The Two-Echelon Multi-products Location-Routing problem with Pickup and Delivery: Formulation and heuristic approaches

Younes Rahmani, Wahiba Ramdane, Ammar Oulamara

The Two-Echelon Multi-products Location-Routing problem with Pickup and Delivery: Formulation and heuristic approaches

Younes Rahmani[1], Wahiba Ramdane Cherif - Khettaf[1] and Ammar Oulamara[2]

[1] *LORIA Laboratory, Lorraine University, Nancy, France*

[2] *University of Lorraine, Ile de Saulcy, 57045, Metz, France*

Abstract. The two-echelon location routing problem (LRP-2E) considers the first-level routes that serve from one depot a set of processing centers, which must be located and the second-level routes that serve customers from the open processing centers. In this paper, we consider an extension of the LRP-2E, where the second level routes include three constraints, that have not been considered simultaneously in the location routing literature, namely, multi-product, pickup and delivery and the use of the processing center as intermediate facility in the second-level routes. This new variant is named two-Echelon Multi-products Location-Routing problem with Pickup and Delivery (LRP-MPPD-2E). The objective of LRP-MPPD-2E is to minimize both the location and the routing costs, considering the new constraints. The first echelon deals with the selection of processing centers from a set of potential sites simultaneously with the construction of the first-level routes, such that each route starting from the main depot, visits the selected processing centers and returns to the main depot. The second echelon aims at assigning customers to the selected processing centers and defining the second-level routes. Each second-level route, starts at a processing center, visits a set of customers, through one or several processing centers, and then returns to the first processing center. We present a mixed integer linear model for the problem and use a Cplex solver to solve small-scale instances. Furthermore, we propose non-trivial extensions of nearest neighbour and insertion approaches. We also develop clustering based approaches that seldom investigated on location routing. Computational experiments are conducted to evaluate and to compare the performances of proposed approaches. The results confirm the effectiveness of clustering approaches.

.

Keywords: Distribution Network, Vehicle Routing, Location-Routing, Pickup and Delivery, Mixed Integer Programming, Heuristics.

1 introduction

One of the most important problems in a supply chain is the design of logistic network. The transportation costs often represent an important part of the logistic network cost and substantial saving can be reached by improving the transportation system. This transportation system design must also cope with the new challenges of sustainable development. For instance, decisions about the number and the localization of facilities (platforms, factories, depots) are among the important strategic decisions to consider in the design of the transportation system.

Many studies in literature considers the facility location problem, in which facilities must be located, considering that customers must be delivered directly from facilities and the vehicle routes are ignored. This may lead to suboptimal solutions. In fact, location and routing are interdependent and must be considered together, Salhi and Rand [32], Prins et al. [23].

The location routing problem (LRP) and its variants are the models of the literature that allows combining the strategic decisions (related to the selection of potential sites) with the tactical and operational decisions (related to the assignment of customers to the selected potential sites and the construction of vehicle routes in order to serve all customer demands). The objective of the LRP is to minimize the total cost including routing costs, vehicle fixed costs, and potential site opening costs.

In the context of distribution system with two levels, the LRP variant is called Two-Echelon LRP (LRP-2E). It arises for example in freight distribution. The distribution system consists of the first-echelon facilities (depots), the second-echelon intermediate facilities (processing centers), and the third level including a set of customers. The LRP-2E aims at locating the subsets of the first and the second echelon facilities, constructing the primary (first echelon vehicle routes) and the secondary routes (second echelon vehicle routes). Each primary route starts from opened depots, visits a subset of opened processing centers and returns to the initial depot. Each secondary route starts from an opened processing center, visits customers once and returns to the initial opened processing center. Two sets of homogenous fleet of vehicles are used, one fleet for each level. The fleet of second echelon uses a smaller vehicle capacity than the first level.

The first studies on the LRP-2E are initiated by Jacobsen and Madsen [33], Madsen [34]. These studies consider real applications in newspaper delivery with 4500 customers. In the newspaper distribution system, newspapers are delivered from the printing factories (depots) to transfer points (processing centers) and from these points to customers. A recent survey on LRP and LRP-2E can be found in Prodhon and Prins [24].

The most studied variant of LRP-2E considers that the distribution system is composed of one of the following variants: (i) more than one depot in the first level and a set of potential processing centers, either without capacity constraints (Lin and Lei [13]), or with capacity constraints (Boccia et al. [2], Contardo et al. [6], Schwengerer et al. [30]), (ii) one fixed depot and a set of potential processing centers with fixed capacity and opening costs (Nguyen et al. [21], Nguyen et al. [38]), (iii) one or more depots and a set of potential processing centers, without capacity constraint and without opening cost, such as the truck and trailer routing problems (Villegas et al. [31]).

Several exact heuristics and metaheurisitcs approaches are proposed in the literature to solve the LRP-2E. In Boccia et al. [3] different mixed integer models of the LRP-2E were proposed. Contardo et al. [6], Rodríguez-Martín et al. [40], Coelho and Laporte [41] investigated a branch and cut approach. The metaheuristic approaches were more considered such as genetic algorithm, (Hamdi-Dhaouia et al. [14], Lin and Lei [13], Karaoglan and Altiparmak [43]), GRASP with path relinking, (Nguyen et al. [21]), multi-start Iterative Local Search (Nguyen et al. [21]), Tabu Search, (Boccia et al. [2]), Variable Neighborhood Search - VNS, (Schwengerer et al. [30]), Tabu Neighborhood Search (Escobar et al.[42]), multi-start Simulated Annealing (Lim [44]). A comparison of the performance of these metaheuristics methods was given in Prodhon and Prins [24].The clustering approach have not been considered for LRP-2E. Only a few studies have developed a such approach for LRP, Ozdamar and Demir [22], Barreto et al., [1], Zare Mehrjerdi and Nadizadeh [32], Guerrero and Prodhon, [10] however several authors have recognized the potential of cluster analysis, such as Bruns and Klose [4], Barreto et al. [1].

Other problems derived from LRP-2E are considered in the literature, such as, (i) Two-Echelon vehicle routing problem (VRP-2E) when there is no fixed cost for using depots and processing centers, (Crainic et al. [7], and Cuda et al. [45]), (ii) Two-Echelon capacitated facility location problem, in which the clients are directly linked to the facility, (Li et al. [12]). The LRP is more investigated than the LRP-2E. Some review of LRP models, approaches and applications could be found in Min et al. [16], Nagy and Salhi [20], Duhamel et al. [9], Derbel et al. [8], Lopesa et al. [15], Prodhon and Prins, [24], Prins et al. [25].

In the literature all papers on LRP-2E have considered a classical VRP constraints within the LRP-2E, i.e., each vehicle starts from a processing center, delivers goods to a number of customers, such that each customer is visited once, and returns to the same processing center. However, in practice, customers can have pickup and delivery demands, they request several products and vehicles can visit one or more processing centers in the same route to refill.

In this paper we propose to study a new location routing problems including a non classical VRP constraints. More precisely, we consider the following constraints: (i) pickup and delivery in the same route, (ii) the use of one or more intermediate processing centers in the same route, and (iii) multi-products demands. This new proposed model is called LRP-MPPD-2E for Two-Echelon Multi-products Location-Routing problem with Pickup and Delivery. To the best of our knowledge, the three constraints listed above were not considered simultaneously in LRP and LRP-2E literature except in Rahmani et al. [26, 27]. Recently, Rieck et al. [28] considers many-to-many LRP with inter-hub transport and multi-commodity pickup and delivery, but the vehicle route may be pure pickup, pure delivery, or mixed, where some pickup locations have to be visited before serving the first delivery location. The intermediate hubs are not considered in the vehicle routes as in our model but direct paths between hubs are considered. The LRP-MPPD-2E allows modeling problems arising in a number of applications like drink distribution, home delivery service and grocery store chains, e.g. Carrera et al. [5]. These applications are characterized by complex transportation network that may include factories, warehouses, customers and suppliers. Consolidating freight through one or more processing centers in the same route, allows considerable savings. The LRP-MPPD-2E is also a general case for several problems, such as:

1. The traveling salesman location problem with pickup and delivery introduced by Mosheiov [18].

2. The LRP-SPD (LRP with Simultaneous Pickup and Delivery), introduced by Karaoglan et al. [35], in which the pickup and delivery are considered simultaneously in the vehicle routes. This vehicle routing problem is known in the literature as vehicle routing problem with simultaneous pickup and delivery - VRPSPD, Berbeglia et al. [36]; Parragh et al. [37].

3. Many to many LRP introduced by Nagy and Salhi [19] in which several customers wish to send goods to others and flows between processing centers are permitted.

4. Multi-commodity pickup and delivery vehicle routing problem, Hernandez-Perez and Salazar-Gonzalez [11]; Rodriguez-Martin and Salazar-Gonzalez [29].

5. Vehicle routing problem with intermediate facilities or with satellite facilities, Moin et al. [17].

In this paper, we extend an initial study presented in Rahmani et al. [26, 27]. We propose to investigate a non trivial extension of classical vehicle routing heuristics, namely Nearest Neighbor Heuristic

(NNH) and Sequential Best Insertion Heuristic (SBIH). Another approach, based on clustering analysis namely Hybrid Clustering Algorithm (HCA) is also proposed.

The rest of this paper is organized as follows. Section 2 presents the considered LRP-MPPD-2E problem and it mathematical model. Section 3 describes the heuristic approaches and four strategies for each heuristic. Experimentation and concluding remarks are discussed in the section 4 and section 5, respectively.

2 Problem Description and mathematical model

In the Two-Echelon Multi-products Location-Routing problem with Pickup and Delivery - LRP-MPPD-2E, two levels are considered. At the first level, routes are constructed from a main depot to a set of active processing centers that must be selected, and at the second level, a set of vehicles of smaller capacity visit customers from the selected processing centers. We denote *primary* and *secondary routes*, the routes constructed at the first and the second level, respectively.

The LRP-MPPD-2E is modeled as an undirected and weighted graph $G = (N, A, l)$. N refers to the set of nodes, where $N = \{0\} \cup N_0 \cup N_c$, in which N_0 represent the sets of potential processing centers nodes, N_c represents the set of customers and node 0 is considered as a depot. A is the set of edges and l refers to a function that associate a positive cost (time) to each arc (typically travel time). At depot there is a set V_1 of homogeneous fleet of *primary* vehicles. Each primary vehicle has a limited capacity C_{V1} and a fixed cost F_{V1}. Another set V_2 of homogeneous fleet of *secondary* vehicles is available at the processing centers sites. Each secondary vehicle has a limited capacity C_{V2} and a fixed cost F_{V2}. We consider the general case when C_{V1} is different from C_{V2}. Each potential processing center has an opening cost.

Each client asks for one or several type of products, denoted c-products, known in advance and could be satisfied. In each processing center, pickup and delivery operations are performed. Primary products, denoted h-products, are delivered from main depot to active processing centers. Each active processing center can receive only one type of h-products. The h-products are transformed into final products, denoted c-products. Each processing center should provide exactly one secondary c-product.

