I\},\ \forall k \in \{1,\dots,K\} \tag{16}$$

$$\sum_m T2_{kjm} = D_{kj} \quad \forall j \in \{1,\dots,J\},\ \forall k \in \{1,\dots,K\} \tag{17}$$

Constraints (12) and (13) determine which of J available trucks in each link (ik and kj) should be used. Constraints (14) and (15) ensure each of the incomplete flow for the destination j in the link of ik or kj does not assign to more than one truck. Constraints (16) and (17) count the number of needful truck in each link (ik and kj).

Heuristic algorithm for 3-dimensional cross-docking network

The exact model is so difficult, and software such as LINGO cannot find even a feasible solution for small cases in 24 h runtime. So, a heuristic algorithm role is so important in this problem to find good results in efficient time. Before proceeding to the heuristic algorithm, the SA is investigated as core of this algorithm.

Simulated annealing

This algorithm was proposed by Kirkpatrick et al. (1983) as an extension of the Metropolis-Hastings algorithm (Metropolis et al., 1953). Its name refers to physical process of annealing in metallurgy. Achieving a minimum energy crystalline structure is a reason that SA technique is extended in metallurgy, and it requires heating and slow cooling of materials. The SA algorithm follows this process with the aim of finding a good solution while providing the opportunity to escape from local optima. The opportunities to jump from local optima are depend on the temperature. The more the temperature, the more the opportunity. As the process 'cools' the focus is on

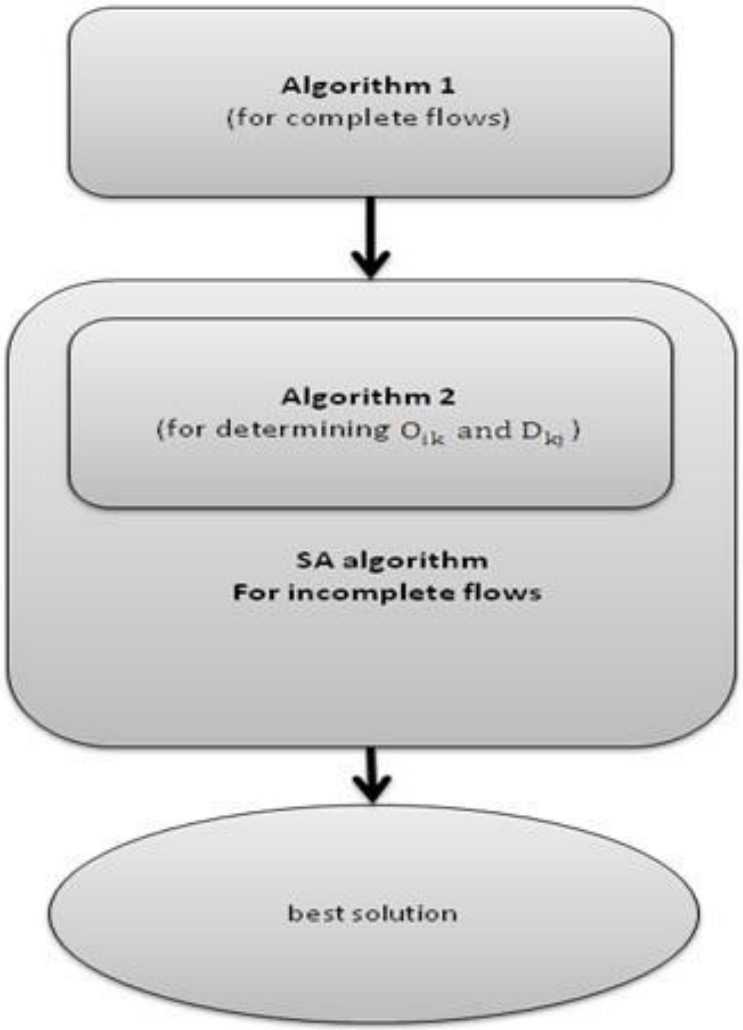

Figure 2. Heuristic algorithm.

finding an optimal solution, so the probability of a jump to a new neighborhood is reduced.

