A multi Operator Genetic Algorithm For Solving The Capacitated Vehicle Routing Problem with Cross-Docking Problem

Alaa Touihri
Institut Supérieur de Gestion,
University of Tunis, Tunisia
Email: alaa.touihri2011@gmail.com

Olfa Dridi
Institut Supérieur de Gestion,
University of Tunis, Tunisia
Email: dridi_o@yahoo.fr

Saoussen Krichen
Institut Supérieur de Gestion,
University of Tunis, Tunisia
Email: krichen_s@yahoo.fr

***Abstract*—This study will discuss the capacitated vehicle routing problem with simultaneous pick-up and delivery in a cross–docking environment. The transportation system includes three levels in the supply chain management:(1)Suppliers, (2)Retailers, (3)Customers. This paper addresses the CVRPCD, where a set of homogeneous vehicles are used to transport products from the suppliers to the corresponding customers via a cross-dock. The objective of the CVRPCD is to minimize the total traveled distance while respecting time window constraints at the nodes and a time horizon for the whole transportation operation. In this paper, a mixed integer programming formulation for the CVRPCD is proposed.**

I. Introduction

The advantages of the cross-docking technique have been increasingly appreciated in literature and in practice. This appreciation, coupled with the advances of numerous applications in the vehicle routing problem (VRP) across numerous practical contexts, presents an opportunity to explore the Capacitated VRP with cross-docking (CVRPCD). This study considers a single product and single cross-dock wherein capacity-homogeneous vehicles start at cross-dock and finish in the cross-dock after serving all customers(Delivery process) and all suppliers (different pickup points). The vehicles are scheduled to route in the network synchronously to arrive at the cross-dock center simultaneously. In the delivery operations, all customers must be served at most once and deliveries should be finished at a predetermined duration. We model CVRPCD as a mixed-integer linear program that minimizes the total cost (vehicle hiring cost and transportation cost). A genetic algorithm (GA) is proposed to solve the problem. GA is first verified by solving the CVRPCD benchmark instances; We have tested our algorithm on a set of VRPCD benchmarks with some modification of the benchmark to be adapted with our approach, the results are compared with those obtained by Augerat et al (2012) instances. Computational results show that GA can obtain optimal solutions to more than 80% of instances.
The remainder of this paper is organized as follows. A detailed description of the CVRPCD is given in the next section. A mixed integer formulation of the problem is then presented, followed by a hybrid heuristic approach to solve the problem. Computational results are presented and conclusion follow.

II. Related Work

Cross-docking is a new warehousing strategy in logistics. It is defined as the consolidation of products from incoming shipments so that they can be easily sorted at a distribution center for outgoing shipments. The distribution center in this case is referred to as a cross-dock. It essentially eliminates the inventory holding function of a traditional warehouse while still allowing consolidation.
The advantages of the cross-docking technique have been increasingly appreciated in literature and in practice. This appreciation, coupled with the advances of numerous applications in the vehicle routing problem (VRP) across numerous practical contexts, presents an opportunity to explore the Capacitated VRP with cross-docking (CVRPCD). Many companies are trying to develop efficient strategies to control the physical flow of their supply chain. The important aspects in finding new strategies is minimizing the total cost and achieving a high level of agility, flexibility, and reliability for various demands. Cross-docking is one innovative strategy to minimize unnecessary cost, particularly in terms of inventory and customer service level (Apte et Viswanathan, 2000). The shipments arriving from disparate sources are regrouped and dispatched directly by the outgoing trailers without being stored. Shipments typically spend less than 24 hours at the cross-dock, sometimes less than an hour. This way, cross-docking not only provides good customer service but also yields substantial advantages over traditional warehousing: reduction in inventory investment, storage space, handling cost and order-cycle time, as well as faster inventory turnover and accelerated cash flow (Cook, Gibson and MacCurdy, 2005; Apte and Viswanathan, 2000). Agustina, Lee, and Piplani (2010) noted that cross-docking is important for the efficient operation of a distribution network because it reduces or eliminates the storage activities that belong to the warehousing system. In general, the concept of cross-docking does not allow products to be stored at the cross-docking center but may occur whenever the inventory cost incurred is lower than

the gain from consolidation or a delay of shipment (Vahdani, Soltani, and Zandieh, 2009).

