

Singapore Management University

Institutional Knowledge at Singapore Management University

Research Collection School Of Computing and Information Systems

School of Computing and Information Systems

3-2020

Vehicle routing problem for multi-product cross-docking

Aldy GUNAWAN

Singapore Management University, aldygunawan@smu.edu.sg

Audrey Tedja WIDJAJA

Singapore Management University, audreyw@smu.edu.sg

Benjamin GAN

Singapore Management University, benjamingan@smu.edu.sg

Vincent F. YU

Panca JODIAWAN

Follow this and additional works at: https://ink.library.smu.edu.sg/sis_research



Part of the [Computer Sciences Commons](#), [Operations and Supply Chain Management Commons](#), and the [Operations Research, Systems Engineering and Industrial Engineering Commons](#)

Citation

GUNAWAN, Aldy; WIDJAJA, Audrey Tedja; GAN, Benjamin; YU, Vincent F.; and JODIAWAN, Panca. Vehicle routing problem for multi-product cross-docking. (2020). *Proceedings of the International Conference on Industrial Engineering and Operations Management 2020: Dubai, UAE, March 10-12*. 66-77.

Available at: https://ink.library.smu.edu.sg/sis_research/5258

This Conference Proceeding Article is brought to you for free and open access by the School of Computing and Information Systems at Institutional Knowledge at Singapore Management University. It has been accepted for inclusion in Research Collection School Of Computing and Information Systems by an authorized administrator of Institutional Knowledge at Singapore Management University. For more information, please email cherylds@smu.edu.sg.

Vehicle Routing Problem for Multi-Product Cross-Docking

Aldy Gunawan¹, Audrey T. Widjaja¹, Benjamin Gan Kok Siew¹, Vincent F. Yu², Panca Jodiawan²

¹School of Information Systems
Singapore Management University
Singapore, 178902

aldygunawan@smu.edu.sg, audreyw@smu.edu.sg, benjamingan@smu.edu.sg

² Department of Industrial Management
National Taiwan University of Science and Technology
Taipei 106, Taiwan
vincent@mail.ntust.edu.tw, pancajodiawan@gmail.com

Abstract

Cross-docking is a logistic technique that can reduce costs occurred in a supply chain network while increasing the flow of goods, thus shortening the shipping cycle. Inside a cross-dock facility, the goods are directly transferred from incoming vehicles to outgoing vehicles without storing them in-between. Our research extends and combines this cross-docking technique with a well-known logistic problem, the vehicle routing problem (VRP), for delivering multiple products and addresses it as the VRP for multi-product cross-docking (VRP-MPCD). We developed a mixed integer programming model and generated two sets of VRP-MPCD instances, which are based on VRPCD instances. The instances are solved by a commercial software AMPL with CPLEX solver. The findings show that the small instances can be solved optimally by CPLEX. However, larger instances cannot be solved optimally within predefined computational times.

Keywords

Vehicle routing problem, Cross-docking, Multiple products, Mathematical programming model

1. Introduction

Speed and productivity of a supply chain have become important factors in the growth of organizations. A supply chain is expected to be reactive and fast, while keeping prices low. Cross-docking is a logistic technique that can reduce costs occurred in a supply chain network while increasing the flow of goods, and thus shortening the shipping cycle. Inside a cross-dock facility, the incoming goods are directly transferred to the outgoing vehicles, such that inventory is kept to a minimum level (Ladier and Alpan 2015). The advantages of using a cross-dock facility compared to traditional distribution centers are reductions in labor costs, delivery time to customers, and the need for warehouse space (Boysen and Flidner 2010, Chen and Song 2009, Galbreth et al. 2008).

Cross-docking has been widely adapted by various companies such as Walmart, as it is able to reduce costs and time occurred during the delivery process within a supply chain network. In this network, several suppliers need to supply products to several customers based on their customers' demand. Some customers may have similar demand with each other, resulting in direct shipping of products to each customer from the same supplier. In this case, each supplier needs to send one or more vehicle(s) to the customers' site in order to satisfy the individual customer demand. Each customer, on the other hand, is visited by multiple suppliers' vehicles for each product they order.

The idea of cross-docking is to first consolidate those supply products in a cross-dock facility before sending it to the customers. This may result in fewer vehicle visiting each customer. It helps to eliminate the long origin-to-destination paths and the large number of vehicles that occurred in direct shipments (Rezaei and Kheirkhah 2017). The reduction in delivery route reduces the transportation cost and time, as well as congestion. Walmart is often cited as the first

company to consider the cross-docking process in its supply chain network. It runs 85% of its products through a cross-dock facility and saves as much as 2-3% in costs compared to other companies in 1992 (Stalk et al. 1992). Other companies such as Asda, Track 'n Trail, Canadian Tire, Saks, and Sears also implemented cross-docking (White 1998, Richardson 1999, Maloney 2002, Loudin 2002, Richardson 2004). The DHL Eastern China Domestic Transportation Hub features advanced cross-dock operations and IT solutions to enhance and improve its service capabilities and offerings (https://www.dhl.com/en/press/releases/releases_2009/logistics/300609.html).

The vehicle routing problem (VRP) plays an important role in the domain of logistics and transportation (Barbarosoglu and Ozgur 1999). VRP helps to design the least-cost delivery routes from a depot to a set of customers (Eksioglu et al. 2009) and has been extensively studied by considering several side constraints such as time windows (VRPTW), capacitated vehicles (CVRP), and pickup and delivery (VRPPD). However, the integration of VRP with a cross-docking strategy has only been recently investigated. Thus, this research extends and combines the concept of cross-docking and VRP for multiple products, so-called VRP for multi-product cross-docking (VRP-MPCD). We developed a mixed integer programming model to formulate the proposed VRP-MPCD and generated two sets of VRP-MPCD instances based on VRPCD benchmark instances as described in Lee et al. (2006). The newly generated VRP-MPCD instances are then solved by commercial software AMPL with CPLEX solver. Results show that the small instances with 10-nodes can be solved optimally by CPLEX. However, larger instances with 30-nodes cannot be solved optimally within predefined computational times.

