CIRRELT

Centre interuniversitaire de recherche sur les réseaux d'entreprise, la logistique et le transport

Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation

Multi-Commodity Location-Routing: Flow Intercepting Formulation and Branch-and-Cut Algorithm

Maurizio Boccia
Teodor Gabriel Crainic
Antonio Sforza
Claudio Sterle

July 2016

CIRRELT-2016-38

Bureaux de Montréal :
Université de Montréal
Pavillon André-Aisenstadt
C.P. 6128, succursale Centre-ville
Montréal (Québec)
Canada H3C 3J7
Téléphone : 514 343-7575
Télécopie : 514 343-7121

Bureaux de Québec :
Université Laval
Pavillon Palasis-Prince
2325, de la Terrasse, bureau 2642
Québec (Québec)
Canada G1V 0A6
Téléphone : 418 656-2073
Télécopie : 418 656-2624

www.cirrelt.ca

Multi-Commodity Location-Routing: Flow Intercepting Formulation and Branch-and-Cut Algorithm

Maurizio Boccia[1], Teodor Gabriel Crainic[2,*], Antonio Sforza[3], Claudio Sterle[3]

[1] Department of Engineering, University of Sannio, Piazza Roma, 21, 82100, Benevento, Italy

[2] Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation (CIRRELT) and Department of Management and Technology, Université du Québec à Montréal, P.O. Box 8888, Station Centre-Ville, Montréal, Canada H3C 3P8

[3] Department of Electrical Engineering and Information Technology, University "Federico II" of Naples, Via claudio 21, 80125 Naples, Italy

Abstract. Research on the location-routing problem (*LRP*) is very active, producing a good number of effective exact and approximated solution approaches. It is noteworthy that most of the contributions present in the literature address the *single-commodity LRP*, whereas the *multi-commodity* case has been scarcely investigated. Yet, this issue assumes an important role in many *LRP* applications, particularly in the context of designing single-tier freight distribution City Logistics (*CL*) systems. To fill this gap, we define a new multi-commodity *LRP*, proposing an original ILP formulation for it. The proposed formulation takes into account the multi-commodity feature of the problem, modeling the strategic location and the tactical routing decisions using the flow intercepting approach. We therefore name this problem the *flow intercepting facility location-routing problem* (*FIFLOR*). The problem is solved by a branch-and-cut algorithm which exploits cuts derived and adapted from literature. The proposed method is successfully experienced and validated on test instances reproducing different network topologies and problem settings.

Keywords: Multi-commodity location-routing, flow intercepting facility location, branch-and-cut, city logistics.

Acknowledgements. Partial funding for this project has been provided by the Natural Sciences and Engineering Research Council of Canada (NSERC) through its Discovery Grants program. We also gratefully acknowledge the support of the Fonds de recherche du Québec through their infrastructure grants. While working on this project, the second author was Adjunct Professor with the Department of Computer Science and Operations Research, Université de Montréal. The research activity of the third and fourth author was partially funded by the Department of Electrical Engineering and Information Technology of University "Federico II" of Naples, within the research project on "Optimization Models and Methods for ITS".

---

* Corresponding author: teodorgabriel.crainic@cirrelt.net

Dépôt légal – Bibliothèque et Archives nationales du Québec
Bibliothèque et Archives Canada, 2016

1. Introduction

The literature on the location-routing problem (*LRP*) has significantly increased in the last ten years, even if the first contribution combining location and routing decisions dates back to the work by Maranzana (1964). In its basic version, *LRP* is referred to as an integrated *strategic* and *tactical* decision problem and it can be defined as follows. Given a set of potential facility locations and a set of customer demands to be satisfied, we have to simultaneously determine the number and position of one or more facilities (*strategic location decisions*); the customer-to-facility (one-to-one) assignment (*strategic assignment decisions*); the size of the vehicle fleet used to serve the customers and the routes to be performed by each vehicle dispatched from the located facilities (*tactical routing decisions*). The aim is the minimization of the total system costs, given by two components: the fixed location cost, which may depend on the specific position of the facility to be installed, and the distribution cost, associated to the delivery operations (generally this cost incorporates also a fixed cost for the vehicle usage).