In the supply chain, the classical vehicle routing problem (VRP) plays an important role in distribution management and logistics, as well as the costs associated with operating vehicles (Barbarosoglu and Ozgur, 1999). VRP finds optimal delivery or pick-up routes from a depot to a set of customers subject to various side constraints (Eksioglu, Vural, and Reisman, 2009). Because of its importance, VRP has been studied extensively over the past decades with many extensions and different solution approaches (Braekers, Ramaekers, and Van Nieuwenhuyse, 2015). Among those studies, the integration of a cross-docking strategy has only been recently investigated. Several recent studies published VRP with cross-docking (VRPCD) as a variant of the classical VRP. Lee, Jung, and Lee (2006) considered such a variant to have synchronous product arrival times with stable demand for consolidation. Their objective was to find the optimal number of vehicles and routing schedule to minimize transportation cost. The results of the proposed tabu search (TS) algorithm were compared with those obtained by an enumeration method. On average, a 4% error existed in the near-optimal solutions from 1000 search iterations. Liao, Lin, and Shih (2010) proposed a new TS algorithm for the VRPCD. They used the TS algorithm to solve the set of benchmark problems introduced by Lee et al. (2006) and the results show improvements in terms of solution quality and computational time. The average improvement was as high as 1036% for problems of various sizes compare to the results obtained by TS of Lee et al. (2006). Wen, Larsen, Clausen, Cordeau, and Laporte (2009) investigated another version of VRPCD slightly different from the one introduced by Lee et al. (2006) where asynchronous arrival is allowed. The dependency among the vehicles is determined by consolidation decisions. They modeled the problem to minimize the total distance traveled. Tarantilis (2013) and Morais, Mateus, and Noronha (2014) also investigated the same problem as defined by Wen et al. (2009). Tarantilis (2013) proposed a heuristic based on the adaptive multi-restart procedure associated with a TS heuristic to solve VRPCD, which provides better solutions than the solutions obtained by Wen et al. (2009) for 14 out of 20 instances. Morais et al. (2014) applied the iterated local search heuristic (ILS) to solve VRPCD. Their computational results showed that ILS outperformed the tabu search heuristic proposed by Wen et al. (2009) and the adaptive multi-restart TS heuristic of Tarantilis (2013). Hasani-Goodarzi and Tavakkoli-Moghaddam (2012) applied the cross–docking strategy for a vehicle fleet that was allowed to make split deliveries and pick-ups in different nodes of the network. They called this variant the split VRP, which was formulated as a mixedinteger programming model that aims to minimize transportation cost by using the GAMS optimization software.Mousavi and Tavakkoli–Moghaddam (2013) proposed the location and routing scheduling problems with cross–docking which aims to design a cross–dock location and a vehicle routing scheduling model. The algorithm based on a two–stage hybrid simulated annealing (HSA) with a tabu list in the TS algorithm is proposed to solve the problem.Mousavi, Tavakkoli–Moghaddam, and Jolai (2013) studied the location and VRP in the cross–docking distribution networks under uncertainty, and proposed a hybrid fuzzy possibilistic-stochastic programming solution approach. Agustina, Lee, and Piplani (2014) integrated cross-docking, vehicle scheduling and routing in food supply chain to ensure that food can be delivered to customers just in time. They formulated the problem as a mixed integer linear program and used the concepts of customer zones and hard time windows for delivery to reduce the solution space and then solved the problem by CPLEX.Kkoglu and ztrk (2015) introduced VRPCD with 2–dimensional truck loading. They hybridized TS with simulated annealing (SA) algorithm to solve the problem. The combinatorial nature of VRP makes this type of problem an NP-hard problem. Thus, studies with the same intrinsic complexity usually use heuristic and meta-heuristic solution approaches. For example, Dondo and Cerd (2013) proposed a sweep heuristic algorithm and Morais et al. (2014) used an iterated local search heuristic to solve VRPCD. Pisinger and Ropke (2007) used an adaptive large neighborhood search heuristic algorithm to solve five different variants of VRP, including the vehicle routing problem with time windows, capacitated vehicle routing problem, multi-depot vehicle routing problem, and site-dependent vehicle routing problem. Some studies employed meta–heuristic approaches to solve VRP and its variants, such as tabu search (Gendreau, Hertz, & Laporte, 1994;Gendreau, Laporte, Musaraganyi, & Taillard, 1999; Lee et al., 2006;Liao et al., 2010), genetic algorithm (Baker & Ayechew, 2003; Hwang, 2002; Kergosien, Lent, Billaut, & Perrin, 2013), simulated annealing (Lin, Yu, & Chou, 2009; Wang, Mu, Zhao, & Sutherland,2015; Yu & Lin, 2014, 2015a, 2015b; Yu, Lin, Lee, & Ting, 2010), particle swarm optimization (Ai & Kachitvichyanukul, 2009a, 2009b; Kachitvichyanukul, Sombuntham, & Kunnapapdeelert, 2015; MirHassani & Abolghasemi, 2011), and some recently developed hybrid heuristic algorithms (Goksal, Karaoglan, & Altiparmak,2013; Ho, Ho, Ji, & Lau, 2008; Marinakis & Marinaki, 2010;Mousavi & Tavakkoli-Moghaddam, 2013; Subramanian, Penna,Uchoa, & Ochi, 2012; Subramanian, Uchoa, & Ochi, 2013; Yu, Ding, & Zhu, 2011). The computational results of these studies show that these hybrid approaches can find optimal or near-optimal solutions to large-scale problems in a competitive computational time.