The paper is organized as follows. Section 2 presents the literature review. Section 3 introduces the mixed integer programming model of the VRP-MPCD. Section 4 presents the experimental set-up, benchmark instances, experimental results, and analysis. Finally, Section 5 concludes and provides some ideas for future works.

2. Literature Review

Cross-docking is able to reduce inventory costs and increase the flow of goods, thus shortening the shipping cycle (Ladier and Alpan 2016). In a traditional distribution center, the cost for holding inventory and order picking are high due to its heavy dependency on labor (Van Belle et al. 2012). The services provided by a cross-dock facility is able to reduce the cost of these two activities through a consolidation of various size shipments to the same destination. The consolidation requires minimal handling with little or no storage between unloading and loading of the goods. Because of the potential benefit offered by a cross-dock facility, several research studies have been published regarding the numerous applications of cross-docking. For example, determining the location of cross-dock (Sung and Yang 2008, Ross and Jayaraman 2008, Bachlaus et al. 2008), cross-docking networks (Miao et al. 2008, Musa et al. 2010, Ma et al. 2011), dock-door assignment (Jarrah et al. 2014, Cohen and Keren 2009, Yu et al. 2008), and truck scheduling (Chen and Lee 2009, Vahdani and Zandieh 2010, Arabani et al. 2011).

In order to effectively apply the services provided by a cross-docking facility, one must consider both the pickup and the delivery processes. These two processes are strongly related to the vehicle routing and scheduling problem (Lee et al. 2006). Consequently, the integration of cross-docking activities into VRP has gained momentum, resulting in a new problem called vehicle routing scheduling for cross docking. Lee et al. (2006) are the pioneer in this area of research. Wen et al. (2009) continued the research and is the first to mention this problem as the vehicle routing problem with cross-docking (VRPCD). From these initial works in VRPCD, the literature has been expanded to address different cases (Liao et al. 2010, Dondo et al. 2011, Morais et al. 2014, Yu et al. 2016, Grangier et al. 2017). In the next few paragraphs, we present several works in VRPCD.

Due to the importance of efficiently controlling the physical flow in a supply chain network, cross-docking is considered as a good method to improve the responsiveness to customer demands (Lee et al. 2006). Their study aims to fill the lack of research on the operational viewpoint of cross-docking in order to find the optimal vehicle routing schedule. Therefore, the combined cross-docking and vehicle routing scheduling problem is addressed with the objective of minimizing vehicles hiring (fixed) and traveling costs. Since this problem is considered as an NP-hard problem, Lee et al. (2006) developed a tabu search algorithm to solve the problem, and the result is compared to enumeration technique. Experimental results show that the solution's gap is less than 4% within a reasonable amount of time.

Liao et al. (2010) developed a new tabu search (TS) algorithm and compared with the one proposed by Lee et al. (2006). There are two major differences between their algorithms. One difference is Liao et al. (2010) arranged a single node to another vehicle one at a time, whereas Lee et al. (2006) exchanged nodes between two vehicles. Another difference is removing an empty vehicle is allowed while it was prohibited in Lee et al. (2006). The benchmark instances utilized to test the proposed TS algorithm consist of three different sets that are adopted from Lee et al. (2006). Liao et al. (2010) claimed that their proposed algorithm outperformed Lee et al. (2006)'s solution on average by as high as 10.6%, 36.0%, and 14.9% for each of the three benchmark instances, respectively. Moreover, their computational time was significantly less than Lee et al. (2006)'s. The proposed TS of Liao et al. (2010) only requires 0.12 seconds, 0.26 seconds, and 0.41 seconds for the three set of benchmark instances, respectively. In comparison, Lee et al. (2006)'s TS needed 2.02 seconds, 2.86 seconds, and 7.82 seconds, respectively.

Wen et al. (2009) addressed the VRPCD where a set of homogeneous vehicles is used to deliver goods from the suppliers to the corresponding customers through a cross-dock facility. The objective was to minimize the total traveled distance while respecting the time window constraints at each node and a time horizon for the whole transportation operation. Tabu search with adaptive memory procedure was proposed to solve the problem. The instances were obtained from a real setting provided by Transvision, a Danish logistics consultancy based in Copenhagen. In addition, the paper introduced a new aggressive skip procedure to the proposed heuristic search. The skip procedure narrowed down the number of moves to search, and consequently made it able to find high quality solutions within a short computational time.

Birim (2016) studied VRPCD with heterogenous capacity vehicles. All routes are started and ended at the cross-dock, and all pickup and delivery nodes are visited by only one vehicle. The objective was to minimize the total transportation costs and the fixed costs of the vehicles. The utilized instances in this problem were adapted from Lee et al. (2006) by adjusting several factors, i.e. the capacity of vehicles, the randomly generated asymmetrical transportation cost, and the addition of the fixed cost of vehicles. Additionally, among three benchmark instances proposed by Lee et al. (2006), only the smallest benchmark instance was adopted by Birim (2016). In total, there were 30 testing problems. In order to solve the problem, a simulated annealing (SA) algorithm was proposed. The proposed algorithm converged and terminated before 2 seconds except for one problem. Birim (2016) made no further assessment of the algorithm.

Hasani-Goodarzi and Tavakkoli-Moghaddam (2012) extended the problem by introducing a split vehicle routing problem (SVRP) with capacity constraint for multi-product cross-docks. The split pick-up (delivery) characteristic allows suppliers (customers) to send (receive) multiple shipments, and each node can be served by more than one vehicle. A mixed-integer linear programming model was proposed in their paper and solved by GAMS software with the objective of determining the best vehicle routes and the optimal number of utilized vehicles. There were 10 instances generated randomly in small sizes where the number of pick-up nodes ranged from 3 to 6 nodes and the number of delivery nodes ranged from 4 to 6 nodes.