SA has great potential for problems with mixed discrete and continuous variables such as our problem (Bennage and Dhingra, 1995). That is why we consider SA as a core of our algorithm. The process starts with the highest temperature (T_0), which reduces after each repeat. After an initial solution is generated, a random search is conducted to move from the current solution to a neighborhood solution. The neighborhood range selection is on the shoulder of users, and it is very important for the success of SA algorithm. A new solution with a better objective value will always be accepted. But for the solution with worse objective value, there is also an opportunity to accept based on a probability p which is given by $p = e^{-\frac{\Delta}{T}}$, where Δ is the difference between the new solution and the current solution, and T is the current temperature.

Heuristic algorithm

This heuristic algorithm consists of different algorithms (Figure 2). Algorithm 1 is designed for planning complete flows. For incomplete flows, SA algorithm is proposed but it cannot plan the incomplete flows lonely. Due to the fact that the calculation of O_{ik} and D_{kj} is very difficult, an algorithm (heuristic or metaheuristic) is needed to solve these variables, so algorithm 2 is used for this purpose. Finally, interaction of algorithm 2 and SA in each repeat of SA can leads to find the best solution.

Algorithm 1

Step1: calculate all cases (K+1) of dispatching costs both directly and indirectly for complete flows. For example, for sending complete flow of destination j from origin i, calculate: (C_{ij} and $C_{ik}+C_{kj}$ $\forall k \in \{1, \dots, K\}$).

Table 1. Results.

Problem			Exact model (LINGO results)		approximate model (LINGO results)	heuristic algorithm
I	J	K	objective function value	time (s)	objective function value	objective function value
5	5	3	166	3600	165	166
10	10	7	-	86400	775	806
12	12	8	-	86400	1030	1260
15	15	10	-	86400	1574	2137
20	20	12	-	86400	2600	3537
25	25	14	-	86400	4061	5662

Step 2: compare all calculated costs of step 1, and choose the minimum of them as the best way of transferring related completed flow.
Step 3: repeat steps 1 and 2 for different i and j.
Step 4: compute

$$\sum_i \sum_j N_{ij} R1_{ij} C_{ij} + \sum_i \sum_j \sum_k Z1_{ij}^k N_{ij} (C_{ik} + C_{kj})$$

Simulated annealing (SA) algorithm parameters

The following parameters are applied for SA algorithm:

T_0 (Initial temperature) = the objective function value of initial solution;
The number of remaining iterations $= I \times K \times J$;

$$T_{w+1} = {}^{T_w}/_{(1+\alpha)} \qquad \text{w: iteration number} \tag{18}$$

$$\alpha = {}^{1}/_{K} \tag{19}$$

Runtime limitation= 1 hour

Algorithm 2

This algorithm is divided into 2 parts. In the first part O_{ik} is calculated and in the second part D_{kj} is computed.

First part:

Step 1: O_{ik}=0 and truck capacity is 1.
Step 2: compute $Z2_{ij}^k({}^{Q_{ij}}/_{V_{ij}})\ \forall j \in \{1, \dots, J\}$ for origin i and cross-dock center k, then sort them and create a list and put them in.
Step 3: choose the minimum positive number from step 2 and reduce it from truck capacity. After that, omit it from the list.
Step 4: choose the maximum positive number from step 2 that is not greater that truck capacity, and then reduce it from truck capacity. After that, omit it from the list. If there is no suitable number go to step 7.
Step 5: choose the minimum positive number from step 2 that is not greater that truck capacity and then reduce it from truck capacity. After that, omit it from the list. If there is no suitable number go to step 7.
Step 6: repeat step 5.
Step 7: if the truck capacity is less than 1, then $O_{ik}=O_{ik}+1$.
Step 8: repeat steps 2 to 7 for J times.
Step 9: repeat the algorithm for different i and j.

Second part:

Step 1: D_{kj}=0 and truck capacity is 1.
Step 2: compute $Z2_{ij}^k({}^{Q_{ij}}/_{V_{ij}})\ \forall i \in \{1, \dots, I\}$ for cross-dock center k and destination j, then sort them and create a list and put them in.
Step 3: choose the minimum positive number from step 2 and reduce it from truck capacity. After that, omit it from the list.
Step 4: choose the maximum positive number from step 2 that is not greater that truck capacity, and then reduce it from truck capacity. After that, omit it from the list. If there is no suitable number go to step 7.
Step 5: choose the minimum positive number from step 2 that is not greater that truck capacity and then reduce it from truck capacity. After that, omit it from the list. If there is no suitable number go to step 7.
Step 6: repeat step 5.
Step 7: if the truck capacity is less than 1, then $D_{kj}=D_{kj}+1$.
Step 8: repeat steps 2 to 7 for I times.
Step 9: repeat the algorithm for different i and j.