The problem considered in our study is the Capacitated Vehicle Routing Problem with Cross-Docking (CVRPCD). The problem is similar to that of Lee, Jung and Lee (2006)(VRPCD) where a heterogenous vehicle can pick-up or deliver more than one supplier or customer, and the pick-up and delivery routes start and end at the cross-dock. However, there is no constraint on simultaneous arrival for all the vehicles in our problem. Instead, the dependency among the vehicles is determined by the consolidation decisions. Moreover, each pick-up and delivery has predetermined time windows.

III. Problem statement and mathematical model

This study tackle three levels in supply chain management which are (suppliers, retailers'cross-docks' and customers). The relation between suppliers and cross-docks is called the pick-up process while the relation between cross-docks and customers is called the delivery process. Let p is a set of pick-up nodes, and D is a Set of delivery nodes and O is the cross-dock. The goal is to pick up a set of products from suppliers and deliver them to a set of Customers Inquiries according to their demands through a set of available vehicles M via cross–docking O. The mathematical model can be divided into two parts: the pick-up process model and delivery process model. Constraints are added to the model to cover the general situations that often occur in the distribution network. Moreover, to address the cross–docking network, we introduce a cross-dock in the model. The cross-dock was assumed to connect to all possible nodes with a distance not null. We provide the notations used in formulating CVRPCD in the following.

IV. Mathematical formulation

A. Notation

We enumerate in the main symbols used throughout this paper.

TABLE I
Notations

Symbols	Description
P	Set of pick-up nodes; $P = 1, 2, 3, ..., p$
D	Set of delivery nodes; D = 1, 2, 3, ... , d
O	Cross-dock
M	Set of available vehicles;$M = 1, 2, 3, ..., m$
Q_i	Vehicle capacity
d_i	Quantity of products to be collected at pick-up node i
D_j	Quantity of products to be delivered to delivery node j
ci, j	Transportation cost from node i to node j in the pick-up process
$c'_{i,j}$	Transportation cost from node node j in the deivery process

B. Mathematical Model

The CVRPCD is formulated as:

$$Min \quad Z(x) = \sum_{i,j \in E} \sum_{k \in K} c_{ij} x_{ij}^k \tag{1}$$

The first objective function computes the total transportation cost incurred in the pick-up processes.

- Pick-up process

$$\sum_{i \in O \cup P} \sum_{i \neq j, k \in M} x_{i,j}^k \leq 1 \forall j \in P \tag{2}$$

$$\sum_{j \in O \cup P} \sum_{j \neq i, k \in M} x_{i,j}^k \leq 1 \forall i \in P \tag{3}$$

$$\sum_{i,j} x_{Oj}^k \leq 1 \forall k \in M \tag{4}$$

$$\sum_{i,j} x_{iO}^k \leq 1 \forall k \in M \tag{5}$$

$$\sum_{i,j} D_i x_{i,j}^k \leq Q_i \forall i, j \in P \tag{6}$$

$$\sum_{i \in p} p_i x_{ij}^k \tag{7}$$

In the pick-up process, constraints (2) and (3) express that only one vehicle can arrive at and depart from every pick-up node. Constraint (4) ensures the consecutive movement of vehicles. The vehicles leave the cross-dock as stated and are required to visit the cross-dock immediately after the last pick-up node as stated in constraint. Constraints (5) and (6) calculate the total amount of products that have been collected by a vehicle when the vehicle leaves a pick-up node. Constraint (7) calculates the number of vehicles used in the pick-up process.

$$Min \quad Z(x) = \sum_{i,j \in E} \sum_{k \in K} c'_{i,j} + x'^k_{ij} \tag{8}$$

The second objective function computes the total transportation cost incurred in the delivery processes

- Delivery Process

$$\sum_{i \in O \cup P} \sum_{i \neq j, k \in M} x'^k_{i,j} \leq 1 \forall j \in D \tag{9}$$

$$\sum_{j \in O \cup P} \sum_{j \neq i, k \in M} x'^k_{i,j} \leq 1 \forall i \in D \tag{10}$$

$$\sum_{i,j} x'^k_{Oj} \leq 1 \forall k \in M \tag{11}$$

$$\sum_{i,j} x'^k_{iO} \leq 1 \forall k \in M \tag{12}$$

$$\sum_{i,j} D_i x'^k_{i,j} \leq Q_i \forall i, j \in D \tag{13}$$

$$\sum_{i \in p} p_i x'^k_{ij} \tag{14}$$

In the delivery process, we employ the same constraints. Constraints (9) and (10) are included to ensure that only one vehicle arrives at and leaves from every delivery node. Constraint (11)warrants the consecutive movement of vehicles. Vehicles return to the cross-dock as stated in constraint and are required to start from the cross-dock as stated in constraint. Constraints 12 calculate the total amount of products that have been delivered by a vehicle when the vehicle leaves a delivery node. Constraint (13) ensures that the total amount of products delivered by a vehicle does not exceed the vehicles capacity Q. The number of vehicles used in the delivery process is calculated by constraint (14).