Yu et al. (2016) introduced open vehicle routing problem with cross-docking (OVRPCD). The main difference between the original VRPCD and the OVRPCD is that OVRPCD occurred in an open network where the flow started from a pickup point and ended at a deliver point through a cross-dock facility without forming any loop. Generally, one may find this model in a retailer that considers outsourcing a logistic service as a cost-effective option. The objective of OVRPCD is to minimize the total of vehicle hiring cost and transportation costs. The problem was modelled as a mixed integer linear programming. A simulated annealing (SA) algorithm was proposed to solve the problem. In order to assess the performance of SA algorithm, CPLEX solver was also utilized to solve the problem. CPLEX provides the optimal result for the first two benchmark instances. The proposed SA obtained exactly the same result for these two benchmark instances with significantly less computational time. For the third benchmark instance, the proposed SA outperforms CPLEX in terms of both objective value and computational time. The gap of objective value between the solutions obtained by SA and by CPLEX is on average 0.86% for the third benchmark instance. In addition, the average computational time of the proposed SA is 2.63 seconds whereas CPLEX needs 21600 seconds in order to obtain the results.

3. Mixed Integer Programming Model

This model extends the work done by Lee et al. (2006) as described in Section 2. It models a VRP-MPCD solution using a set of homogenous vehicles for completing pickup and delivery processes of multiple products within a time horizon and considering vehicle capacity constraint.

Consider two directed network graphs $G' = (CU0, A')$ and $G'' = (SU0, A'')$, where 0 represents the cross-dock, $C = \{1, \dots, |C|\}$ is the set of customers, and $S = \{1, \dots, |S|\}$ is the set of suppliers; $A' = \{(i, j): i \neq j \in CU0\}$ and $A'' = \{(i, j): i \neq j \in SU0\}$ each refers to the set of arcs connecting two different nodes i and j . In VRP-MPCD, each supplier only supplies one type of product, and each customer may request multiple products. A set of homogeneous vehicles $V = \{1, \dots, |V|\}$ is available at the cross-dock and can be utilized to perform the supplier pickup or customer delivery process. Figure 1 illustrates the VRP-MPCD.

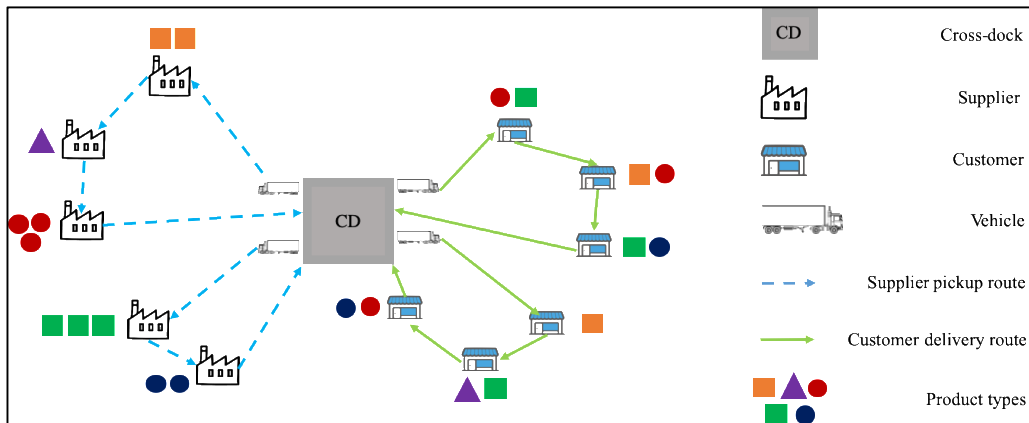


Figure 1. the VRP-MPCD

The following parameters and decision variables are used in the VRP-MPCD mathematical model.

Parameters:

- e'_{ij} distance between nodes i to j ($i, j \in \{C \cup 0\}$)
- e''_{ij} distance between nodes i to j ($i, j \in \{S \cup 0\}$)
- t'_{ij} traveling time between nodes i to j ($i, j \in \{C \cup 0\}$)
- t''_{ij} traveling time between nodes i to j ($i, j \in \{S \cup 0\}$)
- d'_{ik} demand of customer i for product type k ($i \in C, k \in S$)
- q vehicle capacity
- T_{max} planning horizon
- c travel cost per unit distance (\$/unit distance)
- L large number
- H hiring cost per vehicle

Decision Variables:

- $x'_{ij}{}^v$ 1 if vehicle v moves from node i to j in the customer delivery process and 0 otherwise ($i, j \in \{C \cup 0\}, v \in V$)
- $x''_{ij}{}^v$ 1 if vehicle v moves from node i to j in the supplier pickup process and 0 otherwise ($i, j \in \{S \cup 0\}, v \in V$)
- $q_0{}^v$ initial load of vehicle v upon leaving cross-dock in the customer delivery process ($v \in V$)
- $q''_0{}^v$ initial load of vehicle v upon leaving cross-dock in the supplier pickup process ($v \in V$)
- q'_i amount of load remaining in the vehicle upon visiting node i in the customer delivery process ($i \in C$)
- q''_i amount of load remaining in the vehicle upon visiting node i in the supplier pickup process ($i \in S$)

u'_i defines the order in which node i is visited on a tour in the customer delivery process ($i \in C$)
 u''_i defines the order in which node i is visited on a tour in the supplier pickup process ($i \in S$)

Tcd_{max} maximum traveling duration time for the customer delivery process
 Tsp_{max} maximum traveling duration time for the supplier pickup process