LRP, related variants and applications have been largely addressed through exact and approximated approaches, as illustrated by the recent surveys of Prodhon and Prins (2014), Drexl and Scheneider (2015) and Cuda *et al*. (2015). Yet, to the best of the authors' knowledge, most of the *LRP* contributions present in literature deal with *single-commodity* flows, whereas the *multi-commodity* case has been scarcely investigated. This issue assumes a relevant role in many *LRP* applications, in particular in the ones arising in City Logistics (*CL*), concerning the design of *single* and *multi-tier* freight distribution systems (Bektas *et al*., 2016; Crainic *et al*., 2009; Boccia *et al*., 2010; Mancini *et al*., 2014). Indeed, the goods/service demand to be managed in an urban area is highly customized. Hence, the corresponding distribution problem is strongly and inherently a *multi-commodity flow problem*.

This work is aimed at filling this gap. Its focus is on the multi-commodity *LRP* definition and solution. The problem will be defined in a *CL* perspective and the design of a *single-tier* urban freight distribution system will be the driving application. Hence, for the sake of clarity, we briefly recall the *single tier* basic idea (Taniguchi *et al*., 1999; Crainic *et al*., 2004), highlighting new design issues that have to be integrated in a multi-commodity *LRP*. We then focus on the proposed methodological aspects.

The aim of a *single-tier* system is to prevent the penetration of a large number of commercial vehicles, coming from the city outskirts (intermodal platforms, commercial hubs, etc.) and directed to the city center, stopping them at intermediate logistic facilities. Here the freight flows of different carriers are deconsolidated, transferred and consolidated into *smaller* and *green vehicles*, more suitable to perform the delivery to the final customers. In other words, each freight flow has to pass through one load break point, i.e. an intermediate logistic facility (generally referred to as city distribution center, CDC) and undergo several handling activities (cross-docking, unloading/loading and deconsolidation/consolidation). This allows to reduce the vehicle kilometers traveled and to remove vehicles from the urban network. Such systems have already found their applications in several urban areas, e.g., in Munich, Siena and Padova, where large size vehicles are interdicted to enter in the city center and final distribution is performed by freight distribution companies (Crainic *et al*., 2009 and Morana J., 2014).

Given this description, we can easily recognize the associated *LRP*. Indeed, in order to design an efficient and effective *single-tier* system, we have to determine, at the same time, the number and the location of the intermediate logistic facilities and the routes to be performed by the vehicles to supply customer demands, minimizing the total system cost.

Moreover, two other important issues should be also properly taken into account in a *CL* perspective looking for solutions that are acceptable and convenient for all the stakeholders (public authorities, goods/service companies, freight carriers and final customers). First, the customer-to-facility assignment should be made not only considering the position of the final customer, but also the origin of the commodity required by the customer. This, on one side makes the distribution system more coherent with traditional flows and used corridors. On the other side, it avoids long trips to reach the intermediate logistic facilities and/or the final customers, so impacting on the delivery time. Second**,** in many cities, and in particular in smaller ones, the choice of the logistic facility locations is strongly conditioned by the urban infrastructure and road system. Hence they should be located in correspondence of pre-existing structures (as for example available parking areas) and if possible, along or near the main entrance roads, consistently with the vehicular travel demand pattern. This, on one side, allows to reduce the installation costs, on the other side it makes the *CL* measure well-accepted by the stakeholders.

The main goals of this paper can then be stated as: 1) Defining a multi-commodity *LRP* integrating the just described *CL* design issues; 2) Proposing an original ILP formulation for the problem using a flow-intercepting approach for the location decisions; and 3) Proposing an exact solution approach and benchmark instances for this new problem.