Decision variables

$$x_{ij}^{k}\begin{cases}1: if\ vehicle\ k\ moves\ from\ node\ i\ to\ node\ j\ in\\ \quad the\ pick-up\ process;\\ \\ 0: Otherwise\end{cases}$$

$$x_{ij}^{\prime k}\begin{cases}1: if\ vehicle\ k\ moves\ from\ node\ i\ to\ node\ j\ in\\ \quad the\ delivery\ process;\\ \\ 0: Otherwise\end{cases}$$

Example 1

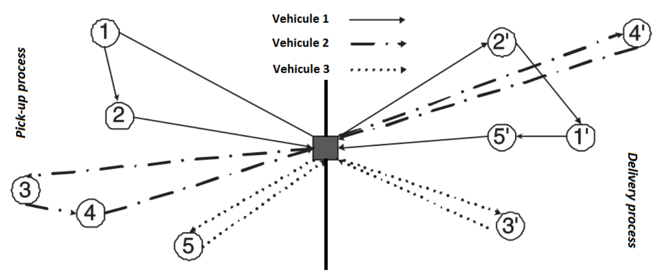

Fig. 1. Illustration of CVRPCD

To strengthen the understanding of the description of the problem, let consider the following example where P is a set of 5 pick-ups nodes $p = (1, 2, 3, 4, 5)$ and D is a set of 5 delivery nodes $D = (1', 2', 3'4', 5')$ and M is a set of 3 vehicles $M = (v_1, v_2, v_3)$ the transportation cost of the vehicle from i to j is represented by vector of both pick-up and delivery.
The cost is different between vehicles because of their proprieties.

	P_1	P_2	P_3	P_4	P_5
V_i	6	3	7	5	2

Same thing for the delivery process:

	D_1	D_2	D_3	D_4	D_5
V_i	2	6	1	7	15

The result is a matrix that shows the nodes visited by each vehicle and the cost of consumed pick-up.

$$P = \begin{pmatrix}0 & 1 & 0 & 1 & 1\\ 0 & 0 & 1 & 0 & 0\\ 1 & 0 & 0 & 0 & 0\end{pmatrix} = \begin{pmatrix}10\\ 7\\ 6\end{pmatrix}$$

The result of the delivery process:

$$D = \begin{pmatrix}0 & 1 & 0 & 1 & 1\\ 0 & 0 & 1 & 0 & 0\\ 1 & 0 & 0 & 0 & 0\end{pmatrix} = \begin{pmatrix}28\\ 1\\ 2\end{pmatrix}$$

The final result means the total cost consumed by each vehicle that is an addition of the two matrices P and D.

$$P + D = \begin{pmatrix}10+28\\ 7+1\\ 6+2\end{pmatrix} = \begin{pmatrix}38\\ 8\\ 8\end{pmatrix}$$

the final results of the project is the summation of all vehicles costs that is equal in this example:
38+8+8 = 54

V. GA-CVRPCD

CVRPCD belongs to the class of NP-hard problems (Avci & Topaloglu, 2016; Yu, Jewpanya, & Redi, 2016; Zachariadis, Tarantilis, & Kiranoudis, 2015; Zare-Reisabadi & Mirmohammadi, 2015), for that reason the exact solution methods become highly time–consuming as the problem instances increase in size. Therefore, due to the combinatorial nature of the CVRP and the GA's efficiency in solving combinatorial problems, a GA based approach is developed to solve the vehicle routing problem with cross-docking.
GA's can easily be adapted to various types of problems therefore many different GA approaches exist depending on the problems studied. There are several ways to maintain the population and several GA operators. However, all GA approaches must have a good genetic representation of the problem, an initial population generator, appropriate fitness function, and genetic operators such as crossover and mutation in order to work effectively.

A. GA-CVRPCD operators:

1) Initial population:: The encoding of a chromosome is designed in a vector form that expresses the vehicle in which we assigned the nodes travelled in the pick–up and delivery process. For example, in table 2 nodes are assigned to one vehicle in the pick-up process and it's the same representation for the delivery process.

Chromosomes	6	1	2	4	3	5

Fig. 2. Individual Encoding for the pick-up process

In figure 3, we show a set of chromosomes (Population) in which each line represent a solution and each gene represent a node from the pick-up process.

$$\begin{array}{ccccccc} s_0 & 4 & 1 & 2 & 6 & 3 & 5\\ s_1 & 5 & 2 & 1 & 6 & 3 & 4\\ . & . & . & . & . & . & .\\ . & . & . & . & . & . & .\\ . & . & . & . & . & . & .\\ \mathrm{s}_n & 1 & 4 & 5 & 3 & 2 & 6\end{array}$$

Fig. 3. Population encoding

As numerous, methods to select the best chromosomes, roulette wheel selection, Boltzman selection, tournament selection, rank selection, steady state selection and some others. In our algorithm we used to select the chromosome from using the roulette wheel selection.