Min

$$c(\sum_{v \in V} \sum_{i \in \{C \cup 0\}} \sum_{j \in \{C \cup 0\}} x'_{ij}{}^v e'_{ij} + \sum_{v \in V} \sum_{i \in \{S \cup 0\}} \sum_{j \in \{S \cup 0\}} x''_{ij}{}^v e''_{ij}) + H(\sum_{v \in V} \sum_{j \in C} x''_{0j}{}^v + \sum_{v \in V} \sum_{j \in S} x''_{0j}{}^v) \quad (1)$$

The objective function (1) minimizes the total traveling and vehicle hiring costs to perform pickup process in the supplier site and delivery process in the customer site.

$$\sum_{j \in C} x''_{0j}{}^v + \sum_{j \in S} x''_{0j}{}^v \leq 1 \quad \forall v \in V \quad (2)$$

Constraint (2) ensures that each vehicle v can only be utilized or used by one of the two processes: customer delivery or supplier pickup process.

$$\sum_{i \in \{C \cup 0\}} \sum_{j \in \{C \cup 0\}, j \neq i} x'_{ij}{}^v t'_{ij} \leq Tcd_{max} \quad \forall v \in V \quad (3)$$

$$\sum_{i \in \{S \cup 0\}} \sum_{j \in \{S \cup 0\}, j \neq i} x''_{ij}{}^v t''_{ij} \leq Tsp_{max} \quad \forall v \in V \quad (4)$$

$$Tcd_{max} + Tsp_{max} \leq T_{max} \quad (5)$$

Constraints (3)-(5) relate to the traveling time limitations. Constraint (3) records the maximum time spent for performing the customer delivery process. Constraint (4) records the maximum time spent for performing the supplier pickup process. Constraint (5) ensures the total time of both customer delivery and supplier pickup processes is less than the planning horizon T_{max} .

$$\sum_{v \in V} \sum_{i \in \{C \cup 0\}, i \neq j} x'_{ij}{}^v = 1 \quad \forall j \in C \quad (6)$$

$$\sum_{i \in C} \sum_{j \in C, j \neq i} x'_{ij}{}^v \leq L \sum_{j \in C} x'_{0j}{}^v \quad \forall v \in V \quad (7)$$

$$\sum_{i \in \{C \cup 0\}, i \neq l} x'_{il}{}^v = \sum_{j \in \{C \cup 0\}, j \neq l} x'_{lj}{}^v \quad \forall l \in C, \forall v \in V \quad (8)$$

$$\sum_{i \in C} x'_{0i}{}^v \leq 1 \quad \forall v \in V \quad (9)$$

Constraints (6)-(8) link to the customer delivery process. Constraint (6) ensures all customers are visited. Constraint (7) ensures that if a vehicle visits a customer, then the vehicle needs to start the journey from the cross-dock. Constraint (8) ensures the consecutive movement of vehicles. Constraint (9) limits each vehicle to only leave the cross-dock at most once.

$$q_0{}^v = \sum_{i \in \{C \cup 0\}} \sum_{j \in C} \sum_{k \in S} d'_{jk} x'_{ij}{}^v \quad \forall v \in V \quad (10)$$

$$q_i \geq q_0{}^v - \sum_{k \in S} d'_{ik} - L(1 - x'_{0i}{}^v) \quad \forall i \in C, \forall v \in V \quad (11)$$

$$q_i \leq q_0{}^v - \sum_{k \in S} d'_{ik} + L(1 - x'_{0i}{}^v) \quad \forall i \in C, \forall v \in V \quad (12)$$

$$q'_j \geq q'_i - \sum_{k \in S} d'_{jk} - L(1 - x'_{ij}{}^v) \quad \forall i, j \in C, \forall v \in V \quad (13)$$

$$q'_j \leq q'_i - \sum_{k \in S} d'_{jk} + L(1 - x'_{ij}{}^v) \quad \forall i, j \in C, \forall v \in V \quad (14)$$

$$q_0{}^v \leq q \quad \forall v \in V \quad (15)$$

$$q'_j \leq q \quad \forall j \in C \quad (16)$$

Constraint (10) determines the total amount of load in a vehicle upon leaving cross-dock to visit customers. Constraints (11)-(14) track the remaining load in a vehicle upon delivering products to a customer. Constraints (15) and (16) limit the vehicle capacity.

$$u'_j \geq u'_i + 1 - |C|(1 - \sum_{v \in V} x''_{ij}{}^v) \quad \forall i, j \in C \quad (17)$$

Finally, Constraint (17) serves as a sub-tour elimination for the customer delivery process (for more detail explanation about sub-tour elimination, readers are referred to Laporte (1986)).

$$\sum_{v \in V} \sum_{i \in \{S \cup 0\}, i \neq j} x''_{ij}{}^v = 1 \quad \forall j \in S \quad (18)$$

$$\sum_{i \in S} \sum_{j \in S, j \neq i} x''_{ij}{}^v \leq L \sum_{j \in S} x''_{0j}{}^v \quad \forall v \in V \quad (19)$$

$$\sum_{i \in \{S \cup 0\}, i \neq l} x''_{il}{}^v = \sum_{j \in \{S \cup 0\}, j \neq l} x''_{lj}{}^v \quad \forall l \in S, \forall v \in V \quad (20)$$

$$\sum_{i \in S} x''_{0i}{}^v \leq 1 \quad \forall v \in V \quad (21)$$

Constraints (18)-(21) link to the supplier pickup process. Constraint (18) ensures all suppliers are visited. Constraint (19) ensures that if a vehicle visits a supplier, then the vehicle needs to start the journey from the cross-dock. Constraint (20) ensures the consecutive movement of vehicles. Constraint (21) limits each vehicle to only leave the cross-dock at most once.