We consider two types of vehicles as explained above. The primary vehicles should pick up the h-products from the main depot and deliver them to the active processing centers, which have been opened, such as each processing center is visited only once in each primary route. When satisfying the demand of processing centers, the secondary vehicles can pickup c-products, which are available in the

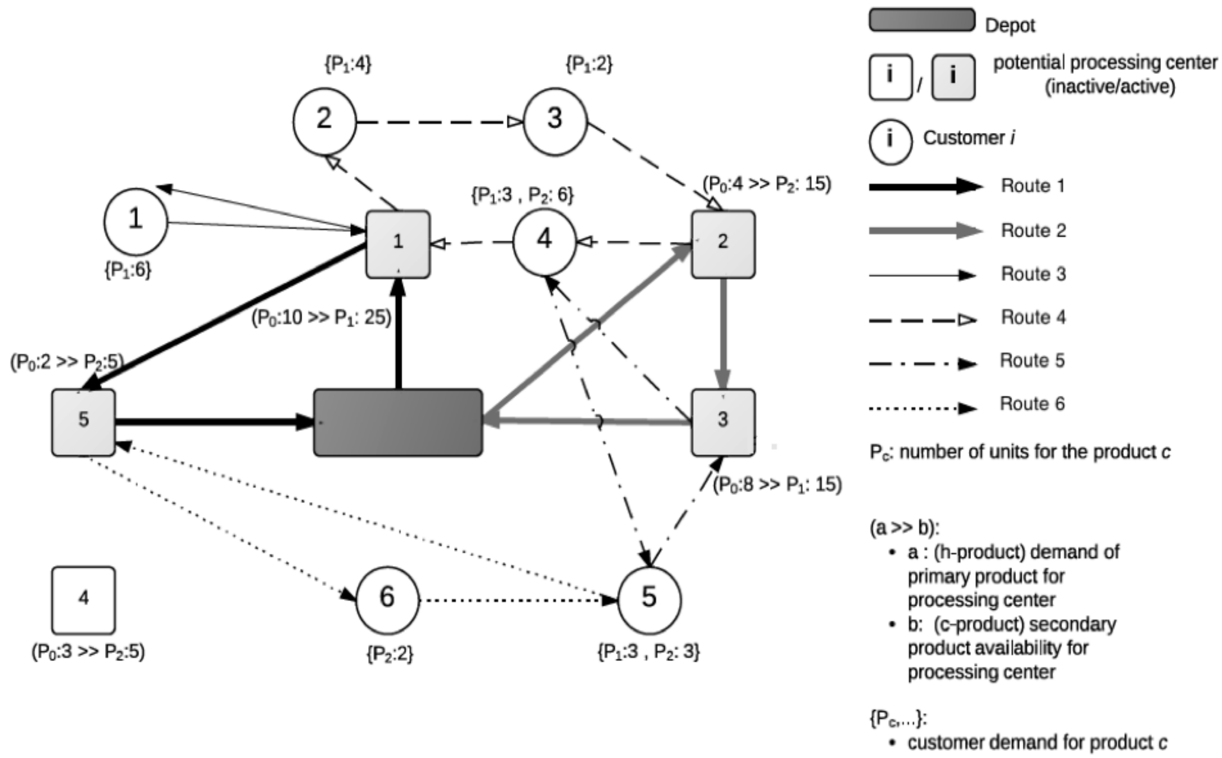

Figure 1: Example of LRP-MPPD-2E with 5 processing centers and 6 clients.

processing centers, and continue their trips in a way that each customer and processing center is visited at most once by each secondary trip. The secondary trips start from an active processing center, which will represent the departure node, serve several customers, can visit one or several processing centers and must end up at the departure node. We assume that products have the same size, the splitting demand of customers for a given c-product is not allowed. The goal of LRP-MPPD-2E is to determine the location of active processing centers, the assignment of customers to the opened processing centers and the construction of the corresponding primary and secondary routes with a minimum total cost. The total cost includes the opening cost of processing centers, the exploitation cost of vehicles and the sum of edges costs traversed by vehicles. An illustrative example of the two-echelon model is given in Figure 1.

The following notations are used in our mathematical model.

$d(i,j)$: the cost of the shortest path in G between vertices i and j.

d_{is}: the demand of client i for c-product s.

p_{ks}: amount of c-product s that could be provided by potential processing center k.

FD_k: a fixed cost associated with each potential active processing center $k \in N_0$.

V_1: a set of identical primary vehicles, available at the main depot.

V_2: a set of identical secondary vehicles, available at the processing center sites.

V: the set of all primary and secondary vehicles.

C_{V1} (C_{V2}): the capacity of primary (secondary) vehicle.

F_{V1} (F_{V2}): the exploitation cost including the acquiring cost of the primary (secondary) vehicles.

d_{ks}: primary product demand of processing center $k \in N_0$ for product $s \in P$.

H_{is}: is equal to 1 if node i asks for product $s \in P$, otherwise is equal to 0.

M_{ks}: is equal to 1 if processing center k produces c-product $s \in P$, otherwise it is equal to 0.

$|P_j|$: the number of c-products types that customer $j \in N_c$ asks for them.

The following decision variables are used in our mathematical model.

$x_{ij}^v = 1$ if the vehicle v travels directly from node i to node j and 0 otherwise.

$y_{ks} = 1$ if supplier k is setup on a site for product $s \in P$ and 0 otherwise

$q_{is}^v = 1$ if vehicle $v \in V$ satisfy demand s of node $i \in N$ and 0 otherwise

l_{is}^v: Total residual load in vehicle v just after having serviced customer (processing center) i for product s.

U_{is}^v: Amount of pickup processing center i for product s by vehicle v.

The problem can be formulated as follow:

$$\min \sum_{i \in N} \sum_{j \in N} \sum_{v \in V} c_{ij} x_{ij}^v + \sum_{k \in N_0} \sum_{s \in P} FD_k y_{ks} + \sum_{k \in K} \sum_{v \in V_1} FV_1 x_{0k}^v + \sum_{v \in V_2} FV_2 h_v \tag{1}$$

$$\sum_{i \in N_0} \sum_{j \in N} x_{ij}^v \geq h_v \qquad \forall v \in V_2 \tag{2}$$

$$\sum_{i \in N_0} \sum_{j \in N} x_{ij}^v \leq M h_v \qquad \forall v \in V_2 \tag{3}$$

$$\sum_{i \in N} \sum_{v \in V_2} x_{ij}^v \leq |P_j| \qquad \forall j \in N_c \tag{4}$$

$$\sum_{i \in N} \sum_{v \in V_2} x_{ij}^v \geq 1 \qquad \forall j \in N_c \tag{5}$$

$$\sum_{i \in N} x_{ij}^v \leq 1 \qquad \forall j \in N, v \in V \tag{6}$$

$$\sum_{v \in V} \sum_{i \in N} x_{ij}^v \geq y_{js} \qquad \forall j \in N_0, s \in P \tag{7}$$

$$\sum_{j \in N} x_{ji}^v - \sum_{j \in N} x_{ij}^v = 0 \qquad \forall i \in N, v \in V \tag{8}$$

$$\sum_{i \in Y} \sum_{j \in Y} x_{ij}^v \leq |Y| - 1, \forall v \in V_1, Y \subset N - \{0\}, |Y| \leq 2 \tag{9}$$

$$\sum_{i \in N_0} x_{0j}^v \leq 1 \qquad \forall v \in V_1 \tag{10}$$

$$\sum_{i \in N_c} x_{i0}^v \leq 1 \qquad \forall v \in V_2 \tag{11}$$

$$\text{if } x_{ij}^v = 1 \forall i \in N_0 + \{0\}, \Rightarrow l_{is}^v - d_{js} q_{js}^v = l_{is}^v \qquad \forall j \in N_0, s \in P, v \in V_1 \tag{12}$$

$$\text{if } x_{ij}^v = 1 \forall i \in N, \Rightarrow l_{is}^v - d_{js} q_{js}^v = l_{is}^v \qquad \forall j \in N_c, s \in P, v \in V_2 \tag{13}$$

$$l_{is}^v - d_{js} + (1 - x_{ij}^v) M \geq (q_{js}^v - 1) M \forall i \in N + \{0\}, s \in P, j \in N, v \in V \tag{14}$$

$$\sum_{v \in V} U_{is}^v \leq p_{is} y_{is} \qquad \forall i \in N_0, s \in P \tag{15}$$

$$\sum_{s \in P} y_{ks} \leq 1 \qquad \forall k \in N_0 \tag{16}$$

$$\sum_{s \in p} l_{is}^v \leq C_{V1} \qquad \forall i \in N_0, v \in V \tag{17}$$

$$\sum_{s \in p} l_{is}^v \leq C_{V2} \qquad \forall i \in N_0, v \in V \tag{18}$$

$$l_{0s}^v = \sum_{k \in N_0} q_{ks}^v d_{ks} \qquad \forall s \in P, v \in V_1 \tag{19}$$

$$\sum_{b \in P} \sum_{v \in V_1} q_{kb}^v = \sum_{s \in P} y_{ks} \qquad \forall k \in N_0,\ H_{kb} = 1 \tag{20}$$

$$\sum_{v \in V} \sum_{s \in P} q_{is}^v = \mid P_j \mid \qquad \forall j \in N_c \tag{21}$$

$$\sum_{i \in N} \sum_{s \in P} q_{is}^v \geq \sum_{j \in N_c} x_{0j}^v \qquad \forall v \in V_1 \tag{22}$$

$$\sum_{k \in N_0} \sum_{i \in N_c} d_{is} z_{ik} \leq \sum_{k \in N_0} p_{ks} y_{ks} \qquad \forall s \in P \tag{23}$$

$$\sum_{i \in N_0} \sum_{s \in P} q_{is}^v \leq M \sum_{j \in N_0} x_{0j}^v \qquad \forall v \in V_1, i \in N \tag{24}$$

$$\sum_{s \in P} q_{is}^v \leq 2 \sum x_{ij}^v \qquad \forall v \in V_1 \tag{25}$$

$$\mu_{iv} - \mu_{jv} + 1 \leq (\mid N \mid -1)(1 - x_{ij}^v) \qquad \text{j} \in N, v \in V_2, i, j \neq 1 \tag{26}$$

$$2 \leq \mu_{iv} \leq \mid N \mid \qquad \forall i \in \{2, ..., \mid N \mid\} \tag{27}$$

$$x_{ij}^v \in \{0, 1\} \qquad \forall i, j \in N, v \in V \tag{28}$$

$$h_v \in \{0, 1\} \qquad \forall v \in V_1 \tag{29}$$

$$y_{ks} \in \{0, 1\} \qquad \forall s \in P, k \in N_0 \tag{30}$$

$$q_{is}^v \in \{0, 1\} \qquad \forall s \in P, k \in N_0, v \in V \tag{31}$$

The objective function (1) minimizes the total travel costs, the opening cost of the selected processing center's and the vehicle fixed costs. The constraints (2) and (3) ensure that a secondary vehicle starts through an active processing center. The constraints (4) and (5) guarantee that the number of vehicles that pass through each customer is at most equal to the number of c-products required by the customer and at least equal to one. The constraint (6) ensures that a vehicle could visit each node maximum once. The constraint (7) ensures that when a processing center is opened, there is at least one visit to this processing center. The constraint (8) is known as degree constraint and guarantees that the number of entering and leaving arcs to each node are equal. (9) is a constraint of sub tour-elimination for the primary vehicles. The constraints (10) and (11) assume that once a secondary vehicle is being exploited, it starts its route from a processing center and ends up to the same processing center. The constraints (12)-(15) assure the compatibility between routes and vehicle capacity. The constraint (16) ensures that each opened processing center produces only one type of c-product. The constraint (17) and (18) assure that the total load on each processing center does not exceed the primary and the secondary vehicles capacity. The Constraint (19) guarantees that the total load in vehicle at depot is equal to the requested quantity of all h-products by the processing centers, which will be satisfied by the given vehicle. The constraints (20)-(25) handle the satisfaction of demands at each node. In order to eliminate the sub-tours of secondary level, an alternative formulation with polynomial size introduced by Miller, Tucker, and Zemlin [39] is used in constraints (26) and (27). (28)-(31) are known as integrity constraints.