2) Crossover operator: Two types of crossover operators are considered: one-point and two-point crossover.

- **One-point Crossover**:
  Two individuals, denoted as Parent 1 and Parent 2 are selected, Then an integer number q is generated randomly between 1 and J to obtain two new individuals: Son 1{1....q} and Son 2{q....j}. Pick-up or delivery Nodes in positions $i = 1...q$ in Son 1 are taken form Parent 1. Pick-up or delivery Nodes in positions $i = q + 1; ...; J$ in Son 1 are taken from Parent 2. As an example, let us consider Parent 1 {1,3,2,5,4,6 } and Parent 2 {2,4,6,1,3,5}. With $q = 3$, Son 1 {1,3,2,4,6,5}.

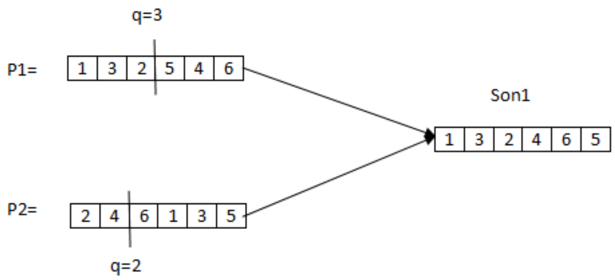

Fig. 4. One point crossover

- **Two-point Crossover:**
  Two-point crossover is an extension of one-point crossover. Two integer numbers q_1 and q_2 are randomly generated with $1 \leq q_1 \leq q_2 \leq j$. Now, Son 1 is generated with nodes list on positions $i = 1.....q_1$ taken from Parent 1, nodes in positions $i = q_1 + 1; ...; q_2$ are taken from Parent 2 and finally positions $i = q_2 + 1, ..., j$ are again taken from Parent 1.
  Taken the same example of the previous subsection, let us consider Parent 1 {1,3,2,5,4,6 } and Parent 2 {2,4,6,1,3,5}. With $q_1 = 1$ and $q_2 = 2$ Son 1 is {1,2,4,3,5,6 }.

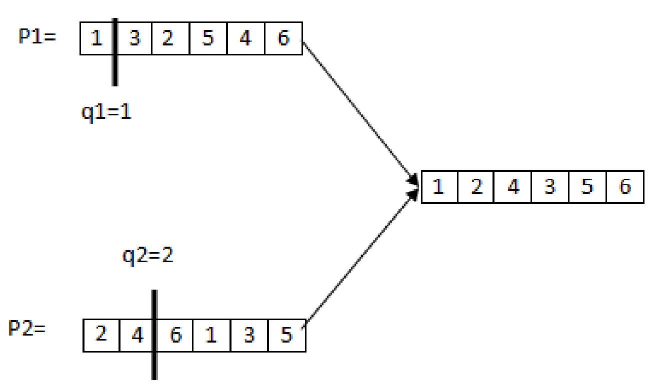

Fig. 5. Two-point crossover

3) Mutation operator: Given the chromosome based on the nodes-list $P + D$, the mutation operator modifies the structure of the chromosome as follows. For every position $i \doteq 1......j - 1$, nodes j_i and j_{i+1} switch their positions with probability $p_{mut} \in [0, 1]$ if the resulting nodes-list continues to meet the satisfaction of constraints. Mutation operator may create nodes-list that have not yet been created by crossover operators.
Only the index 0 of the cross-dock can't be switched.

0	1	3	8	2	7	0	**1**	4	5	0
0	1	3	8	2	7	0	**8**	4	5	0

Fig. 6. Mutatation for a chromosomes

B. GA-CVRPCD Algorithm:

The inputs of the algorithm are respectively the Delivery and pick-up nodes demands, the vehicle capacity and the number of nodes P and D.
The best solution of the procedure is the Routes with minimum cost traveled for serving pick-ups and delivery nodes.

INPUTS	- Delivery and pick-up nodes demands - Vehicle capacity - Number of nodes P, D
OUTPUTS	- Best solution: Routes with minimum cost traveled for serving pick-ups and delivery nodes

In order to run the algorithm, This algorithm consists of 6 steps: First Generate initial population then we have to evaluate the population, select the chromosomes, Perform crossover on selected elements, perform mutation on selected elements and finally evaluate the new generation of elements. We repeat all this instruction until the stopping criteria is met.

Algorithm 1 Genetic Algorithm (GA-VCRPCD)

Step1: Initialization
1. set t=0
2. Generate initial population P
3. Evaluate The population p

Algorithm For g=2 to N-1 DO
4. While [not termination condition] do
5. Select some elements from P to copy in p'
6. Perform crossover on selected elements of P and move them to P'
   - One-point crossover
   - Two-point crossover
7. Create p' from p by mutation operator
8. Evaluate P'
9. P = P'

VI. COMPUTATIONAL STUDY ON BENCHMARKS

Our algorithm is coded in JAVA using Netbeans in Win64 mode. All tests were performed on a computer with an Intel Core I5-2410M CPU 2.30 GHz with 4 GB of RAM. In all our experiments, we have used a population size of 100 individuals. We run our GA until a maximum number of iterations set depending on the problem size. In order to examine and evaluate the performance of our GA-CVRP, we run it on all the instances of Augerat, et al(2012) in Class 'A','B' and 'P'.