$$q''_0{}^v = 0 \quad \forall v \in V \quad (22)$$

$$q''_i \geq q''_0{}^v - \sum_{z \in C} d'_{zi} - L(1 - x''_{0i}{}^v) \quad \forall i \in S, \forall v \in V \quad (23)$$

$$q''_i \leq q''_0{}^v - \sum_{z \in C} d'_{zi} + L(1 - x''_{0i}{}^v) \quad \forall i \in S, \forall v \in V \quad (24)$$

$$q''_j \geq q''_i - \sum_{z \in C} d'_{zj} - L(1 - \sum_{v \in V} x''_{ij}{}^v) \quad \forall i, j \in S \quad (25)$$

$$q''_j \leq q''_i - \sum_{z \in C} d'_{zj} + L(1 - \sum_{v \in V} x''_{ij}{}^v) \quad \forall i, j \in S \quad (26)$$

$$q''_0{}^v \leq q \quad \forall v \in V \quad (27)$$

$$q''_j \leq q \quad \forall j \in S \quad (28)$$

Constraint (22) ensures the initial load in a vehicle upon leaving cross-dock to visit suppliers equals zero. Constraints (23)-(26) track the remaining load in a vehicle upon picking up products from a supplier. Constraints (27) and (28) limit the vehicle capacity.

$$u''_j \geq u''_i + 1 - |S|(1 - \sum_{v \in V} x''_{ij}{}^v) \quad \forall i, j \in S \quad (29)$$

Finally, Constraint (29) serves as a sub-tour elimination for the supplier pickup process (for more detail explanation about sub-tour elimination, readers are referred to Laporte (1986)).

4. Computational Study

This section presents the computational results of the proposed mathematical model. First, we describe how to set up the experiments and generate new sets of instances. Second, we summarize and analyze the experimental results.

4.1 Experimental Set-up and Benchmark Instances

The computation was performed on a computer with Intel(R) Core™ i7-8700 CPU @ 3.20GHz processor, 32.0 GB RAM. The mathematical model for different instances was solved by commercial software, AMPL with CPLEX 12.9.0.0 solver. We modified the VRPCD benchmark instances which were formerly generated by Lee et al. (2006). The instances are divided into two sets, where each set consists of 30 problems. The first set, which is the 10-nodes set, includes four suppliers and six customers. The second set, which is the 30-nodes set, includes seven suppliers and 23 customers. It is assumed for all sets that the planning horizon, T_{max} , is 16 hours, meaning that both customer delivery and supplier pickup processes must be done within 16 hours. The parameter values for VRPCD instances are summarized in Table 1, with additional parameter p'_k as the amount of supply from supplier k ($k \in S$) and parameter d'_i as the amount of demand by customer i ($i \in C$).

Table 1. Parameter values

	Set 1	Set 2
$ S + C $	10	30
$ S $	4	7
$ C $	6	23
$ V $	10	20
T_{max}	960	960
q	70	150
H	1000	1000
c	1	1
t'_{ij}, t''_{ij}	Uniform (20,200)	Uniform (20,100)
e'_{ij}, e''_{ij}	Uniform (48,560)	Uniform (48,480)
p'_k, d'_i	Uniform (5,50)	Uniform (5,20)

Since only a single product was considered in Lee et al. (2006), we further modify the demand parameter to represent our proposed VRP-MPCD model. We first introduce the customer demand as d'_{ik} (refer to Section 3 for more details), while p'_k remains the same as one supplier only supplies one type of product. Here, p'_k is used for the purpose of generating instances only. The value of d'_{ik} is then determined such that $\sum_{k \in S} d'_{ik} = d'_i, \forall i \in C$ and $\sum_{i \in C} d'_{ik} = p'_k, \forall k \in S$ are satisfied.

We illustrate one example from a 10-nodes instance. The values of p'_k for $k \in \{S1, S2, S3, S4\}$ are 38, 34, 9, and 23, respectively, while the values of d'_i for $i \in \{C1, C2, C3, C4, C5, C6\}$ are 15, 27, 13, 20, 18, and 11, respectively. Thus, changing it to d'_{ik} by considering the above-mentioned constraints results in the allocations shown in Table 2. For example, $d'_{C1S1} = d'_{C1S4} = 0$, $d'_{C1S2} = 12$ and $d'_{C1S3} = 3$, which satisfy $\sum_{k \in \{S1, S2, S3, S4\}} d'_{C1k} = d'_{C1} = 15$.

Table 2. An example of demand allocations

	S1	S2	S3	S4	$\sum_{k \in S} d'_{ik}$
C1	0	12	3	0	15
C2	5	22	0	0	27
C3	13	0	0	0	13
C4	20	0	0	0	20
C5	0	0	0	18	18
C6	0	0	6	5	11
$\sum_{i \in C} d'_{ik}$	38	34	9	23	

4.2 Computational Results

Table 3 shows the results of VRP-MPCD solved by commercial software AMPL using CPLEX solver. The column *Instance* lists the name of 60 different problems, 30 problems from the 10-nodes set and 30 problems from the 30-nodes set. Column *Solution* presents the total cost (objective value) obtained by CPLEX. The value is followed by an * if an optimal solution was found. The last column, *CPU Time*, shows the computational time it takes to find the solutions measured in seconds.