The second goal requires a preliminary discussion about the choice of the modeling approach we propose. *LRP* formulations present in literature are generally obtained merging the ILP models of the two sub-problems composing it, i.e. the facility location and the vehicle routing problems. The routing sub-problem is approached with classical path or arc based formulations. A slightly modified arc based formulation is used in this paper. Concerning the location problem, it has always been tackled as a *point-based demand facility location* problem, generally adapting the formulations of the p-median or of the simple plant facility location problem.

In the proposed formulation we treat the location decision as a *flow-based demand facility location problem* (Zeng *et al*., 2010; Sterle *et al*. 2016), i.e., as a location problem where facilities do not generate or attract flows, but intercept them along their pre-planned trips from their origins to their destinations (Hodgson, 1981; Berman *et al*., 1992; Boccia *et al*. 2009). More precisely, knowing all the paths that carry non-zero flows on the network and assuming that a path can be intercepted exclusively by a facility located in the nodes composing it, two basic problems can be defined. The first is aimed at determining the location of a prefixed number of facilities maximizing the intercepted flow. The second is aimed at determining the location of the minimum number of facilities able to intercept the total flow traversing the network or a prefixed percentage of it. For these problems the expression "*flow intercepting facility location problems*" (*FIFLP*) is used and a path coverage approach is adopted.

The choice of using *FIFLP* in our *LRP* formulation has a twofold motivation. As will be shown in the following, *FIFLP* allows to easily integrate the other *CL* design issues discussed above. Moreover a path-based approach, including information about origin and destination of each commodity, naturally fits with *multi-commodity* flows. Indeed even if the point-based formulation can be easily modified to take into account more commodities, it generates a significant increase in the size of the problem, making it unsolvable in a reasonable time. For this reason, we call *flow-intercepting facility location-routing* problem (*FIFLOR*) the proposed multi-commodity *LRP* formulation. To the best of authors' knowledge, *FIFLOR* has never been treated before.

FIFLOR will be optimally solved by a branch-and-cut algorithm based on valid inequalities derived and adapted from literature. A heuristic procedure for the upper bound (*UB*) computation, exploiting the solution of the linear relaxation of the *FIFLOR* formulation, will also be presented. The proposed *LRP* approach has been experienced and validated on several test instances representing different scenarios of the *single-tier* freight distribution design problem.

The paper is structured as follows: *Section 2* recalls the main contributions on *FIFLP* and multi-commodity *LRP*, in order to position the *FIFLOR* in the literature. We describe the *FIFLOR* problem in *Section 3* presenting an original ILP formulation. *Section 4* is dedicated to the branch-and-cut algorithm and the heuristic procedure. Finally, we present and discuss numerical results in *Section 5*.

2. Literature Review

In the following, we provide a short literature review about the *FIFLP* and the multi-commodity *LRP*.

The literature on *FIFLP* is rather scarce compared to that of classical point-based facility location problems. To the best of authors' knowledge, *FIFLP* has never been used before in the *CL* context. Reviews of the main contributions can be found in Berman *et al*. (1995), Boccia *et al*. (2009) and Zeng *et al*. (2010). Contributions can be cited in several fields including: urban traffic management (counting sensors, cameras and variable message signs; Gentili and Mirchandani, 2005, Yang *et al*., 2006, and Sterle *et al*., 2016); park-and-ride (Horner and Groves, 2007); flow control/monitoring for security (inspection stations and checkpoints; Gendreau *et al*., 2000, Selmic *et al*., 2010, and De Cillis *et al*., 2013); logistics (convenience stores and refueling stations: Upchurch *et al*., 2009, Kim and Kuby, 2013, and Wen *et al*., 2014).

This brief list confirms the wide applicability of the *FIFLP* approach in different contexts. However the solution methodologies proposed for these applications cannot be easily generalized and transferred from an application to another, because they take into account practical constraints which are typical of the specific problem under investigation.