3 Solution methodology

As the *LRP-MPP-2E* is NP-hard problem, and since it results from the combination of complex constraints, large instances can hardly be solved by exact methods. Then the best way to tackle this problem is using heuristics approaches. In this section, we investigated a non-trivial extension of classical vehicle routing heuristics, namely, the Nearest Neighbour Heuristic (NNH), and a Sequential Best Insertion Heuristic (SBIH). Another approach, based on clustering analysis namely Hybrid Clustering Algorithm (HCA) is also proposed. In the following we provide details of each heuristic.

3.1 The Nearest Neighbour Heuristic (NNH)

The Nearest Neighbour Heuristic (NNH) consists of two steps. The first step aims of locating processing centers and provides the first-level routes. The goal of the second step is to use the selected processing centers of the first step to construct the second level tours. Both steps are detailed in the following.

Step 1. **First-level routes construction**. The first level of model involves location and routing phases where primary vehicles, pass through potential processing center by activating them and satisfying their demands of $h-$products. Each route starts and ends at the depot. This step consists on two phases. In the first phase, a processing center k^* is selected to start the route using the *best score* calculated by formula (32) or by using a random selection.

$$k^* = \text{argmin}_{k \in N_0} \left\{ d(k,0) + \frac{p_{ks}}{FD_k} + \sum_{i \in Z} d(k,i) \right\} \tag{32}$$

Where Z in (32) is the set of non-satisfied clients that request c-product of processing center k. The nearest neighbour strategy is used in the second phase to complete the routes. More precisely, a neighbour inactive processing center k that provides $c-$product s is inserted in the route if (i) there is enough amount of $h-$product to satisfy the demand of processing center k, (ii) route duration doesn't exceed the time limit, and (iii) the sum of the $c-$product s available in the processing center already opened is less than the total customers demand of product s. Otherwise a second neighbour candidate (inactive processing center) will be checked, until no processing center can be inserted in the current tour. In that case, a new tour is created.

Step 2. **Second-level routes construction.** In this level, the secondary vehicles try to fulfill the customers' requests. Figure. 2 shows the approach process of this level. In this step, we ignore all inactive processing centers that have not been opened in step 1.

Firstly, a processing center k^* is selected as depot center either randomly or according to the score calculated by using formula (33) to start the tour.

$$k^* = \text{argmin2}_{k \in N_0} \left\{ \sum_{i \in Z} d(k,i) \right\} \tag{33}$$

Where Z in (33) is the set of non-satisfied clients that request c-product of processing center k. Then, the nearest neighbour strategy is used to construct the tours. This process is repeated until all client requests are satisfied. Note that a processing center is chosen once as a depot-center, but can be visited by several tours for a pickup of its $c-$product, then within a current tour neighbour candidate (active processing center or client) is inserted if all constraints represented are satisfied otherwise a second neighbour candidate will be checked, until neither processing center nor client can be inserted in the tour (Figure. 2).

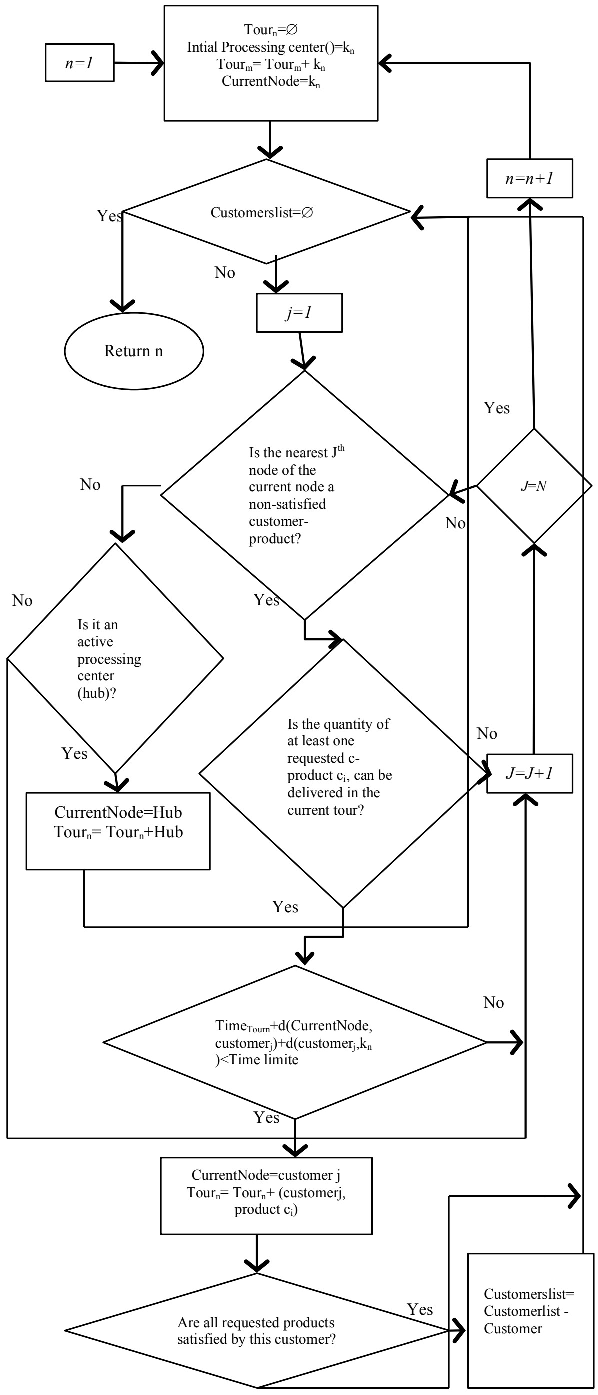

Figure 2: Secondary Level Routing of NNH Algorithm

3.2 Best Sequential Insertion Heuristic (BSIH)

In this section we describe the insertion heuristic method, developed to solve the LRP-MPPD-2E. The heuristic method consists in two phases: (i) initialization phase and (ii) insertion phase. The initialization phase determines the first processing center in each new route, and the second phase, tries to insert the maximum number of clients and processing centers into the current route. The two phases are repeated until the total customers demand is satisfied. The routing between the selected processing centers for the first level (first-level routes) is obtained by a vehicle routing nearest neighbour heuristic described in section 3.1, step 1.

Step 1. **Initialization Phase.** The goal of this step is to determine, for each $c-$product, the processing center that will start a new route. In order to select the best processing center, we use the score value of formula (32) for each $c-$product. The score function takes into account the distance between the processing center and the depot, the sum of distances between a processing center and non-satisfied clients that request a $c-$product, and the opening cost of the processing center. A processing center with minimal score is selected to start the route. The Initialization function will check for each product, whether the already opened processing centers of the relevant product could satisfy the clients, who asked the mentioned given products.

Step 2. **Route insertion phase.** After inserting a processing center via the initialization step, the construction of a route is realized by two processes 2.1 and 2.2 described below. These two processes will be repeated until the time limit of the route is reached. In this case if all clients requests are satisfied, the algorithm stops; otherwise, a new route is initialized by selecting a processing center according to Step 1.

2.1. *Client insertion.* Let $(i, \{s_1, \ldots, s_n\})$ be a pair of a customer i, and its request of n $c-$products $\{s_1, \ldots, s_n\}$ and let LC be the list of pairs $(i, \{s_1, \ldots, s_n\})$ of non satisfied clients-products sorted according to one of this two strategies: (i) Random Client Insertion (RCI) in which the customers are sorted randomly, and (ii) Nearest Neighbour Client Insertion (NNCI) in which customers are sorted according to their proximity to the route under construction.

In the client insertion process, the request of client i can be splitted into (i, S_1) and (i, S_2) such that $S_1, S_2 \subseteq \{s_1, \ldots, s_n\}$ and $S_1 \cup S_2 = \{s_1, \ldots, s_n\}$. The client insertion process can be described as bellow:

Select a client i from the head of LC. The algorithm inserts the client (i, S) at the best position k^* that maximize the sum of demands of client i over the insertion cost Δ. More precisely, if SP is the subset of $c-$products requested by client i that may be inserted at position k between two already inserted node a and b, in L, where L is the subset of all feasible positions in current route, then

$$k^* = \text{argmax}_{k \in L} \left\{ \frac{\sum_{s \in SP} d_{is}}{\Delta} \right\} \tag{34}$$

Where $\Delta = d(a,k) + d(k,b) - d(a,b)$.

If the total demand of client i is satisfied in position k^*, then the pair (i, S) is removed from LC, otherwise all products already satisfied are removed from subset S and list LC is updated. If LC is empty then all demands are satisfied and the algorithm stops, otherwise we scan the list LC to insert the rest of clients, once all clients are tested the algorithm continues with the processing center insertion process.

2.2 *Processing center insertion.* All available processing centers are candidate for insertion. A processing center pc is chosen by its closeness to the current route. Two insertion strategies are used BIH_1 and BIH_2. In BIH_1, the goal is to shift the insertion of pc at the last position when the current route time is so far from time limit of the route. Let T be the duration time of the current route and HP a subset of clients in the current route that could be satisfied by given processing center pc. If $T \leq \textit{time limit} \times \alpha$ where $\alpha \in [0, 1]$, then pc is inserted at the last feasible position, otherwise all positions are checked to compute the best insertion position l^* according to (35). In BIH_2 only the best insertion position rule is used where the best position l^* is calculated according to (35).

$$l^* = \text{argmax}_{l \in L} \left\{ \frac{\sum_{i \in HP} d_i s}{\Delta} \right\} \tag{35}$$

If a processing center is inserted then the algorithm continues with the client insertion procedure, otherwise when there is no processing center to be inserted into the current route, then the current route is closed, and the algorithm restarts with *the initialization phase.*

3.3 Hybrid Clustering Algorithm (HCA)

The proposed HCA algorithm is a non-trivial extension of a greedy clustering method proposed in [32] for a classical LRP with a fuzzy demands. The HCA algorithm proceeds in five steps (see Figure. 3). In the first step, customers are clustered using an algorithm based on nearest neighbour, such that each cluster should involve only clients that request the same product (Figure. 3.a). In the second step, the gravity center of each cluster is calculated. This allows to select a set of potential processing centers (Figure. 3.b). In the third step, clusters are merged as well as possible in order to create the Global-Clusters (GC) in which only one vehicle will be exploited, i.e., each Global-Cluster represents one feasible secondary route. This merging step considers the distance between the gravity center of the clusters as well as the route time limit (Figure. 3.c). In order to ensure the feasibility of the solution in each Global-Cluster, the merged clusters should not have any common client, since the exploited vehicle for each Global-Cluster must visit only once each customer and each processing center. The clusters are allocated to the opened processing centers in the forth step, considering the distance between the processing centers and the gravity center of the clusters as well as the capacity of the processing centers (Figure. 3.d). Finally, in the fifth step, Cplex solver is used to find a feasible secondary route in each Global-Cluster (Figure. 3.e). The routing between the selected processing centers in the first level (primary tours) is obtained by a vehicle routing nearest neighbour heuristic. Details of the HCA steps are given bellow.