For the instances in the class 'A', both customer locations and demands are random. The instances in class 'B', however, are clustered instances. The instances in class 'P' are modified versions of instances from the literature.

We applied some modification on the benchmarks to adapt our approach GA-CVRPCD.

A. Augerat, et al. Instances

TABLE II
PARAMETERS OF AUGERAT, ET AL.'CLASS 'A','B', 'P' BENCHMARKS

Parameter	Description
M	Number of Vehicles
n	Number of Nodes
Q	Capacity of Vehicles
$D+P$	Total OF Pick-up and Delivery Nodes
$Optimal$	Minimum Cost

Finally, we start our algorithm with a population size equal to 50 with 150 number of iteration. We tested the GA-CVRP on the class of Augerat, et al(2012) benchmark and we found many best known. To present our results, we used the same parameters with some modifications like changing the distance matrix with a cost matrix in the three tables which are respectively IV, V, and VI, then we added a cross dock position generated randomly in the benchmark and we doubled the number of nodes and the number of demands for the delivery process.

- Class: A, B, P, A set of Instances
- Augerat benchmark: Best Known Value
- GA-CVRPCD: The results of our procedure
- AVG: the sum of all GAP divided by the number of instances for each class
- Hits: the number of appositions of results of GA-CVRPCD among the instances of Augerat benchmark.
- GAP:

$$GAP = \frac{Best\ Known\ Value -\ GACVRP\ Value}{100} \tag{15}$$

- AVG: The sum of all Gap devised by all instances for each Class

TABLE III
RESULTS GA-CVRPCD "CLASS A"

Class (SET A)	Augerat benchmark	GA-CVRPCD	GAP
A-n32-k5	784	700	0.84
A-n33-k5	661	661	0
A-n33-k6	742	740	0.02
A-n34-k5	778	775	0.03
A-n36-k5	799	603	1.96
A-n37-k5	669	627	0.42
A-n37-k6	949	926	0.23
A-n38-k5	730	689	0.41
A-n39-k5	822	749	0.73
A-n39-k6	831	807	0.24
A-n44-k6	937	937	0
A-n45-k6	944	944	0
A-n45-k7	1146	1146	0
A-n46-k7	914	914	0
A-n48-k7	1073	1073	0
A-n53-k7	1010	1010	0
A-n54-k7	1167	1159	0.08
A-n55-k9	1073	1086*	0.13
A-n60-k9	1354	1345	0
A-n61-k9	1034	1034	0
A-n62-k8	1288	1288	0
A-n63-k9	1616	1616	0
A-n63-k10	1314	1314	0
A-n64-k9	1401	1395	0.06
A-n65-k9	1174	1174	0
A-n69-k9	1159	1171*	0.12
A-n80-k10	1763	1763	0
AVG	-	-	**1.69**

TABLE IV
RESULTS GA-CVRPCD "CLASS B"

Class (SET B)	Augerat benchmark	GA-CVRPCD	GAP
B-n31-k5	672	672	0
B-n34-k5	788	693	0.95
B-n35-k5	955	948	0.07
B-n38-k6	805	805	0
B-n39-k5	549	549	0
B-n41-k6	829	823	0.06
B-n43-k6	742	738	0.04
B-n44-k7	909	701	2.08
B-n45-k5	751	751	0
B-n45-k6	678	678	0
B-n50-k7	741	741	0
B-n50-k8	1312	1312	0
B-n51-k7	1018	978	0.5
B-n52-k7	747	739	0.08
B-n56-k7	707	541	0.66
B-n57-k7	1144	1119	0.46
B-n57-k9	1598	1598	0
B-n63-k10	1496	1496	0
B-n64-k9	861	843	0.18
B-n66-k9	1316	1093	2.23
B-n67-k10	1032	1032	0
B-n68-k9	1272	1272	0
B-n78-k10	1221	1221	0
AVG	-	-	**0.317**

TABLE V
RESULTS GA-CVRPCD "CLASS P"

Class (SET P)	Augerat benchmark	GA-CVRPCD	GAP
P-n16-k8	450	443	0.07
P-n19-k2	212	212	0
P-n20-k2	216	216	0
P-n21-k2	211	211	0.02
P-n22-k2	216	204	0.12
P-n22-k8	590	592*	0.02
P-n23-k8	529	529	0
P-n40-k5	458	452	0.06
P-n45-k5	510	503	0.07
P-n50-k7	554	554	0
P-n50-k8	629	622	0.07
P-n50-k10	696	536	1.60
P-n51-k10	741	733	0.08
P-n55-k7	568	557	0.11
P-n55-k10	694	694	0
P-n55-k15	945	945	0
P-n60-k10	744	744	0
P-n60-k15	968	968	0
P-n65-k10	792	788	0.04
P-n70-k10	827	822	0.05
P-n76-k4	593	593	0
P-n76-k5	627	627	0
P-n101-k4	681	681	0
AVG	-	-	**0.743**