Concerning *LRP*, a review of the first papers on the theme can be found in Laporte *et al*. (1988) and Nagy and Salhi (2007). For the most recent contributions the interested reader is addressed to the surveys by Prodhon and Prins (2014), Drexl and Scheneider (2015) and Cuda *et al*. (2015). In particular the first is devoted to the basic *LRP*, its formulations and solving approaches. The second mainly deals with *LRP* variants and extensions. Finally the third is on the two-echelon vehicle routing problems, with or without location decisions (*2E-VRP* and *2E-LRP*). As will be clarified in the following, *FIFLOR* can be interpreted as a particular case of the *2E-LRP*. Hence all these concurrent reviews, even if present some overlapping, can be considered as complementary and provide a wide and detailed review of all the last advances (exact and heuristic approaches) and applications of *LRP*.

From these reviews, we can say that multi-commodity *LRP* has been scarcely treated in literature. To the best of authors' knowledge, the first work explicitly tackling multi-commodity *LRP* is the one by Burks (2006). The author adapted and integrated the ILP model proposed by Perl and Daskin (1985) and presented an ILP formulation solved by a tabu search metaheuristic. Similar developments were proposed by Hamidi *et al*. (2012) and (2014), where a multi-commodity *LRP* was modeled by an ILP formulation and solved by metaheuristic and local search approaches. Recently, Rath and Gutjahr (2014) defined a multi-commodity *LRP* in the context of disaster relief. However, assuming the goods to be distributed as homogenous, they formulated the problem as a single commodity *LRP*, solving it by a matheuristic. Giannessi et al. (2015) tackled a particular variant of the multi-commodity *LRP*, where the facilities to be located have to be connected via a ring. The authors proposed an ILP formulation and solved the problem by an exact method, a matheuristic and a hybrid approach. Finally, Rahmani et al. (2015a) and (2015b) addressed a particular variant of the multi-commodity *LRP* with pickup and delivery. In their problem a single commodity is distributed from origin nodes to intermediate facilities. At these facilities, it is transformed into several commodities to be distributed to the final customers. Also in these works an ILP formulation of the problem was provided and then it was solved by heuristic approaches.

As discussed above, focusing on the modeling aspect of these works, we can say that all the contributions share the same idea. They propose ILP formulations that extend the classical single-commodity *LRP* formulation to the multi-commodity case. This extension is performed specifying all the variables, or several of them, for each commodity to be distributed. All the authors highlight the high complexity of their formulations and, for this reason, they mainly focus on heuristics. This consideration supports our choice of proposing an original and alternative *LRP* formulation aimed at reducing this drawback.

3. The FIFLOR Problem

We first describe the *FIFLOR* problem, focusing on the issues related to the *single-tier* distribution system described above. We then present an original *ILP* model, highlighting the advantages of this formulation with respect to the classical *LRP* one.

3.1 FIFLOR problem description

Let us consider an urban network where we can identify the following sets belonging to three different layers at decreasing distance from the city center: freight origins (*first layer*); potential intermediate logistic facility locations (*second layer*), each of them characterized by the same capacity (expressed in terms of number of vehicles); freight destinations/final customers (*third layer*).

Each final customer is associated with a known freight demand to be provided by an origin supplying it. This means that each demand is related to a different commodity, which can be identified by its origin/destination (*o-d*) couple. Knowing the *o-d* demand matrix, the paths connecting an *o-d* couple could be retrieved by an equilibrium traffic assignment algorithm (Dial 2006). Each commodity is associated with just one of these *o-d* paths, which can be the shortest or the most frequently used one.