1. **Clustering the customers.** The customers are separated into different groups considering their intra distance, the sum of their demands, the vehicle capacity, the time route limit, and an estimation of the route travel time given in formula (36) in which N^c_{cl} and N^0_{cl} present the number of clients and processing centers in cluster cl, respectively. D^{Max}_C is the maximal distance between each pair of clients in cluster cl. The maximal distance between the processing centers and the clients in cluster cl is denoted by D^{Max}_{PC}.

$$T = ((N^c_{cl} - N^0_{cl}) \times D^{Max}_C) + 2 \times (N^0_{cl} \times D^{Max}_{PC}) \tag{36}$$

The value T associated to a cluster cl, is an overestimation of route that start from a processing center, visiting all the customers assigned to the cluster cl, and ending at the starting processing center. More precisely, for each $c-$product p, a set of non-clustered customers (NCC_p) is initialized by all customers j such as $d_{jp} > 0$, where d_{jp} is the quantity of product p asked by the customer j. At first, a customer is selected randomly from a set NCC_p, then the nearest customer

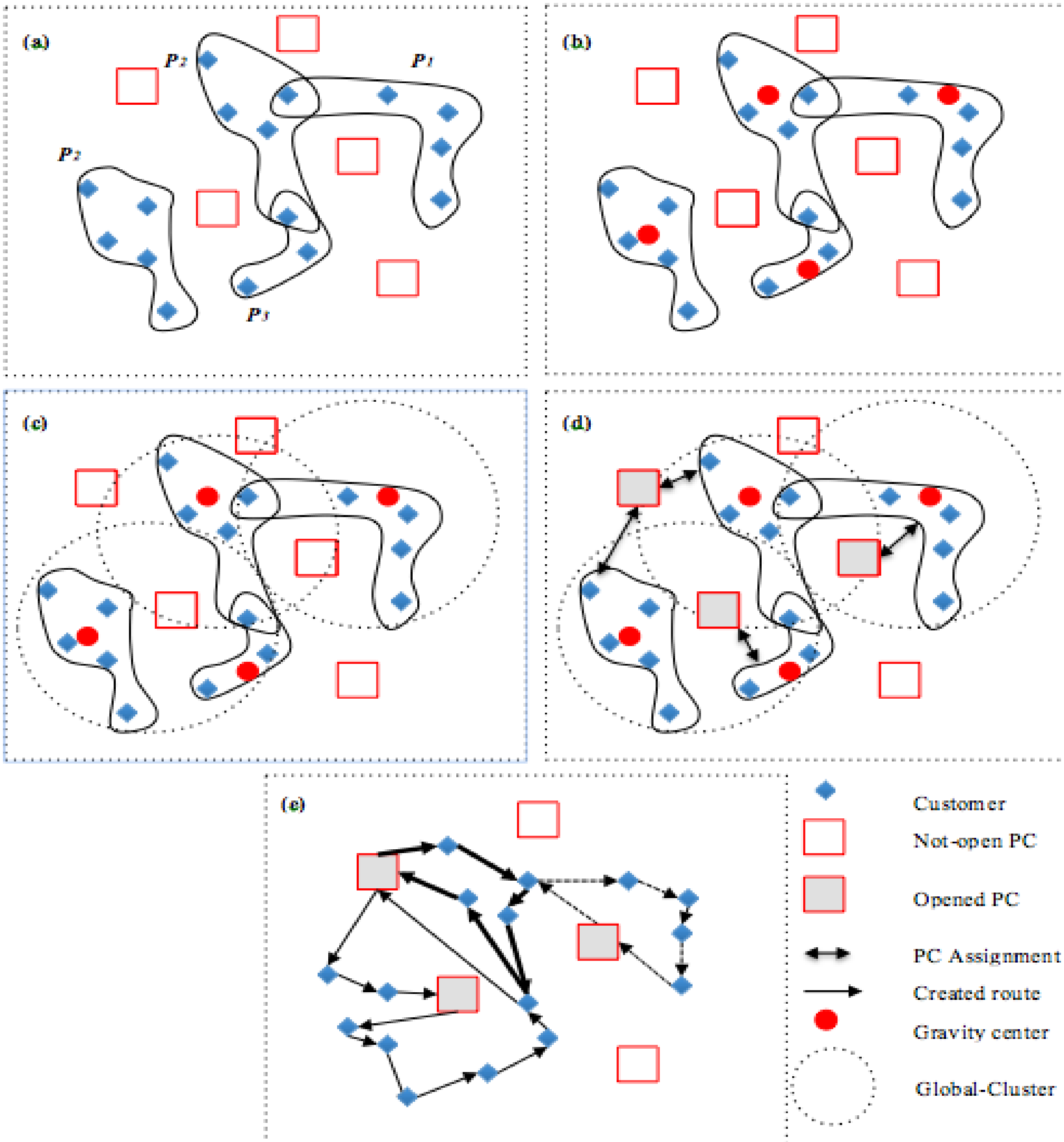

Figure 3: HCA for the LRP-MPPD-2E.

to the last selected customer of the current cluster is chosen from NCC_p. Therefore the clusters are formed for a single $c-$product. The nearest customer is selected as follow: when a new customer j asking for product p is selected from NCC_p set (j is the closest customer regarding the distance to the last inserted customer in the cluster cl), before its assignment to cl, in order to limit the size of each cluster, we verify two conditions, (i) the sum of the amounts requested by the assigned clients to the cluster cl should not exceed the secondary vehicle capacity C_{V2}, (ii) the estimated travel time in a cluster cl, i.e., value T in (36) with $N^0_{cl} = 1$, doesn't exceed the time route limit. If these two conditions are fulfilled, the new customer is assigned to the current cluster, otherwise, the algorithm searches in NCC_p for the next closest customer to the last added customer. The algorithm creates a new cluster if there is no customer to be assigned to the current cluster. The algorithm stops when there is no unassigned customer. Figure (4) illustrates the cluster's selection algorithm.

2. **Processing Center (PC) selection.** In the second step of the HCA, the method of [32] to establish the list of opened processing centers is used. This method is based on a gravity center criterion as illustrated by equation (37), in which (X_{cl}, Y_{cl}) is the coordinates of the gravity center of the cluster cl and (x_i, y_i) is the coordinates of customer i, where n_{cl} is the number of customers assigned to cluster cl.

$$(X_{cl}, Y_{cl}) = \left(\frac{\sum_{i \in cl} x_i}{n_{cl}}, \frac{\sum_{i \in cl} y_i}{n_{cl}} \right) \tag{37}$$

For each processing center, we calculate the sum of distances between this potential site and all gravity centers. The potential sites are re-indexed in non-decreasing order of their Euclidean distance to the gravity center of the clusters. If the current opened top-ranked potential site is not able to fulfill all the remaining customers' demands, the next potential site of the sorted list is selected to be opened. This procedure is repeated until all the clusters are covered. Therefore, each selected processing center will be assigned to one or more cluster and each cluster is covered by one or more processing centers.

3. **Merging the clusters into Global-Cluster.** In this step, the clusters are merged in order to create a set of Global-Clusters, GC, in which GC represents one feasible secondary route. Since the assigned vehicle to each GC must visit customers and processing centers only once, then the merged clusters should not have any common client.

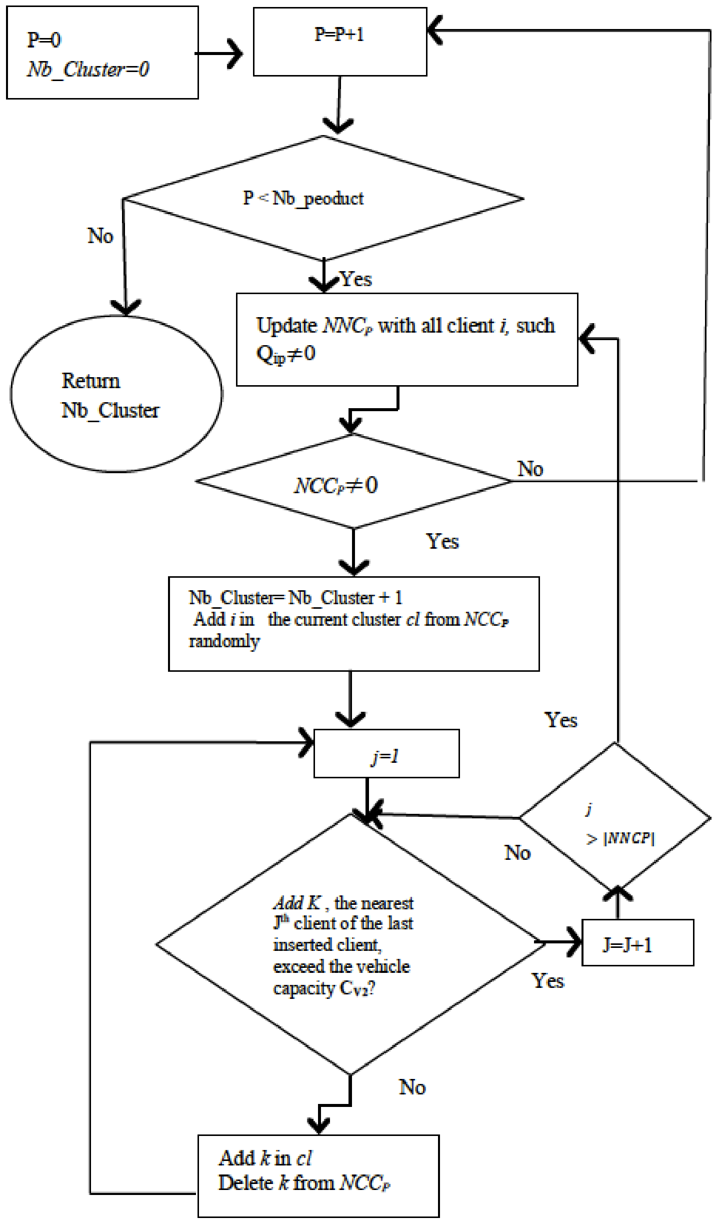

Figure 4: Clustering Customers Algorithm.

At first, a cluster cl is selected randomly, and then a sorted list of the not merged clusters cl' is constructed according to the distance between the gravity centers of cl and cl'. The first cluster in the list is added into the current Global-Cluster GC if the value of T calculated by (36) with N^0_{cl} equal to the number of merged clusters in GC did not exceed the time route limit (Figure. 3.c). This procedure is repeated until no cluster can be added to the current Global-Cluster. In that case, either the process stops because all the clusters are merged or the process is restarted to search for a new Global-Cluster.