Table III displays a comparison between the Augerat et al. (2012) instances (first class "CLASS A") with the results generated from our approach "GA-CVRPCD". Like we said before, for the instances in the class 'A', both customer locations and demands are random. From 27 instances in the class 'A' we found 13 Best known solutions and we improved 2 instances witch are "A-n55-k9" and "A-n69-k9".

Table IV displays a comparison between the Augerat et al(2012) instances (second class "CLASS B") with the results generated from our approach "GA-CVRPCD". The instances in class 'B, however, are clustered instances. Since we changed the parameters of the benchmark, we found only 12 best known solutions from 23 instances. Table V shows the performance of our approach cause we found 12 best known solutions from 23 instances and we improved 1 instance which is P-n22-k8. Our algorithm is executed 95 time for each class of Augerat et al.(2012) instances to ensure that our results are feasible. We note that the instances in class "P" are modified versions of instances from the literature. Figure 7 represents the number of best Known solutions found by comparison to Augerat et al(2012) instances.

B. *Van Breedam instances*

We tested the GA-CVRP on the class of Van Breedam instances and we found all the best known. To present our results, we used the same parameters with some modifications like changing the distance matrix with cost matrix in the three tables which is respectively in table IV, V and VI.

VII. Conclusion

The Capacitated Vehicle Routing Problem with cros–docking is a challenging problem that can be applied to a wide variety of practical applications. In this paper, we developed a framework which represent a new procedure noted GA-CVRPCD based on the genetic algorithm technique, to solve the Capacitated Vehicle Routing Problem with Cross–Docking.

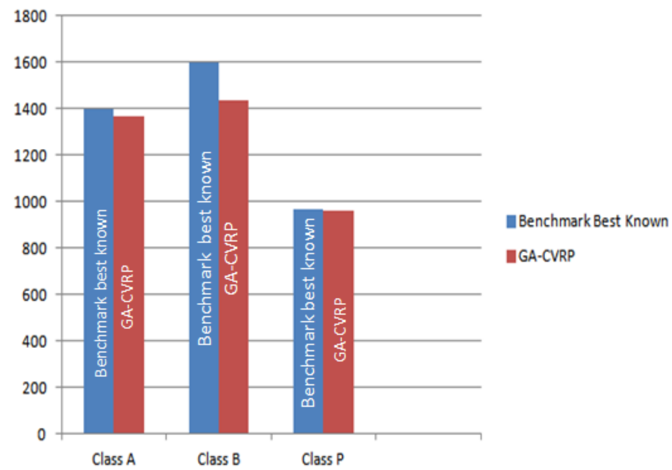

Fig. 7. A graphical illustration of the performances of the GA-CVRPCD for Augerat et al. instances

TABLE VI
Parameters of Van Breedam Benchmarks

Description	Value
VRP instances	60
Number of depot(Cross-dock)	1
Number of demands Q	10
Nr. of Vehicles	Unlimited
Capacity of Vehicles C	50

GA-CVRPCD is easy to understand, easy to follow, and easy to implement. It generated competitive quality solutions in a regular frame time. When applied to 73 August et al.(201) all instances, GA-CVRPCD is found to be the best known or optimal solutions to 37 instances found as three new best solutions on instances. In further work, we hope to investigate other variants of the Vehicle routing problem with cross–docking and try to apply our procedure for the multi-mode CVRPCD.

Conclusion

this paper investigates the capacitated vehicle routing problem with cross-docking. Due to its combinatorial nature and wide range applicability, we proposed a new framework based on Genetic Algorithm techniques to handle it.

The efficiency of the GA-CVRPCD is ascertained by means of computational experiments using benchmark problems from August et al.(2012).

the empirical results show that the proposed framework is efficient for the CVRPCD.

As future investigations, other variants of the CVRP with cross-docking and multi-mode will be considered under sophisticated metaheuristics for efficiently solving large-scale instances.

ACKNOWLEDGEMENTS

This work has been funded by the Ministry of Higher Education and Scientific Research in Tunisia, in cooperation with the Higher Institute of Management http://www.isg.rnu.tn. We are so grateful for their support to us.

REFERENCES

[1] Apte, U. M., & Viswanathan, S. (2000). Effective cross docking for improving distribution efficiencies. International Journal of Logistics, 3(3), 291-302.