Given a commodity, let us assume that direct shipment from the origin to a final customer is not allowed. The distribution, instead, is made by capacitated vehicles, on which different commodity demands are consolidated. These vehicles perform multi-customer routes starting and ending at the same intermediate logistic facility. Hence a one-to-one commodity-to-facility assignment has to be determined, taking into account the capacity constraints of the facility. Each commodity and, consequently, each customer cannot be assigned to any of the logistic facilities, it must be assigned only to one of those located along the *o-d* path or within a predefined distance from it. In other words, a commodity can be assigned just to a facility that can intercept it. It is important to note that, given that each vehicle is associated with just one facility and the demand of each customer is assumed to be lower than the vehicle capacity, then the commodity-to-facility assignment provides also the assignment of the related customer to one of the vehicles dispatched from the facility.

Three different costs components have to be considered in *FIFLOR*: location, routing and assignment costs. Location and routing cost components are the ones which are generally considered in *LRP* and are related to the second and the third layer. Indeed, the first represents the cost for the installation of a facility at a given position, whereas the second represents the distribution cost. It can be assumed to be proportional to the length of the multi-customer routes performed by the vehicles. The third cost component, instead, is introduced in order to take into account the costs involving the first layer. In particular we refer to the additional cost paid to reach an intermediate facility from the commodity origin and to use the facility services. For this reason this cost can be assumed to be proportional to the demand of a commodity that has to be supplied and to the distance between the origin and the intermediate facility.

On this basis, *FIFLOR* aims to solve the following three decision problems:

- ***Location decisions***: Determine the number and the position of the capacitated facilities needed to serve all the customer (i.e., to intercept all the commodities near or along their *o-d* paths). The aim is the minimization of *facility location costs*.

- ***Assignment decisions:*** Determine the assignment of each commodity to one of the located intermediate facilities, without exceeding their capacity. The aim is the minimization of the *assignment costs*.
- ***Routing decisions:*** Determine the size of the capacitated homogeneous vehicle fleet of each intermediate facility and the routes to be performed for the final distribution. The aim is the minimization of the *routing costs*.

We provide in Figure 1 a network representation of two possible solutions for a small *FIFLOR* problem with six commodities. As explained above, the origin nodes are on the first layer. The intermediate potential facility location nodes are all the ones composing the *o*-*d* paths (second layer). The customers are on the third layer. A solution with three intermediate logistic facilities is represented in Figure 1a. From these facilities, three different vehicles are used to perform the distribution to the final customers (multi-stop routes indicated with dotted lines). Similarly, in Figure 1b, a solution with two intermediate facilities and two vehicles is represented. It is easy to observe that these two solutions provide different tradeoffs between the three cost components. Indeed, in the case of Figure 1a we have a solution where a higher number of facilities is installed with respect to the case of Figure 1b. This allows to intercept all the flows early en-route to their destinations, so avoiding the penetration of large amounts of commercial vehicles in the city center. Hence, the solution of Figure 1a returns low assignment costs, but higher location and routing costs with respect to the solution of Figure 1b. These considerations about costs will be further discussed in *Section 5*, where the results of instances differing in the order of magnitude of the three cost components are analyzed.