Table 3. Computational results

Instance	Solution	CPU Time	Instance	Solution	CPU Time
10-1	6823*	1.4	30-1	7077	3635.3
10-2	6741*	1.6	30-2	6866	3600.6
10-3	9269*	0.8	30-3	7312	3609.7
10-4	7229*	1.1	30-4	6583	3600.3
10-5	6475*	1.4	30-5	6351	3632.4
10-6	7434*	0.4	30-6	5691	3662.8
10-7	11713*	0.5	30-7	7826	3600.4
10-8	8158*	0.9	30-8	7084	3600.3
10-9	6989*	1.0	30-9	6759	3600.4
10-10	5960*	1.1	30-10	7247	3646.2
10-11	6916*	1.3	30-11	6345	3691.6
10-12	7656*	1.7	30-12	6852	3601.2
10-13	6783*	1.5	30-13	7013	3612.6
10-14	4417*	0.5	30-14	6561	3600.3
10-15	7072*	1.5	30-15	6969	3637.6
10-16	8440*	0.7	30-16	6907	3601.7
10-17	9003*	0.8	30-17	6491*	3508.6
10-18	6760*	0.9	30-18	7158	3602.8
10-19	7051*	0.9	30-19	7235	3601.0
10-20	9786*	0.6	30-20	6356	3600.5
10-21	4644*	0.8	30-21	6979	3613.4
10-22	6442*	1.4	30-22	6464*	3185.0
10-23	9156*	1.2	30-23	7392	3602.8
10-24	11976*	0.5	30-24	6782	3616.4
10-25	6346*	1.6	30-25	7811	3642.3
10-26	6817*	1.2	30-26	7221	3688.5
10-27	9541*	0.6	30-27	7249	3648.0
10-28	6782*	1.4	30-28	7467	3604.2
10-29	6591*	1.0	30-29	7160	3636.6
10-30	6919*	1.2	30-30	6491	3600.9

* optimal solutions are found

All problems in 10-nodes instances can be solved optimally in less than 2 seconds. We observe that they are easy problems. However, when the number of nodes increases to 30 nodes, the problems become difficult to solve. Only two out of 30 problems can be solved optimally, which are 30-17 and 30-22. However, the computational time it takes to solve 30-nodes increases significantly when compared to 10-nodes. From less than 2 seconds to up to almost an hour. For the rest problems in the 30-nodes instances, CPLEX is not able to solve optimally within one hour of CPU time, and therefore we only report the best-found solutions so far.

We conducted further analysis for the 10-nodes instances since CPLEX is able to find optimal solutions. We focus on the utilization of the vehicles in terms of the percentage of vehicles used, as shown in Figure 2. It is observed that the percentage of vehicles used ranges from 40% to 70%. It turns out that we do not use all vehicles for serving suppliers and customers. To better utilize the vehicles, the idle vehicles can actually be allocated to other functions or for future

plans when the number of customers increases. Figure 3 presents the number of vehicles used to visit suppliers and customers. Although the number of customers $|C|$ is slightly larger than the number of suppliers $|S|$, the number of vehicles used is the same, except for four instances with a higher number of vehicles used on the supplier side.

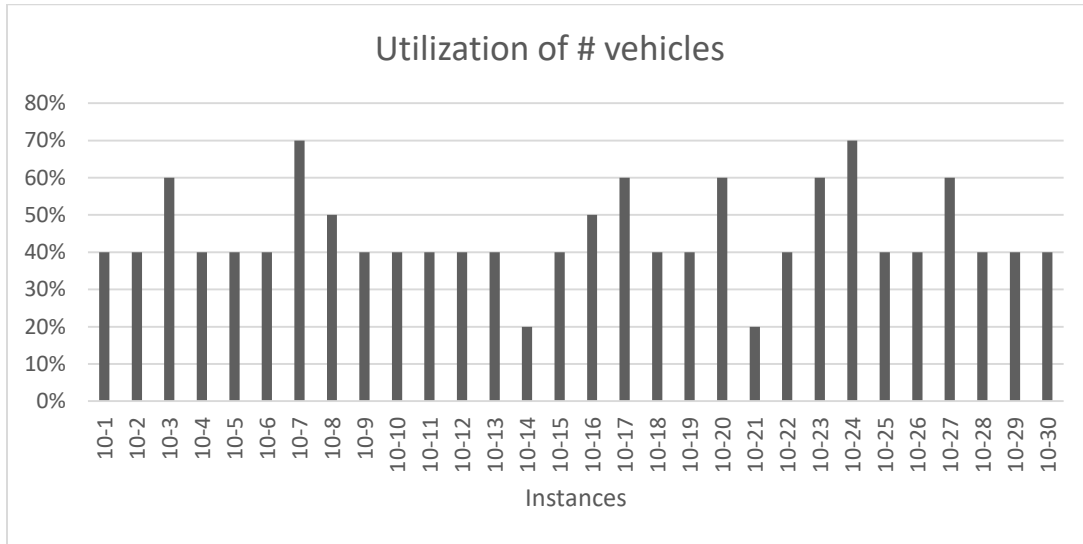


Figure 2. Utilization of vehicles for Set 1 instances

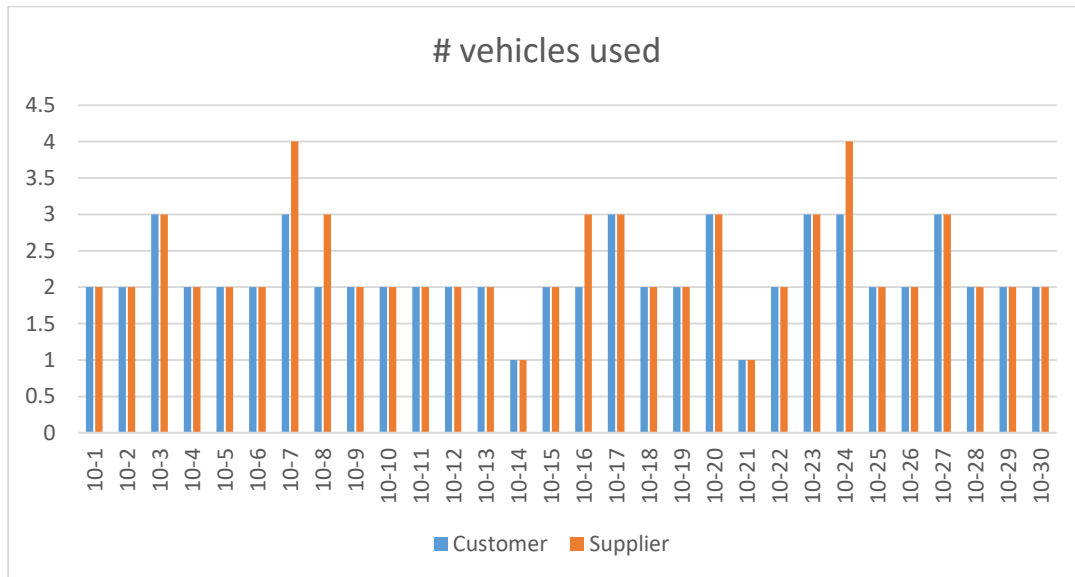


Figure 3. Number of vehicles used for customer and supplier

We further analyze the load of each vehicle in the context of how many suppliers or customers are served. For suppliers, each vehicle has different number of suppliers to visit, especially if the number of vehicles used is two. It could be due to the distances or time travel required between suppliers. For a larger number of vehicles used (e.g. 3 and 4), each vehicle visits the same number of suppliers. The same observation applies to vehicles assigned to customers.