Numerical experiments

The study considers different random numerical examples and solved both approximate and exact model by LINGO software and also heuristic algorithm (Table. 1). It is clear that exact algorithm cannot only solve this problem in an efficient time but also find feasible solution in a long runtime. Although, approximate model is easier

and all previous studies formulate their model by using constraints (8) and (9), but it is obvious that this model cannot find a good solution in 3-dimentional cross-docking network and we can only consider it as a weak lower bound for the exact model results.

CONCLUSION AND FUTURE RESEARCH DIRECTIONS

In this paper, the 3-dimentional cross-docking network was introduced for the first time in order to minimize total transportation cost. This problem is so realistic and can apply in real world easily, but it is a NP-hard problem and exact solvers cannot solve this problem at all. A heuristic algorithm was presented in this study. Authors believe that this algorithm can find good solutions in an efficient time. Although approximate model results are not a good lower bound for this problem, but by comparing the result of heuristic algorithm with these values, it seems the objective function values increased rationally.

This study has lots of future research directions. First of all finding an efficient lower bound for this problem can be so useful. Secondly, by relaxing each of the initial assumptions, new research area will make.

REFERENCES

Aickelin U, Adewunmi A (2006).Simulation Optimization of the Cross-dock Door Assignment Problem. UK Operational Research Society.

Allen J (2001). Cross docking: Is it right for you?. Can. Transp. Logist., 104: 22-25.

Apte UM, Viswanathan S (2000). Effective cross-docking for improving distribution efficiencies. Int. J. Logist., 3: 91-302.

Apte UM, Viswanathan S (2002). Strategic and technological innovations in supply chain management. Int. J. Manufacturing Technol. Manage., 4: 264-282.

Bartholdi JJ, Gue KR (2000). Throughput models for unit-load cross-docking, http://web.nps.navy.mil/~krgue/Publications/tput.pdf.

Bennage WA, Dhingra AK (1995). Single and multiple objective structural optimization in discrete-continuous variables using simulated annealing. Int. J. Num. Meth. Engr., 38: 2753-2773.

Chen P, Guo Y, Lim A, Rodrigues B (2006). Multiple cross-docks with inventory and time windows. Comp. Operat. Res., 33: 43-63.

Donaldson H, Johnson EL, Ratliff HD, Zhang M (1999). Schedule-driven cross-docking networks. Technical report, Georgia Institute of Technology. <http:// www.isye.gatech.edu/apps/research-papers/papers/misc9904.pdf> Accessed, 21.04.09.

Kirkpatrick S, Gelatt CD, Vecchi MP (1983). Optimization by simulated annealing. Science, 220(4598): 671-680.

Liao C-J, Yaoming L, Stephen CS (2010). Vehicle routing with cross-docking in the supply chain. Omega, 37: 6868-6873.

Lee YH, Jung JW, Lee KM (2006). Vehicle routing scheduling for cross-docking in the supply chain. Comp. Ind. Engr., 51: 247-256.

Ley S, Elfayoumy S (2007). Cross Dock Scheduling Using Genetic Algorithms. The 2007 IEEE International Symposium on Computational Intelligence in Robotics and Automation. JacksonvilleFL USA.

Li Y, Lim A, Rodrigues B (2004). Crossdocking-JIT scheduling with time windows. J. Oper. Res. Soc., 55: 1342-1351.

Luton D (2003). Keep it moving: a cross-docking primer. Mater. Manage. Distrib., 48: 29-30.

Metropolis R, Rosenbluth K, Teller T (1953). Equation of state calculations by Fast Computing Machines. J. Chem. Phys., 21: 6.

Pardalos PM, Resende MGC (2002). Handbook of applied optimization. Oxford University Press.

Pardalos PM, Romeijn E (2002). Handbook of global optimization heuristics approaches Kluwer Academic Publishers, Vol. 2.

Rami M, Jean PA, Hosang J (2010). Ant colony optimization algorithm to solve for the transportation problem of cross-docking network. Comp. Ind. Engr., 59: 85-92.

Tsui LY, Chang, CH (1992). An optimal solution to a dock door assignment problem. Comp. Ind. Engr., 23: 283-286.