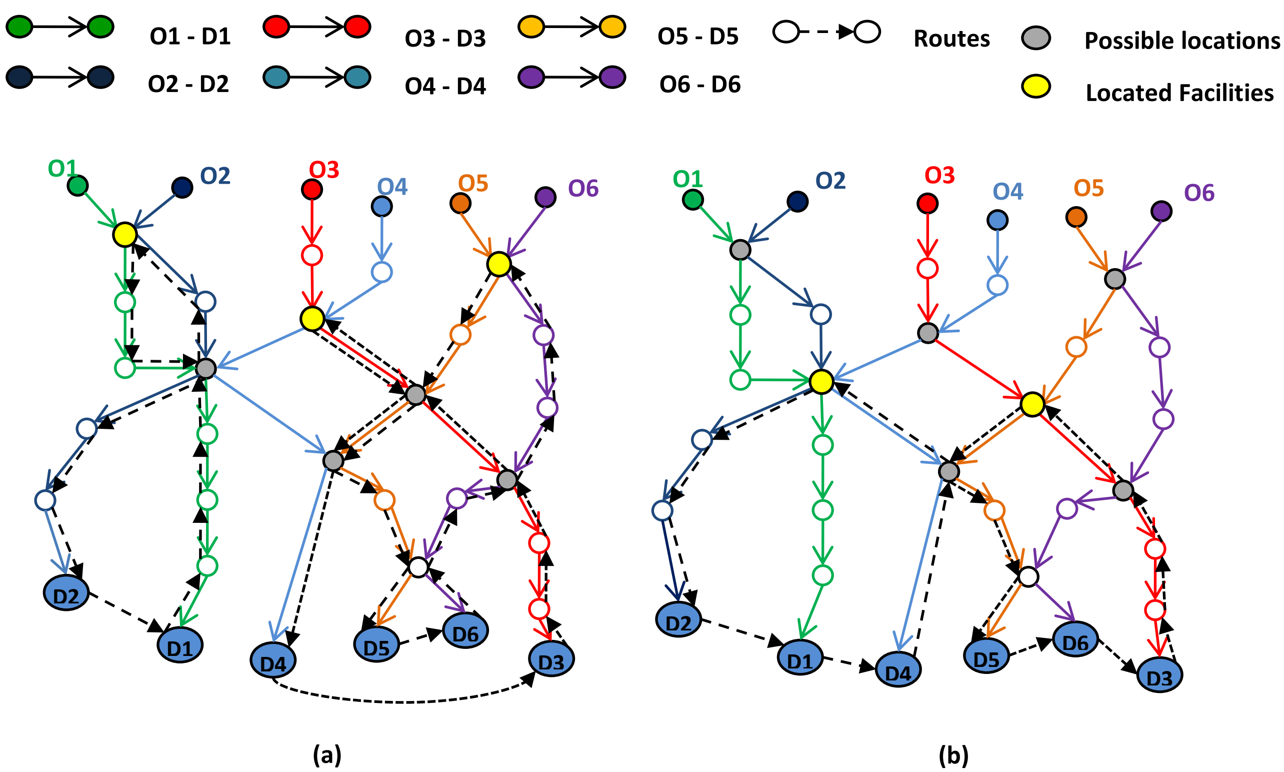


Figure 1. Two solutions of a *FIFLOR* problem: a) solution with three intermediate facilities; b) solution with two intermediate facilities.

It is important to note that the *FIFLOR* problem takes into account the *CL* issues discussed in *Section 1* related to customer-to-facility assignment and urban infrastructure road system. The key element resides in the usage of the *o-d* paths to identify the commodities travelling on the network. Indeed, this easily captures the multi-commodity feature of the problem under investigation. Moreover, the usage of the *o-d* paths for each commodity allows to easily consider the pre-existing urban structures, road system and corridors. Hence, public authorities could reduce the system design costs. Finally, since each commodity has to be assigned to a single logistic facility located along or near its *o-d* path, long deviations are avoided either from the origin nodes towards the logistic facilities or from the logistic facility towards the final customer. This has a twofold advantage. On one side it reduces congestion since it decreases the number of commercial vehicles in the area. On the other side, it makes the solution more acceptable for all involved stakeholders. Indeed, private companies and freight carriers would perform the distribution adopting a solution coherent with the case where no logistic facility is used. This should guarantee acceptable additional costs and delays.

From the above description, it is clear that *FIFLOR* problem is defined on a two-echelon structure, where the first-echelon is composed by the first and second layers, whereas the second echelon is composed by the second and third layers. Hence, using the *LRP* notation introduced by Laporte *et al*. (1988) and extended by Boccia *et al*. (2011), it can be classified as a $3/R/\bar{T}$ problem, i.e. a two-echelon location-routing problem, *2E-LRP*, with location decisions on the second layer, dedicated routes (represented by the assignment) on the first echelon, and multiple node routes on the second echelon. This illustrates the difference with the basic *LRP*, which can be defined as a $2/\bar{T}$ problem, where we just consider the second and the third layers and each customer is associated with a demand that can be supplied by any of the logistic facilities.