4. **Assigning clusters to Processing Centers.** In the forth step of the HCA, the clusters are

allocated to the processing center that were ranked and opened in the processing center selection step. Each processing center serves as many clusters as possible according to its capacity. Note that we can't allocate two clusters cl_1, cl_2 to the same processing center when they were merged in the same GC. Because a vehicle cannot visit a processing center twice in a given route. In order to allocate the clusters to the processing center, the Euclidian distance between the gravity center of each cluster and the opened processing center is calculated. Then the unassigned clusters are ranked in an ascending order based upon the distance of their gravity centers to the processing center. The top-ranked cluster cl_1 will be allocated to the top-ranked processing center p_r if (i) the processing center p_r has enough capacity to cover the total demands of the cluster cl_1, and (ii) the processing center p_r is not already affected to a cluster cl_2, such as cl_1 and cl_2 belong to the same Global-Cluster. The allocation process to the processing center p_r is completed when there is not enough capacity to allocate a new cluster. In that case, the allocation procedure is repeated for the next top-ranked processing center until all clusters are allocated.

5. **Routing problem.** In the fifth and last phase of the HCA, the routing problem is solved for each Global-Cluster GC with the relevant processing centers and assigned clients. Each Global-Cluster is served by exactly one vehicle and the vehicle is not allowed to visit any node two times. Cplex solver is used to create one secondary route per one Global-Cluster.

4 Computational experiments

4.1 Implementation and Benchmark instances

The proposed heuristics were computationally compared within the exact approach. Results are evaluated in terms of quality and running-time. All the experiments were carried out on a PC with Intel (R) Core (TM) Solo CPU 1.40 GHz, 2GB of RAM. The tested algorithms were coded in C++ and the routing steps of HCA and the mixed integer linear model use a Cplex solver version 14.0.

Prodhon et al. [14] have proposed a set of *LRP-2E* Euclidean instances. Prodhon's *LRP-2E* benchmark contains 30 instances, derived from Prodhon's *CLRP* benchmark by converting depots into satellites-depots, and adding one main depot at the origin $(0, 0)$ point. These instances are grouped in four subsets with the following features: number of customers $n \in \{20, 50, 100, 200\}$, uniform distribution demands in interval $[11, 20]$, number of satellites-depots $m \in \{5, 10\}$, with their opening costs, number of clusters $\beta = \{1, 2, 3\}$ of clients (manner of customers distribution), vehicle capacity of second level C_{V2}, vehicle capacity of first level C_{V1}. Since there is some simillar instances with just different vehicle capacity in Prodhon's benchmark, we obtained only 18 instances among 30 instances of Prodhon's Benchmark by fixing capacity $C_{V1} = 200$ and $C_{V2} = 70$. Each instance is described by three parameters: $n - m - \beta$.

In order to adapt these instances to our problem, we have considered the following hypotheses:

- Each satellite-depot corresponds to a processing center in our problem.
- We consider 3-district products: one $h-$product p_0 and two others $c-$products p_1 and p_2.
- Each client asks for products, p_1 or p_2 or both products with equal probability.
- We consider two homogeneous fleets of vehicles, with capacity C_{V1} and C_{V2}. We note that C_{V1} must be greater than quantity of all $h-$product demands.
- We added the $h-$product demand for each processing center, such as the demand of each $h-$product is equal to $\frac{1}{5}$ of $c-$product availability in this processing center.

These instances are used to generate another set of 18 instances having the same characteristics as the first set of instances except the opening cost of their processing center. The imposed changes are such that all processing center with the same secondary product p_c get a unique opening cost equal to the minimum cost of the considered processing center. Table 1. provides the characteristics of instances. Vector p_0 represents the $h-$product demands of all processing centers and vectors p_1 and p_2 give indices of processing centers in which the products p_1 and p_2 are available.

4.2 Comparative analysis

In this section, we analyze the performance of the proposed methods using four strategies for each method. This makes 12 heuristics tested and compared for LRP-MPPD-2E. These heuristics are evaluated by using a lower bound LB obtained by Cplex with the relaxation of sub-tours elimination and secondary vehicles capacity constraints. In Tables 2 - 3, the values in parenthesis represents the Gap between the lower bound LB and the solution obtained by the heuristic. The Gap is calculated as $\frac{Cost(H)-LB}{Cost(H)}$.

Results of Tables 2 - 3 show that when $n \geq 200$ for the first set of instances and when $n \geq 100$ for the second subset, it is not possible to evaluate the performance of the methods, since Cplex cannot generates any solution after 1 hour of execution time.

Instance	#C	# hub	$c-$products (p_1), (p_2)	$h-$products (p_0)
coord20-5-1-2e	20	5	p_1(2,4,5), p_2(1,3)	p_0(28,28,28,28,28)
coord20-5-1b-2e	20	5	p_1(2,4,5), p_2(1,3)	p_0(60,60,60,60,60)
coord20-5-2-2e	20	5	p_1(2,4), p_2(1,3,5)	p_0(14,28,14,28,28)
coord20-5-2b-2e	20	5	p_1(3,5), p_2(1,2,4)	p_0(30,30,60,60,30)
coord50-5-1-2e	50	5	p_1(1,4), p_2(2,3,5)	p_0 (84,84,70,70,70)
coord50-5-2-2e	50	5	p_1(1,4), p_2(2,3,5)	p_0(70,70,70,70,70)
coord50-5-2bBIS-2e	50	5	p_1(1,3,4), p_2(2,5)	p_0(60,60,60,60,60)
coord50-5-2BIS-2e	50	5	p_1(1,3,5), p_2(2,4)	p_0(70,70,70,70,70)
coord50-5-3-2e	50	5	p_1(1,4), p_2(2,3,5)	p_0(70,84,70,84,84)
coord100-5-1-2e	100	5	p_1(2,4,5), p_2(1,3)	p_0(154,140,154,154,154)
coord100-5-2-2e	100	5	p_1(2,4,5), p_2(1,3)	p_0(140,154,154,154,168)
coord100-5-3-2e	100	5	p_1(2,4,5), p_2(1,3)	p_0(154,168,154,154,154)
coord100-10-1-2e	100	10	p_1(2,4,5,6,8), p_2(1,3,7,9,10)	p_0(98,84,84,98,112,98,98,98,84,112)
coord100-10-2-2e	100	10	p_1(2,4,5,6,8), p_2(1,3,7,9,10)	p_0(98,112,112,98,98,112,112,98,84,98)
coord100-10-3-2e	100	10	$p1$(2,4,5,6,8), p_2(1,3,7,9,10)	p_0(98,84,98,112,112,112,84,112,98,98)
coord200-10-1-2e	200	10	p_1(4,6,8,9), p_2(1,2,3,5,7,10)	p_0(200,182,182,200,196,200,200,200,200,200)
coord200-10-2-2e	200	10	p_1(4,6,8,9), p_2(1,2,3,5,7,10)	p_0(182,182,196,200,196,196,299,196,196,182)
coord200-10-3-2e	200	10	p_1(1,4,6,8,9), p_2(2,3,5,7,10)	p_0(196,182,200,200,200,196,196,200,200,200)

Table 1: Characteristic of instances

In Tables 2 - 6, the first column indicates the problem name "$coord.n - m - \beta - 2e$" and the last row indicates for each column: (i) the solution average compared to LB, and (ii) the average of running time.

Table 2 gives the results of *NNH* Algorithm with four strategies compared to LB. In columns 2, 3, 4 and 5, we consider that the processing center is chosen according to the nearest neighbor procedure and in columns 6, 7, 8 and 9, we consider the best score as the criterion for opening processing centers. For both previous cases, we present the result in the case where the first processing center is selected randomly (columns 3 and 7) and in the case where the first processing center is chosen according to the score criterion (columns 5 and 9). The column *Time* and *Cost* represent the computation time (seconds) and the obtained value of the total cost, respectively. Results of table 2 show that *NNH* can reach a solution on average during 4 seconds. The minimum and maximum average *Gap* between the *NNH* solution and LB are 16.08% and 23.35%, respectively. Furthermore, results of Table 2 show that the strategy of choosing processing centers according to the score value (SBPCS-SI) is better on 11 instances out of 18 of the first set of instances and 8 instances out of 18 of the second set of instances. Since all processing centers in second subset of benchmark -with the same c-product have a unique opening cost, the dominance of the score based processing centers strategy is less evident in the second set of instances case. The score based processing center strategy is more efficient to solve the first subset of benchmark. We can also note that when the score based processing centers strategy is not used (column 3 and 5), Scored Initialization strategy is not too useful.

In Table 3, the results of the BIH_1 strategy is reported in columns 2, 3, 4 and 5 and those of BIH_2 strategy are gathered in columns 6, 7, 8 and 9. For each strategy, we test two rules for selecting clients, namely, the Nearest Neighbour Client Insertion (NNCI) rule and Random Client Insertion (RCI). The results of *NNCI* are presented in columns 2, 3, 6 and 7 and those of *RCI* in columns 4, 5, 8 and 9. Columns *Time* and *Cost* present the computation time (seconds) and the obtained value of the total cost, respectively. From Table 3, the best results are obtained by the strategy $BIH_1 - NNCI$ (31 best solutions among 36 instances). The maximum and minimum gap between the obtained solutions by this strategy and LB are 48.7% and 4.2%, respectively and the average running time is 10 seconds. BIH_2 does not obtain any best solutions. This issue confirms use that using a best insertion cost, as the unique criterion for processing center insertion is not enough. The strategy BIH_1 avoids unprofitable insertion positions. Several tests were performed to fix the value of α used in $BSIH$ (step 2 of section 3.2), the best results were obtained with $\alpha = 0.9$.

Table 4 gathers the results of four strategies of clustering approaches *C-NNH, HCA-MA, HCA-AM, HCA-CH.* In *C-NNH*, firstly *NNH* is applied in order to create the routes, then all nodes in each secondary route is considered as a Global-Cluster and Cplex solver is used for each Global-Cluster to create the secondary routes. In *HCA-MA* method (HCA with Merge First and Assignment second), the step of merging clusters into Global-Cluster is done before the clusters assignment to processing centers. These two steps are reversed in *HCA-AM* (HCA with Assignment first and Merge second). In *HCA-CH* method, the processing centers is included in the construction process of clusters. In this case, the customers clustering step starts with a processing center according to the score value.

In Table 4, columns 2, 3, 4 and 5 provide the results of *C-NNH, HCA-MA, HCA-AM*, and *HCA-CH* methods, respectively. We limited the computation time to one hour, however we noticed that the solution is obtained on average after only 10 minutes.

The minimum and maximum *Gap* between the *C-NNH* solution and *LB* solution are 3% and 46.1%, respectively. These values are equal to 1.7% and 43.7% for *HCA-MA*, to 1.7% and 45.1% for *HCA-AM*, 0.5% and 43.8% for *HCA-CH*. The *HCA-MA* method obtains 19 best results versus 2 for *HCA-AM*, 9 for *HCA-CH*, and 0 time for *C-NNH*. Note that *HCA-AM* gives a weak result in comparision with *HCA-MA* because when we fix the assignment before merging, the algorithm fails to form good global clusters. *C-NNH* gives bad results for all instances compared to HCA. The use of the gravity center distance as criterion to form mono-product clusters provides better results than the use of a nearest neighbour approach. Note that the gap of solution for second set of instances are similar to the first set. This issue shows that increasing similarity in opening cost of the processing centers does not influence on the results.

Table 5 summarizes the results of all proposed methods. The dominant strategy for each method has been reported. The results of the best insertion strategy corresponding to $BIH_1 - NNCI$ are represented in columns 2 and 3. The results of the best strategy of *NNH* corresponding to score based processing center selection and scored initialization of Table 2 are reported in columns 4 and 5. The results of the best strategy of Clustering method are gathered in columns 6 that refers to *HCA-MA* method. The value in parenthesis for each strategy represents the gap with the best obtained solution.