5. Conclusion

This research introduces the extension of the VRPCD, the so-called VRP for multi-product cross-docking (VRP-MPCD), which consider the presence of multiple products in a supply chain network instead of single product (as in VRPCD). In order to solve the VRP-MPCD, a mixed integer programming model has been formulated. We consider several constraints with respect to the suppliers and customers, such as the number of vehicles to be used, delivery and pickup processes for multiple products, planning horizon for the entire process, and the vehicle capacity.

We further modify the well-known VRPCD benchmark instances to consider multiple products, so-called VRP-MPCD instances. The VRP-MPCD instances are then solved by our formulated mixed integer programming model by using a commercial software AMPL with CPLEX solver. Optimal solutions for the first set of the VRP-MPCD instances with 10-nodes can be obtained. However, the experiments show that it takes a longer computational time of over an hour to solve the problem when we tested with 30-nodes instances. Therefore, future work may include how to design a heuristic that provides good solutions and solves the problem more efficiently. The heuristic can be utilized to solve real industry data that possibly include a larger number of nodes. Another direction for future work is to validate that the presence of cross-docking facility could really reduce the transportation costs occurred when delivering products from suppliers to customers.

Acknowledgements

This research is supported by the Singapore Ministry of Education (MOE) Academic Research Fund (AcRF) Tier 1 grant.

References

- Arabani, A. R. B., Ghomi, S. M. T. F., and Zandieh, M., Meta-heuristics implementation for scheduling of trucks in a cross-docking system with temporary storage, *Expert Systems with Applications*, vol. 38, no. 3, pp. 1964-1979, 2011.
- Bachlaus, M., Pandey, M. K., Mahaian, C., Shankar, R., and Tiwari, M. K., Designing an integrated multi-echelon agile supply chain network: a hybrid taguchi-particle swarm optimization approach, *Journal of Intelligent Manufacturing*, vol. 19, no. 6, pp. 747, 2008.
- Barbarosoglu, G., and Ozgur, D., A tabu search algorithm for the vehicle routing problem, *Computers & Operations Research*, vol. 26, no. 3, pp. 255-270, 1999.
- Birim, Ş., Vehicle Routing Problem with Cross Docking: A Simulated Annealing Approach, *Proceedings of the International Strategic Management Conference*, Antalya, Turkey, October 28-30, 2016.
- Boysen, N., and Fließdner, M., Cross dock scheduling: Classification, literature review and research agenda, *Omega*, vol. 38, no. 6, pp. 413-422, 2010.
- Chen, F., and Lee, C.-Y., Minimizing the makespan in a two-machine cross-docking flow shop problem, *European Journal of Operational Research*, vol. 193, no. 1, pp. 59-72, 2009.
- Chen, F., and Song, K., Minimizing makespan in two-stage hybrid cross docking scheduling problem, *Computers & Operations Research*, vol. 36, no. 6, pp. 2066-2073, 2009.
- Cohen, Y., and Keren, B., Trailer to door assignment in a synchronous cross-dock operation, *International Journal of Logistics Systems and Management*, vol. 5, no. 5, pp. 574-590, 2009.
- Dondo, R., Méndez, C. A., and Cerdá, J., The multi-echelon vehicle routing problem with cross docking in supply chain management, *Computers & Chemical Engineering*, vol. 35, no. 12., pp. 3002-3024, 2011.
- Eksioglu, B., Vural, A. V., and Reisman, A., The vehicle routing problem: A taxonomic review, *Computers & Industrial Engineering*, vol. 57, no. 4, pp. 1472-1483, 2009.
- Galbreth, M. R., Hill, J. A., and Handley, S., An investigation of the value of cross-docking for supply chain management, *Journal of Business Logistics*, vol. 29, no. 1, pp. 225-239, 2008.
- Grangier, P., Gendreau, M., Lehuédé, F., and Rousseau, L.-M., A matheuristic based on large neighborhood search for the vehicle routing problem with cross-docking, *Computers & Operations Research*, vol. 84., pp. 116-126, 2017.
- Hasani-Goodarzi, A., and Tavakkoli-Moghaddam, R., Capacitated vehicle routing problem for multi-product cross-docking with split deliveries and pickups, *Procedia-Social and Behavioral Sciences*, vol. 62, pp. 1360-1365, 2012.