[2] Wen, M., Larsen, J., Clausen, J., Cordeau, J. F., & Laporte, G. (2009). Vehicle routing with cross-docking. Journal of the Operational Research Society, 60(12), 1708-1718. Chicago

[3] Agustina, D., Lee, C. K. M., & Piplani, R. (2010). A Review: Mathematical Modles for Cross Docking Planning. International Journal of Engineering Business Management, 2(2), 47-54. Chicago.

[4] Amiri, M., Zandieh, M., Soltani, R., & Vahdani, B. (2009). A hybrid multi-criteria decision-making model for firms competence evaluation. Expert Systems with Applications, 36(10), 12314-12322. Chicago

[5] Barbarosoglu, G., & Ozgur, D. (1999). A tabu search algorithm for the vehicle routing problem. Computers & Operations Research, 26(3), 255-270.

[6] Eksioglu, B., Vural, A. V., & Reisman, A. (2009). The vehicle routing problem: A taxonomic review. Computers & Industrial Engineering, 57(4), 1472-1483.

[7] Braekers, K., Ramaekers, K., & Van Nieuwenhuyse, I. (2015). The Vehicle Routing Problem: State of the Art Classification and Review. Computers & Industrial Engineering.

[8] Aad, G., Abajyan, T., Abbott, B., Abdallah, J., Khalek, S. A., Abdelalim, A. A., ... & AbouZeid, O. S. (2013). Jet energy resolution in proton-proton collisions at sqrt mathrm s= 7 mbox TeV recorded in 2010 with the ATLAS detector. The European Physical Journal C, 73(3), 1-27.

[9] 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.

[10] Snchez-Vioque, R., Polissiou, M., Astraka, K., de los Mozos-Pascual, M., Tarantilis, P., Herraiz-Pealver, D., & Santana–Mridas, O. (2013). Polyphenol composition and antioxidant and metal chelating activities of the solid residues from the essential oil industry. Industrial Crops and Products, 49, 150-159.

[11] Vahdani, B., Mousavi, S. M., Tavakkoli-Moghaddam, R., & Hashemi, H. (2013). A new design of the elimination and choice translating reality method for multi–criteria group decision-making in an intuitionistic fuzzy environment. Applied Mathematical Modelling, 37(4), 1781-1799.

[12] Rothman, L. S., Gordon, I. E., Babikov, Y., Barbe, A., Benner, D. C., Bernath, P. F., ... & Campargue, A. (2013). The HITRAN 2012 molecular spectroscopic database. Journal of Quantitative Spectroscopy and Radiative Transfer, 130, 4-50.

[13] Aad, G., Abajyan, T., Abbott, B., Abdallah, J., Khalek, S. A., Abdelalim, A. A., ... & AbouZeid, O. S. (2012). Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC. Physics Letters B, 716(1), 1-29.

[14] Silvestro, R., Fitzgerald, L., Johnston, R., & Voss, C. (1992). Towards a classification of service processes. International journal of service industry management, 3(3), 62-75.

[15] Patel, B. H., Percivalle, C., Ritson, D. J., Duffy, C. D., & Sutherland, J. D. (2015). Common origins of RNA, protein and lipid precursors in a cyanosulfidic protometabolism. Nature Chemistry, 7(4), 301.

[16] Lip, G. Y., Windecker, S., Huber, K., Kirchhof, P., Marin, F., Ten Berg, J. M., ... & Zeymer, U. (2014). Management of antithrombotic therapy in atrial fibrillation patients presenting with acute coronary syndrome and/or undergoing percutaneous coronary or valve interventions: a joint consensus document of the European Society of Cardiology Working Group on Thrombosis, European Heart Rhythm Association (EHRA), European Association of Percutaneous Cardiovascular Interventions (EAPCI) and European Association of Acute Cardiac Care (ACCA) endorsed by the Heart Rhythm Society (HRS) and Asia-Pacific Heart .... European heart journal, 35(45), 3155-3179.

[17] 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.

[18] Augerat, P., Belenguer, J. M., Benavent, E., Corbern, A., Naddef, D., & Rinaldi, G. (1995). Computational results with a branch and cut code for the capacitated vehicle routing problem. IMAG.

[19] Vincent, F. Y., Jewpanya, P., & Redi, A. P. (2016). Open vehicle routing problem with cross-docking. Computers & Industrial Engineering, 94, 6-17.

[20] Lee, C. Y., Lee, Z. J., Lin, S. W., & Ying, K. C. (2010). An enhanced ant colony optimization (EACO) applied to capacitated vehicle routing problem. Applied Intelligence, 32(1), 88-95.

[21] De, A., Mamanduru, V. K. R., Gunasekaran, A., Subramanian, N., & Tiwari, M. K. (2016). Composite particle algorithm for sustainable integrated dynamic ship routing and scheduling optimization. Computers & Industrial Engineering, 96, 201-215.

[22] Baldacci, R., Mingozzi, A., & Roberti, R. (2012). Recent exact algorithms for solving the vehicle routing problem under capacity and time window constraints. European Journal of Operational Research, 218(1), 1-6.