3.2 *FIFLOR* problem setting and formulation

Let $O = \{o_1, o_2, \dots, o_n\}$ be the set of freight *origin nodes* and let $D = \{d_1, d_2, \dots, d_n\}$ be the set of final customer/destination nodes. As explained above each *o-d* couple represents a commodity traveling on a pre-defined *o-d* path. Let $P = \{1, 2, \dots, n\}$ be the set of the commodities or, equally, the set of paths. Let o_p to d_p be the origin and destination nodes (final customers) of p, $p \epsilon P$, respectively. Moreover, let f_p be the flow traveling on path p, $p \epsilon P$, or in other words the demand for commodity p of the final customer d_p.

Let *J* be the set of potential facility location nodes and let $J_p \subseteq J$ be the subset of facilities that can intercept the path $p, p \epsilon P$, (from $o_p$to d_p), i.e., the subset of facilities to which the commodity *p* can be assigned. This information can be summarized in a path-node incidence matrix, *PN*, a binary matrix of size is $P \times J$, where the generic element *(p, j)* is equal to 1 if the path *p* can be intercepted by the facility located at node *j*, *0* otherwise, or in other words, if the commodity *p* can be serviced by facility *j*. This matrix will be used in the *UB* computation procedure described in *Section 4*.

Let $G = (O \cup J \cup D, A \cup E)$ be a graph where *A* is the set of arcs *(i, j)* with $i \in O$ and $j \in J$, and *E* is the set of arcs *(i, j)* with $i \in J \cup D$ and $j \in D$. An unit assignment cost d_{ij} is associated to each arc *(i, j)*, while a routing cost c_{ij} is associated with each arc $(i, j) \in E$. Both of them are proportional to the shortest path from *i* to *j*.

Let h_j be the (fixed) location cost of a facility at node j, $j \in J$, and let $K_j = n_j C$ be the capacity of facility j, $j \in J$, where n_j is the maximum number of vehicles and *C* is their capacity.

We define the following set of binary variables:

- y_j *location variable*, equals 1 if a facility is located at node j, $j \in J$, 0 otherwise.
- z_{pj} *assignment variable*, equals 1 if path (commodity) p, $p \in P$, is assigned to (intercepted by) the facility located at node j, $j \in J_p$, 0 otherwise; in other words it is equal to 1 if the customer d_p is serviced by a vehicle starting its tour from a facility located in j, $j \in J$.

- x_{il}^{j} *routing variable*, equals *1* if arc *(i,l)*,$(i,l) \in E$, belongs to a route made by a vehicles starting its route from facility j, $j \in J$, 0 otherwise.

Hence the *FIFLOR* problem can be formulated as follows:

$$FIFLOR : \min \sum_{j \in J} h_j y_j + \sum_{p \in P} \sum_{j \in J} f_p d_{o_p j} z_{pj} + \sum_{j \in J} \sum_{(i,l) \in E} c_{il} x_{il}^{j} \quad (1)$$

s. t.

$$\sum_{j \in J_p} z_{pj} = 1, \qquad \forall p \in P, \quad (2)$$

$$\sum_{p \in P} f_p z_{pj} \leq C y_j, \qquad \forall j \in J, \quad (3)$$

$$\sum_{l \in P} x_{jl}^{j} \leq n_j y_j, \qquad \forall j \in J, \quad (4)$$

$$\sum_{l \in P} x_{jl}^{j} = \sum_{i \in P} x_{ij}^{j}, \qquad \forall j \in J, \quad (5)$$

$$\sum_{l \in P \cup \{j\}} x_{lp}^{j} = z_{pj}, \qquad \forall p \in P, j \in J, \quad (6)$$

$$\sum_{l \in P \cup \{j\}} x_{pl}^{j} = z_{pj}, \qquad \forall p \in P, j \in J, \quad (7)$$

$$\sum_{l \in (P \setminus S) \cup \{j\}} \sum_{p \in S} x_{pl}^{j} \geq \sum_{p \in S} \frac{f_p}{C} z_{pj}, \qquad \forall j \in J, S \subset P, \quad (8)$$

$$z_{pj} \in \{0,1\}, y_j \in \{0,1\}, x_{lp}^{j} \in \{0,1\} \qquad \forall p \in P, j \in J, (l,p) \in E. \quad (9)$$