- Jarrah, A. I., Qi, X., and Bard, J. F., The destination-loader-door assignment problem for automated package sorting centers, *Transportation Science*, vol. 50, no. 4, pp. 1314-1336, 2014.
- Ladier, A.-L., and Alpan, G., Cross-docking operations: Current research versus industry practice, *Omega*, vol. 62, pp. 145-162, 2016.
- Laporte, G., Generalized Subtour Elimination Constraints and Connectivity Constraints, *The Journal of the Operational Research Society*, vol. 37, no. 5, pp. 509-514, 1986.
- Lee, Y. H., Jung, J. W., and Lee, K. M., Vehicle routing scheduling for cross-docking in the supply chain, *Computers & Industrial Engineering*, vol. 51, no. 2, pp. 247-256, 2006.
- Liao, C.-J., Lin, Y., and Shih, S. C., Vehicle routing with cross-docking in the supply chain, *Expert Systems with Applications*, vol. 37, no. 10, pp. 6868-6873, 2010.
- Loudin, A., The power of WMS, *Warehousing Management*, vol. 9, no. 10, pp. 19-20, 2002.
- Ma, H., Miao, Z., Lim, A., and Rodrigues, B., Crossdocking distribution networks with setup cost and time window constraint, *Omega*, vol. 39, no. 1, pp. 64-72, 2011.
- Maloney, D., Canadian Tire rolls out new DC, *Modern Materials Handling*, vol. 57, no. 11, pp. 15-21, 2002.
- Miao, Z., Fu, K., Fei, Q., and Wang, F., Meta-heuristic algorithm for the transshipment problem with fixed transportation schedules, *Proceedings of the International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, Wroclaw, Poland, June 18-20, 2008.
- Morais, V. W. C., Mateus, G. R., and Noronha, T. F., Iterated local search heuristics for the Vehicle Routing Problem with Cross-Docking, *Expert Systems with Applications*, vol. 41, no. 16, pp. 7495-7506, 2014.
- Musa, R., Arnaout, J.-P., and Jung, H., Ant colony optimization algorithm to solve for the transportation problem of cross-docking network, *Computers & Industrial Engineering*, vol. 59, no. 1, pp. 85-92, 2010.
- Rezaei, S., and Kheirikhah, A., Applying forward and reverse cross-docking in a multi-product integrated supply chain network, *Production Engineering*, vol. 11, no. 4-5, pp. 495-509, 2017.
- Richardson, H. L., Cross docking: information flow saves space, *Transportation & Distribution*, pp. 51-54, 1999.
- Richardson, H. L., Execution at the dock, *Logistics Today*, vol. 45, no. 4, pp. 31-33, 2004.
- Ross, A., and Jayaraman, V., An evaluation of new heuristics for the location of cross-docks distribution centers in supply chain network design, *Computers & Industrial Engineering*, vol. 55, no. 1, pp. 64-79, 2008.
- Stalk, G., Evans, P., and Shulman, L. E., Competing on capabilities: the new rules of corporate strategy, *Harvard Business Review*, vol. 70, no. 2, pp. 57-69, 1992.
- Sung, C. S., and Yang, W., An exact algorithm for a cross-docking supply chain network design problem, *Journal of the Operational Research Society*, vol. 59, no. 1, pp. 119-136., 2008.
- Vahdani, B., and Zandieh, M., Scheduling trucks in cross-docking systems: Robust meta-heuristics, *Computers & Industrial Engineering*, vol. 58, no. 1, pp. 12-24, 2010.
- Van Belle, J., Valckenaers, P., and Cattrysse, D., Cross-docking: State of the art, *Omega*, vol. 40, no. 6, pp. 827-846, 2012.
- Wen, M., Larsen, J., Clausen, J., Cordeau, J.-F., and Laporte, G., Vehicle routing with cross-docking, *Journal of the Operational Research Society*, vol. 60, no. 12, pp. 1708-1718, 2009.
- White, D., Asda floats shipshape supply plan, *Supply Management*, vol. 3, no .8, pp, 15, 1998.
- Yu, V. F., Jewpanya, P., and Redi, A. A. N. P., Open vehicle routing problem with cross-docking, *Computers & Industrial Engineering*, vol. 94, pp. 6-17, 2016.
- Yu, V. F., Sharma, D., and Murty, K. G., Door allocations to origins and destinations at less-than-truckload trucking terminals, *Journal of Industrial and Systems Engineering*, vol. 2, no. 1, pp. 1-15, 2008.

Biographies

Aldy GUNAWAN is Assistant Professor of Information Systems (Practice) at the Singapore Management University. He works among the areas of Operations Research and Artificial Intelligence. He received his Ph.D in Industrial and Systems Engineering from the National University of Singapore. His main research interests include operations research, algorithm design, and data analytics that relate to metaheuristics, algorithm configuration, design of experiments, combinatorial optimization, and automated planning/scheduling. His past studies have been published at top conferences and journals in Operations Research. He has been serving as a committee member of the Operational Research Society of Singapore.

Audrey Tedja WIDJAJA is currently working as a research engineer at the Singapore Management University. She obtained her master degree from the National Taiwan University of Science and Technology (NTUST). Her research interests include operations research and logistics/supply chain management. She has published several conference papers at various conferences, such as ICSIIT, GECCO, CASE, IEEM, and ICLS.

Benjamin GAN is Associate Professor of Information System (Education) at the Singapore Management University (SMU). He receives his Ph.D. degree in Computer Science from the University of Iowa, Iowa City. He has more than 20 years of teaching experience in IS Capstone course, Interaction Design and Prototyping and programming courses. His research interests include project based learning, improving capstone courses and supply chain analytics. His publications are in education and learning journals and conferences. He has received National Day Award Public Administration Bronze Medal and SPRING Singapore Quality and Standards Merit Award, Singapore. He received grants from DHL, NEC and IMDA.

Vincent F. YU is Professor and Chair of Industrial Management at the National Taiwan University of Science and Technology. He received his Ph.D. in Industrial & Operations Engineering from the University of Michigan, Ann Arbor. His current research interests include operations research, logistics management, soft computing, and artificial intelligence. He had published articles in Applied Mathematical Modelling, Applied Soft Computing, Computers & Industrial Engineering, Computers & Operations Research, European Journal of Operational Research, Industrial Marketing Management, International Journal of Production Research, Journal of Cleaner Production, Journal of Intelligent Manufacturing, and Omega.

Panca JODIAWAN is currently a PhD student at the Department of Industrial Management, National Taiwan University of Science and Technology (NTUST). His research interests include operations research and logistics/supply chain management. He has published several conference papers at various conferences, such as IEEM and ICLS.