The results of Table 5 confirm that the clustering approach is more competitive, it finds all the best results for $n < 200$. For the large instances, i.e., $n \geq 200$, the clustering methods fail to provide best results and it is not able to provide any solutions during 10mn. In this case, the insertion method is more competitive.

Table 6, is similar to Table 5 except that the reported results for each method are the best solution found by all tested strategies. The results of table 6 confirm that the clustering approach is the best approach when $n < 200$ and the insertion approach are more efficient for large instances.

5 Conclusion

This paper addresses a new extension of two-echelon location routing problem including new constraints, that have not been considered simultaneously in the literature, namely, multi-product, pickup and delivery and the use of the processing center as intermediate facility in the second-level routes. This new variant, named LRP-MPPD-2E (The Two-Echelon Multi-products Location-Routing problem with Pickup and Delivery), has many realistic applications in supply chain. We have proposed a mixed integer linear model for the problem and use Cplex Solver to solve small-scale instances. Furthermore, we have proposed non-trivial extension of nearest neighbour and insertion approaches. We have developed a clustering based approaches that seldom investigated on location routing. Four strategies were tested for each method. These methods are tested on instances, derived from LRP instances with up to 200 customers and 10 processing centers. An extensive computational experiments shows that the clustering approach is very competitive, it outperforms the other heuristics when $n < 200$. For large instances, the insertion approach is more efficient.

In further researches, we aim to improve HCA with metaheuristic techniques and an iterative process. For instance, we can use a metaheuristic instead of Cplex to solve the routing problem and restart HCA several times with a different initial solutions. It would be also interesting to develop more efficient lower bound. Another perspective is to include more real-life constraints to deal with some other realistic constraints such as splitting of demand per product, the possibility to provide several types of $h-$products per processing center.

References

[1] Barreto, S., Ferreira, C., Paixao, J., and Sousa Santos, B. Using clustering analysis in a capacitated location-routing problem. European Journal of Operational Research 179, 968-977, 2007.

[2] Boccia, M., Crainic, T., Sforza, A., and Sterle, C. A metaheuristic for a two echelon location-routing problem. In P. Festa (Ed.), Symposium on Experimental Algorithms - SEA 2010. Lecture Notes in Computer Science, Vol. 6049, pp. 288-301, 2010.

[3] Boccia, M., Crainic, T., Sforza, A., and Sterle, C. Location-routing models for designing a two-echelon freight distribution system. Tech. Rep., CIRRELT-2011-06, Université de Montreal, 2011.

[4] Bruns, A., and Klose, A. An iterative heuristic for location- routing problems based on clustering. In: Proceedings of the Second International Workshop on Distribution Logistics, The Netherlands, pp. 1-6, 1995.

[5] Carrera, S., Portmann, M.C., and Ramdane Cherif, W. Scheduling problems for logistic platforms with fixed staircase component arrivals and various deliveries hypotheses. Lecture Notes in Management Science, Proceedings of the 2nd International Conference on Applied Operational Research - ICAOR, Turku, Finland, pages 517-528, 2010.

[6] Contardo, C., Hemmelmayr, V., and Crainic, T. G. Lower and upper bounds for the two-echelon capacitated location-routing problem. Computers and Operations Research, 39(12), 3185-3199, 2012.

[7] Crainic, T.G., Mancini, S., Perboli, G. and Tadei, R. Heuristics for the two-echelon capacitated vehicle routing problem. WP. CIRRELT-2008-46, CIRRELT, Université de Montréal, Canada, 2008.

[8] Derbel, H., Jarboui, B., Hanafi, S., and Chabchoub, H. Genetic algorithm with iterated local search for solving a location routing problem. Expert Systems with Applications, Vol. 39, Issue 3, pp 2865-2871, 2012.

[9] Duhamel, C., Lacomme, P., Prins, C., and Prodhon, C. A GRASP ELS approach for the capacitated location routing problem. Computers Operations Research. 37(11), 1912-1923, 2010.

[10] Guerrero, W.J., and Prodhon, C. Capacitated hierarchical clustering heuristic for multi depot location-routing problems. International Journal of Logistics Research and Applications: A Leading Journal of Supply Chain Management, Volume 16, Issue 5, pp 433-444, 2013.

[11] Hernandez-Perez, H., and Salazar-Gonzalez, J.-J. The multi-commodity one-to- one pickup-and-delivery traveling salesman problem. European Journal of Operational Research, 196, 987995, 2009.

[12] Li, J. Prins, C. and Chu, F. A scatter search for a multi-type transshipment point location problem with multi-commodity flow. Journal of Intelligent Manufacturing. 23(4), pp 1103-1117, 2012.

[13] Lin, J.-R., Lei, H.-C. Distribution systems design with two-level routing considerations. Annals of Operations Research, 172, 329-347, 2009.

[14] Hamdi-Dhaouia, K. Labadieb, N. Yalaouib, A. 2013. The bi-objective two-dimensional loading vehicle routing problem with partial conflicts, International Journal of Production Research.

[15] Lopesa, R. B. , Ferreira, C., Beatriz Sousa Santo., B.S., and Barreto., S. A taxonomical analysis, current methods and objectives on location routing problems. International Transactions in Operational Research 20(6), 795-822, 2013.

[16] Min, H., Jayaraman, V., and Srivastava, R. Combined location-routing problems: A synthesis and future research directions. European Journal of Operational Research 108(1), 1-15, 1998.

[17] Moin, N.H., Salhi, S., and Aziz, N.A.B. An efficient hybrid genetic algorithm for the multi-product multi-period inventory routing problem. International Journal of Production Economics, Volume 133, Issue 1, pp 334-343, 2011.

[18] G. Mosheiov, The pickup delivery location problem on networks. Networks 26, 243-251, 1995.

[19] Nagy, G., Salhi, S. The many-to-many location routing problem. TOP. Vol. 6, 261-275, 1998.

[20] Nagy, G., Salhi, S. Location-routing: Issues, models and methods. European Journal of Operational Research 177 (2), 649-672, 2007.

[21] Nguyen, V.-P., Prins, C., and Prodhon, C. Solving the Two-Echelon Location Routing Problem by a hybrid GRASPxPath Relinking complemented by a learning process. European Journal of Operational Research, 216, pp. 113-126, 2012.

[22] Ozdamar, L., and Demir, O. A hierarchical clustering and routing procedure for large scale disaster relief logistics planning. Transportation Research Part E: Logistics and Transportation Review, Volume 48, Issue 3, pp 591-602, 2012.

[23] Prins, C., Prodhon, C., Wolfler Calvo, R. Solving the capacitated location-routing problem by a GRASP complemented by a learning process and a path relinking. 4OR 4,221-238, 2006.

[24] Prodhon C, and Prins, C. A survey of recent research on location-routing problem. European journal of Operational Research, vol. 238(1), 1-17, 2014.

[25] Prins, C. Labadi, N. Reghioui, M. 2009. Tour splitting algorithms for vehicle routing problems, International Journal of Production Research, 47(2), pp. 507-536.

[26] Rahmani, Y., Oulamara, A., and Ramdane Cherif, W. MultiProducts LocationRouting problem with Pickup and Delivery: Two-Echelon model.11th IFAC Workshop on Intelligent Manufacturing - IMS 2013, Sao Paulo, Brazil, pages 124-129, 2013. ISBN: 978-3-902823-33-5.

[27] Rahmani, Y., Oulamara, A., and Ramdane Cherif, W. Multi-products Location-Routing problem with Pickup and Delivery, ICALT IEEE, Sousse, Tunisie, pages 115-122, 2013.

[28] Rieck. J., Ehrenberg, C., and Zimmermann, J. Many-to-many location-routing with inter-hub transport and multi-commodity pickup-and-delivery, European Journal of Operational Research 236(3), 863878, 2014.

[29] Rodriguez-Martin, I., and Salazar-Gonzalez, J.-J. The multi-commodity one-to-one pickup-and-delivery traveling salesman problem: A matheuristic. 5th International Conference INOC 2011. Lecture Notes in Computer Science, 6701, pp. 401-405, 2011.

[30] Schwengerer, M., Pirkwieser, S., and Raidl. G. R. A variable neighborhood search approach for the two-echelon location-routing problem. In J.-K. Hao and M. Middendorf, editors, Evolutionary Computation in Combinatorial Optimisation - EvoCOP 2012. Lecture Notes in computer Science, Vol. 7245, pp 13-24, 2012.

[31] Villegas, J.G., Prins, C. Prodhon, C., Medaglia, A.L., and N. Velasco, N. GRASPVND and multi-start evolutionary local search for the single truck and trailer routing problem with satellite depots, Engineering Applications of Artificial Intelligence. 23, 780-794, 2011.

[32] Zare Mehrjerdi, Y., and Nadizadeh, A. Using greedy clustering method to solve capacitated location-routing problem with fuzzy demands. European Journal of Operational Research. Vol. 229, Issue 1, Pages 7584, 2013.

[33] Jacobsen, S., Madsen, O. (1980). A comparative study of heuristics for a two-level routing-location problem. European Journal of Operational Research, 6, 378387.

[34] Madsen, O. (1983). Methods for solving combined two level location-routing problems of realistic dimensions. European Journal of Operational Research, 12, 295301.

[35] Karaoglan, I., Altiparmak, F., Kara I., Dengiz, B. (2012). The location-routing problem with simultaneous pickup and delivery: Formulations and a heuristic approach, OMEGA - The International Journal of Management Science.

[36] G. Berbeglia, J.-F. Cordeau, I. Gribkovskaia and G. Laporte, 2007. Static Pickup and Delivery Problems: A Classification Scheme and survey. TOP, 15:1-31, 2007.

[37] Parragh, S.N., K.F. Doerner, R.F. Hartl. 2008. A survey on pickup and delivery problems Part II: Transportation between pickup and delivery locations. Journal fu?r Betriebswirtschaft 58(2) 81 117.

[38] Nguyen, V.-P., Prins, C., Prodhon, C. (2012b). A multi-start iterated local search with tabu list and path relinking for the two-echelon location-routing problem. Engineering Applications of Artificial Intelligence, 25(1), 5671.

[39] Miller, C. E., Tucker, A.. W. and Zemlin, R. A. (1960). Integer programming formulation of traveling salesman problems, Journal of ACM, Vol. 7, pp.3269.

[40] Rodríguez-Martín, I., Salazar-González, J., Yaman, H. 2014. A branch-and-cut algorithm for the hub location and routing problem Original Research Article. Computers Operations Research, Volume 50, Pages 161-174.

[41] Coelho, L. C., Laporte, G. 2013. A Branch-and-Cut Algorithm for the Multi-Product Multi-Vehicle Inventory-Routing Problem, International Journal of Production Research,Volume 51, Issue 23-24.

[42] Willmer Escobar, J., Linfati, R., G. Baldoquin, M. Toth, P. 2014. A Granular Variable Tabu Neighborhood Search for the capacitated location-routing problemOriginal Research Article. Transportation Research Part B: Methodological, Volume 67, Pages 344-356.

[43] Karaoglan, I., Altiparmak, F. 2014. A memetic algorithm for the capacitated location-routing problem with mixed backhauls Original Research Article. Computers Operations Research, In Press, Corrected Proof, Available online 17 June 2014.

[44] Lin, S-W., F. Yu, V. 2014. Multi-start Simulated Annealing Heuristic for the Location Routing Problem with Simultaneous Pickup and Delivery Original Research Article. Applied Soft Computing, In Press, Accepted Manuscript, Available online 26 June 2014.

[45] Cuda, R., Guastaroba, G., Speranza, M.G. 2014. A survey on two-echelon routing problemsOriginal Research Article. Computers Operations Research, In Press, Corrected Proof, Available online 17 June 2014.

Instances	Nearest Neighbor processing center selection				Score based processing center selection			
	NNPCS-RI		NNPCS-SI		SBPCS-RI		SBPCS-SI	
	Time	Cost(%)	Time	Cost(%)	Time	Cost(%)	Time	Cost(%)
coord20-5-1-2e	0	41085(15,70)	1	37447(7,51)	1	37488(7,62)	0	37447(7,51)
coord20-5-1b-2e	1	29875(39,22)	1	24113(24,70)	0	24692(26,46)	0	19877(8,65)
coord20-5-2-2e	0	36938(3,94)	1	50969(30,38)	2	41526(14,55)	1	36800(3,58)
coord20-5-2b-2e	2	48719(58,48)	0	35053(42,29)	1	27937(27,59)	3	27941(27,60)
coord50-5-1-2e	0	44147(32,37)	1	53833(44,54)	2	34800(14,21)	4	44125(32,34)
coord50-5-2-2e	3	56540(12,81)	1	57900(14,86)	2	54395(9,38)	3	55242(10,77)
coord50-5-2bBIS-2e	2	36093(22,65)	1	35784(21,98)	2	34678(19,49)	4	34174(18,30)
coord50-5-2BIS-2e	4	35479(17,72)	2	35556(17,90)	2	36224(19,41)	2	35556(17,90)
coord50-5-3-2e	2	52546(25,13)	1	53234(26,10)	2	44461(11,52)	3	43859(10,30)
coord100-5-1-2e	4	208053(13,65)	2	204336(12,08)	3	204441(12,12)	4	196936(8,78)
coord100-5-2-2e	3	216913(7,88)	3	223017(10,40)	3	231267(13,60)	5	217352(8,07)
coord100-5-3-2e	5	210661(5,43)	2	211117(5,64)	3	212231(6,13)	4	209812(5,05)
coord100-10-1-2e	2	285302(26,24)	2	324373(35,13)	2	336742(37,51)	4	271890(22,60)
coord100-10-2-2e	4	213916(11,56)	2	229191(17,46)	4	225103(15,96)	5	199861(5,34)
coord100-10-3-2e	4	245942(26,16)	2	265139(31,51)	2	219894(17,41)	4	243523(25,43)
coord200-10-1-2e	8	434322	4	459966	5	419625	5	351502
coord200-10-2-2e	5	453836	3	458940	5	519466	6	534204
coord200-10-3-2e	6	450966	4	388641	4	384127	8	374447
coord20-5-1-2e	0	31713(5,96)	1	32624(8,59)	1	38072(21,67)	0	32158(7,26)
coord20-5-1b-2e	0	19820(8,57)	0	20329(10,86)	0	20029(9,53)	0	20329(10,86)
coord20-5-2-2e	1	46886(25,66)	1	46350(24,80)	1	40485(13,91)	1	36655(4,91)
coord20-5-2b-2e	1	27348(27,80)	0	27345(27,79)	1	37814(47,78)	0	25092(21,31)
coord50-5-1-2e	0	36901(43,09)	2	45215(53,55)	2	44874(53,20)	2	39933(47,41)
coord50-5-2-2e	2	54218(13,82)	1	53010(11,86)	1	51839(9,87)	1	53010(11,86)
coord50-5-2bBIS-2e	1	31715(24,22)	1	32858(26,86)	2	33055(27,29)	1	32858(26,86)
coord50-5-2BIS-2e	2	32671(18,74)	1	32668(18,73)	10	32582(18,52)	1	32668(18,73)
coord50-5-3-2e	1	41773(34,88)	1	41726(34,81)	2	35535(23,45)	1	35989(24,42)
coord100-5-1-2e	4	197096	2	192741	11	194026	2	192741
coord100-5-2-2e	2	217694	2	217180	10	217649	2	217180
coord100-5-3-2e	2	207631	2	205755	14	204793	2	205197
coord100-10-1-2e	3	275802	1	320904	13	272930	1	314142
coord100-10-2-2e	3	215783	2	220747	16	215182	1	195688
coord100-10-3-2e	2	193425	2	233956	3	190204	2	189526
coord200-10-1-2e	5	387708	3	459966	7	417922	3	410020
coord200-10-2-2e	4	422734	3	422704	11	436954	2	525280
coord200-10-3-2e	6	422670	3	380102	8	480593	4	368830
Average	2,61 Seconds	(21,74%)	1,69 Seconds	(23,35%)	4,38 Seconds	(19,92%)	2,52 Seconds	(16,08%)

SBPCS-SI: Score based processing center selection - Scored Initialization. SBPCS-RI: Nearest Neighbor processing center selection - Random Initialization. NNPCS-SI: Nearest Neighbor processing center selection - Scored Initialization. NNPCS-RI: Nearest Neighbor processing center selection - Random Initialization

Table 2: Computational results for Nearest Neighbor strategy

Instances	BIH_1				BIH_2			
	$BIH_1 - NNCI$		$BIH_1 - RCI$		$BIH_2 - NNCI$		$BIH_2 - RCI$	
	Time	Cost(%)	Time	Cost(%)	Time	Cost(%)	Time	Cost(%)
coord20-5-1-2e	2	36705(5,65%)	1	36551(5,25%)	2	37199(6,90%)	2	42573(18,65%)
coord20-5-1b-2e	1	19946(8,96%)	1	19952(8,99%)	1	20629(11,98%)	1	26702(32,00%)
coord20-5-2-2e	1	37039(4,20%)	2	37476(5,32%)	3	37564(5,54%)	0	38008(6,64%)
coord20-5-2b-2e	2	27436(26,26%)	0	27912(27,52%)	1	28007(27,77%)	0	27994(27,73%)
coord50-5-1-2e	5	46481(35,77%)	6	45784(34,79%)	6	47051(36,55%)	4	47326(36,92%)
coord50-5-2-2e	4	55224(10,74%)	4	55656(11,43%)	5	57670(14,52%)	4	57222(13,85%)
coord50-5-2bBIS-2e	4	34862(19,92%)	5	35369(21,06%)	6	36465(23,44%)	4	37449(25,45%)
coord50-5-2BIS-2e	5	34610(15,65%)	6	35903(18,69%)	4	37122(21,36%)	5	37189(21,50%)
coord50-5-3-2e	5	44666(11,92%)	5	44826(12,24%)	6	47045(16,38%)	3	46533(15,46%)
coord100-5-1-2e	10	198039(9,28%)	9	199568 (9,98%)	15	202088 (11,10%)	6	202634(11,34%)
coord100-5-2-2e	10	217799(8,25%)	9	218459 (8,53%)	16	222830 (10,33%)	5	222432(10,17%)
coord100-5-3-2e	10	209136(4,74%)	9	210427 (5,33%)	16	214537 (7,14%)	6	213981(6,90%)
coord100-10-1-2e	11	271464(22,48%)	7	272681(22,83%)	19	277508(24,17%)	8	276443(23,88%)
coord100-10-2-2e	11	206995(8,61%)	6	204574(7,52%)	13	205600(7,99%)	7	205807(8,08%)
coord100-10-3-2e	9	245693(26,09%)	8	245966(26,17%)	19	250185(27,41%)	13	250219(27,42%)
coord200-10-1-2e	16	349657	15	350907	30	353933	26	358573
coord200-10-2-2e	19	536103	22	538763	40	543690	33	546601
coord200-10-3-2e	20	370812	23	371676	29	378365	25	378308
coord20-5-1-2e	1	31830(6,31%)	1	36551(18,41%)	3	31838(6,33%)	2	32791(9,05%)
coord20-5-1b-2e	0	19857(8,74%)	1	19952(9,18%)	0	20347(10,94%)	1	20927(13,41%)
coord20-5-2-2e	1	36894(5,53%)	2	37476(7,00%)	3	37419(6,85%)	3	46205(24,57%)
coord20-5-2b-2e	1	25243(21,78%)	0	27912(29,26%)	1	25854(23,63%)	2	28457(30,61%)
coord50-5-1-2e	5	40984(48,76%)	6	45784(54,13%)	7	42340(50,40%)	6	49073(57,20%)
coord50-5-2-2e	5	52234(10,55%)	4	55656(16,05%)	6	54620(14,45%)	5	54076(13,59%)
coord50-5-2bBIS-2e	5	33555(28,38%)	5	35369(32,05%)	6	35947(33,14%)	3	35005(31,34%)
coord50-5-2BIS-2e	5	31722(16,31%)	6	35903(26,06%)	6	34234(22,45%)	3	33220(20,08%)
coord50-5-3-2e	7	37045(26,57%)	5	44826(39,32%)	7	38089(28,58%)	4	38197(28,78%)
coord100-5-1-2e	14	193382	9	199568	10	196190	13	198093
coord100-5-2-2e	10	218267	9	218459	14	223042	9	221776
coord100-5-3-2e	13	205256	9	210427	13	210019	10	208456
coord100-10-1-2e	15	315304	7	272681	16	320729	9	319604
coord100-10-2-2e	15	196437	6	204574	11	201251	9	202025
coord100-10-3-2e	14	190480	8	245966	9	194256	6	195907
coord200-10-1-2e	35	413665	15	350907	34	421009	23	419356
coord200-10-2-2e	39	527055	22	538763	29	536500	23	536739
coord200-10-3-2e	33	367286	23	371676	25	375808	22	375460
Average	10,08 Seconds	(16,31%)	7,66 Seconds	(19,05%)	11,97 Seconds	(18,72%)	8,47 Seconds	(21,44%)

BIH: Best Insertion Heuristic. NNCI: Nearest Neighbor client Insertion. RCI: Random client Insertion.

Table 3: Computational results for Insertion method

Instances	Clustering Method			
	C-NNH	HCA-MA	HCA-AM	HCA-CH
coord20-5-1-2e	36853(6,02%)	35799(3,26%)	35799(3,26%)	35312(1,92%)
coord20-5-1b-2e	19826(8,41%)	19824(8,40%)	19834(8,45%)	19332(6,07%)
coord20-5-2-2e	36584(3,01%)	36123(1,77%)	36097(1,70%)	35670(0,52%)
coord20-5-2b-2e	27315(25,94%)	26737(0,65%)	26734(24,33%)	26770(24,43%)
coord50-5-1-2e	45769(34,77%)	43014(30,59%)	43014(30,59%)	42139(6,33%)
coord50-5-2-2e	55322(10,89%)	51740(4,73%)	52255(5,66%)	52366(5,86%)
coord50-5-2bBIS-2e	34300(18,60%)	31466(11,27%)	31938(12,58%)	31629(11,73%)
coord50-5-2BIS-2e	35556(17,90%)	32192(9,32%)	32222(9,40%)	32262(9,51%)
coord50-5-3-2e	45035(12,65%)	41508(5,22%)	43064(8,65%)	41675(5,60%)
coord100-5-1-2e	201283(10,75%)	192376(6,61%)	193815(7,31%)	192535(6,69%)
coord100-5-2-2e	217284(8,04%)	212746(6,08%)	214621(6,90%)	212353(5,90%)
coord100-5-3-2e	212641(6,31%)	204080(2,38%)	206128(3,35%)	204376(2,52%)
coord100-10-1-2e	-	263998(20,29%)	266003(20,89%)	266299(20,98%)
coord100-10-2-2e	-	197236(4,08%)	-	-
coord100-10-3-2e	-	192582(5,70%)	239958(24,32%)	-
coord200-10-1-2e	-	-	-	-
coord200-10-2-2e	-	-	-	-
coord200-10-3-2e	-	-	-	-
coord20-5-1-2e-2	32029(6,89%)	30982(3,74%)	30980(3,73%)	31009(3,82%)
coord20-5-1b-2e-2	20280(10,65%)	19808(8,52%)	19808(8,52%)	19285(6,04%)
coord20-5-2-2e-2	36077(3,39%)	35454(1,69%)	35947(3,04%)	47109(26,01%)
coord20-5-2b-2e-2	24653(19,91%)	25049(21,17%)	25052(21,18%)	24580(19,67%)
coord50-5-1-2e-2	38961(46,10%)	37314(43,72%)	38316(45,19%)	37409(43,86%)
coord50-5-2-2e-2	50932(8,26%)	49148(4,93%)	50157(6,84%)	49337(5,29%)
coord50-5-2bBIS-2e-2	32858(26,86%)	30103(20,16%)	30593(21,44%)	30684(21,68%)
coord50-5-2BIS-2e-2	32685(18,78%)	29322(9,46%)	29271(9,30%)	28917(8,19%)
coord50-5-3-2e-2	37262(27,00%)	33194(18,05%)	34170(20,39%)	33783(19,48%)
coord100-5-1-2e-2	192746	187821	189768	188085
coord100-5-2-2e-2	217268	212659	-	212362
coord100-5-3-2e-2	205192	199624	-	199776
coord100-10-1-2e-2	314032	307634	-	307958
coord100-10-2-2e-2	195560	190501	-	190794
coord100-10-3-2e-2	189263	185085	-	-
coord200-10-1-2e-2	-	-	-	-
coord200-10-2-2e-2	-	-	-	-
coord200-10-3-2e-2	-	-	-	-
Average	(15,77%)	(10,49%)	(13,35%)	(11,91%)

C_NNH: Clustering Nearest Neighbor Heuristic. HCA_MA: Hybrid Clustering Algorithm(First Merging second Assigning). HCA_AM: Hybrid Clustering Algorithm(First Assingning second Merging). HCA_CH: Hybrid Clustering Algorithm (Client_Hub)

Table 4: Computational results for Clustering method

Instances	Dominant Strategy				
	$BIH_1 - NNCI$		SBPCS-SI		HCA-MA
	Time	Cost(%)	Time	Cost(%)	Cost(%)
coord20-5-1-2e-2	2	36705(2,47%)	0	37447(4,40%)	35799(0%)
coord20-5-1b-2e-2	1	19946 (0,61%)	0	19877 (0,27%)	19824 (0%)
coord20-5-2-2e-2	1	37039 (2,47%)	1	36800 (1,84%)	36123 (0%)
coord20-5-2b-2e-2	2	27436 (2,55%)	3	27941 (4,31%)	26737 (0%)
coord50-5-1-2e-2	5	46481 (7,46%)	4	44125 (2,52%)	43014 (0%)
coord50-5-2-2e-2	4	55224 (6,31%)	3	55242 (6,34%)	51740 (0%)
coord50-5-2bBIS-2e	4	34862 (9,74%)	4	34174 (7,92%)	31466 (0%)
coord50-5-2BIS-2e-2	5	34610 (6,99%)	2	35556 (9,46%)	32192 (0%)
coord50-5-3-2e	5	44666 (7,07%)	3	43859 (5,36%)	41508 (0%)
coord100-5-1-2e	10	198039 (2,86%)	4	196936 (2,32%)	192376 (0%)
coord100-5-2-2e	10	217799 (2,32%)	5	217352 (2,12%)	212746 (0%)
coord100-5-3-2e	10	209136 (2,42%)	4	209812 (2,73%)	204080 (0%)
coord100-10-1-2e	11	271464 (2,75%)	4	271890 (2,90%)	263998 (0%)
coord100-10-2-2e	11	206995 (4,71%)	5	199861 (1,31%)	197236 (0%)
coord100-10-3-2e	9	245693 (21,62%)	4	243523 (20,92%)	192582 (0%)
coord200-10-1-2e	16	349657 (0%)	5	351502 (0,52%)	-
coord200-10-2-2e	19	536103 (0,35%)	6	534204 (0%)	-
coord200-10-3-2e	20	370812 (0%)	8	374447 (0,97%)	-
coord20-5-1-2e	1	31830 (2,67%)	0	32158 (2,31%)	30982 (0%)
coord20-5-1b-2e	0	19857 (2,88%)	0	20329 (2,70%)	19808 (0%)
coord20-5-2-2e	1	36894 (3,90%)	1	36655 (3,28%)	35454 (0%)
coord20-5-2b-2e	1	25243 (2,63%)	0	25092 (2,04%)	25049 (0%)
coord50-5-1-2e	5	40984 (9,96%)	2	39933 (0%)	37314 (1,10%)
coord50-5-2-2e	5	52234 (5,91%)	1	53010 (5,19%)	49148 (0%)
coord50-5-2bBIS-2e	5	33555 (8,83%)	1	32858 (3,54%)	30103 (0%)
coord50-5-2BIS-2e	5	31722 (8,84%)	1	32668 (11,25%)	29322 (0%)
coord50-5-3-2e	7	37045 (10,40%)	1	35989 (6,59%)	33194 (0%)
coord100-5-1-2e	14	193382 (2,74%)	2	192741 (2,42%)	187821 (0%)
coord100-5-2-2e	10	218267 (2,71%)	2	217180 (2,22%)	212659 (0%)
coord100-5-3-2e	13	205256 (2,74%)	2	205197 (2,52%)	199624 (0%)
coord100-10-1-2e	15	315304 (0%)	1	314142 (0,09%)	307634 (11,36%)
coord100-10-2-2e	15	196437 (3,02%)	1	195688 (2,65%)	190501 (0%)
coord100-10-3-2e	14	190480 (2,83%)	2	189526 (2,34%)	185085 (0%)
coord200-10-1-2e	35	413665 (0%)	3	410020 (9,49%)	-
coord200-10-2-2e	39	527055 (19,79%)	2	525280 (0%)	-
coord200-10-3-2e	33	367286 (0%)	4	368830 (0,4%)	-
Average	10,08 Seconds	(5,44%)	2,52 Seconds	(4,39%)	(0,72%)

BIH: Best Insertion Heuristic. NNCI: Nearest Neighbor client Insertion. HCA_MA: Hybrid Clustering Algorithm(First Merging second Assigning). SBPCS-SI: Score based processing center selection - Scored Initialization

Table 5: Comparison of the Dominant Strategy for each method

Instances	Best Solution				
	Insertion method		NNH		HCA
	Time	Cost(%)	Time	Cost(%)	Cost(%)
coord20-5-1-2e	2	36551(3,38%)	0	37447 (5,70%)	35312 (0%)
coord20-5-1b-2e	1	19946(3,07%)	0	19877 (2,74%)	19332 (0%)
coord20-5-2-2e	1	37039(3,69%)	1	36800 (3,07%)	35670 (0%)
coord20-5-2b-2e	2	27436(2,55%)	3	27937 (4,31%)	26734 (0%)
coord50-5-1-2e	5	45784(23,99%)	4	34800 (0%)	42139 (17,41%)
coord50-5-2-2e	4	55224(6,30%)	3	54395 (4,88%)	51740 (0%)
coord50-5-2bBIS-2e	4	34862(9,74%)	4	34174 (7,92%)	31466 (0%)
coord50-5-2BIS-2e	5	34610(6,98%)	2	35479 (9,26%)	32192 (0%)
coord50-5-3-2e	5	44666(7,07%)	3	43859 (5,36%)	41508 (0%)
coord100-5-1-2e	10	198039(2,85%)	4	196936 (2,32%)	192376 (0%)
coord100-5-2-2e	10	217799(2,50%)	5	216913 (2,10%)	212353 (0%)
coord100-5-3-2e	10	209136(2,41%)	4	209812 (2,73%)	204080 (0%)
coord100-10-1-2e	11	271464(2,75%)	4	271890 (2,90%)	263998 (0%)
coord100-10-2-2e	11	204574(3,59%)	5	199861 (1,31%)	197236 (0%)
coord100-10-3-2e	9	245693 (21,62%)	4	219894 (12,42%)	192582 (0%)
coord200-10-1-2e	16	349657 (0%)	5	351502 (0,52%)	-
coord200-10-2-2e	19	536103 (3,10%)	6	519466 (0%)	-
coord200-10-3-2e	20	370812 (0%)	8	374447 (0,97%)	-
coord20-5-1-2e	1	31830 (2,67%)	0	31713 (2,31%)	30980 (0%)
coord20-5-1b-2e	0	19857 (2,88%)	0	19820 (2,70%)	19285 (0%)
coord20-5-2-2e	1	36894 (3,90%)	1	36655 (3,28%)	35454 (0%)
coord20-5-2b-2e	1	25243 (2,63%)	0	25092 (2,04%)	24580 (0%)
coord50-5-1-2e	5	40984 (9,96%)	2	36901 (0%)	37314 (1,10%)
coord50-5-2-2e	5	52234 (5,91%)	1	51839 (5,19%)	49148 (0%)
coord50-5-2bBIS-2e	5	33555 (8,83%)	1	31715 (3,54%)	30593 (0%)
coord50-5-2BIS-2e	5	31722 (8,84%)	1	32582 (11,25%)	28917 (0%)
coord50-5-3-2e	7	37045 (10,40%)	1	35535 (6,59%)	33194 (0%)
coord100-5-1-2e	14	193382 (2,74%)	2	192741 (2,42%)	188085 (0%)
coord100-5-2-2e	10	218267 (2,71%)	2	217180 (2,22%)	212362 (0%)
coord100-5-3-2e	13	205256 (2,74%)	2	204793 (2,52%)	199624 (0%)
coord100-10-1-2e	15	272681 (0%)	1	272930 (0,09%)	307634 (11,36%)
coord100-10-2-2e	15	196437 (3,02%)	1	195688 (2,65%)	190501 (0%)
coord100-10-3-2e	14	190480 (2,83%)	2	189526 (2,34%)	185085 (0%)
coord200-10-1-2e	35	350907 (0%)	3	387708 (9,49%)	-
coord200-10-2-2e	39	527055 (19,79%)	2	422704 (0%)	-
coord200-10-3-2e	33	367286 (0%)	4	368830 (0,4%)	-
Average	11,09 Seconds	(6,25%)	3,02 Seconds	(4,41%)	(0,42%)

NNH: Nearest Neighbor Heuristic. HCA: Hybrid Clustering Algorithm.

Table 6: Comparison of the best obtained results by all strategies