The objective function (1) minimizes the sum of the three cost components previously described. Constraints (2) are the path coverage constraints of *FIFLP* formulation. In particular they impose that each path (commodity) has to be intercepted by (assigned to) a single facility located in a node belonging to the path, i.e. each final customer has to be assigned to just one selected intermediate facility. Constraints (3) are the capacity constraints for the intermediate facilities. Constraints (4) and (5) impose that at most n_j vehicles can leave and return to facility *j*, if the facility *j* is open. Constraints (6) and (7) impose that each destination *p* assigned to facility *j* must be serviced by a single vehicle starting its route at facility *j*. Constraints (8) are generalized sub-tour inequalities, *GSI*. Finally, Constraints (9) impose that all the variables of the model are binary.

It is easy to observe that exploiting the one-to-one correspondence among commodities, paths and final customers, the usage of this formulation allows to effectively take into account the multi-commodity flows to be distributed, the assignment costs and the *CL* issues, without a significant increase in the number of variables. Moreover, if we assume that all the elements of the *PN* matrix are equal to *1*, hence, not taking into account the pre-defined paths, then each commodity can be assigned to any of the potential facility locations, as generally assumed in classical *LRP* formulations. This transforms the proposed *FIFLOR* formulation into the multi-commodity version of a classical *LRP* formulation using the point based demand approach.

To conclude this section we highlight that the *FIFLOR* formulation can be easily strengthened by the following additional constraints:

$$z_{pj} \leq y_j, \qquad \forall p \in P, j \in J_p, \tag{10}$$

$$\sum_{j \in J} n_j y_j \geq \left\lceil \sum_{p \in P} \frac{f_p}{C} \right\rceil, \tag{11}$$

where Inequalities (10) are variable upper bound constraints, whereas Constraint (11) imposes a lower bound on the minimum number of vehicles to be used in the routing.

4. Solving the *FIFLOR*: Branch-and-Cut Algorithm and an Upper Bound Heuristic

In this section, we describe the branch-and-cut (B&C) algorithm developed for the *FIFLOR* problem. Then we present a heuristic for the computation of an upper bound (*UB*), based on a rounding procedure of the solution of the *FIFLOR* model linear relaxation.

4.1 B&C algorithm

The proposed B&C algorithm is based on a row generation procedure for the Generalized subtour inequalities (GSI), to find a lower bound at each node of the enumeration tree, and on the separation of several cuts derived and adapted from the literature: Lifted cover inequalities (LCI); Aggregated generalized subtour elimination constraints (AGSI); and R-cut constraints (R-CUT).
All these cuts are introduced at each node of the branch-and-bound tree and their separation has been embedded into the B&C framework provided by the ILOG CPLEX Callable Library 12.6.

Generalized subtour inequalities (GSI). As widely known the number of subtour elimination constraints used in the *FIFLOR* formulation increases exponentially with the size of the problem under investigation and for this reason a separation procedure is required. To this end we used the max-flow min-cut based separation.

It is also important to note that, for each located intermediate facility, we have an orienteering problem (*OP*), since one or more customers can remain unserviced. Hence *OP* generalized subtour elimination constraints (GSEC) used by Fischetti et al (1998) for the *OP* could also be used. We show here that the GSIs of the *FIFLOR* formulation are equivalent to the GSEC. Let us write the GSEC using the notation previously introduced:

$$\sum_{l,p \in S} x_{lp}^j \leq \sum_{p \in S} z_{pj} - z_{kj}, \qquad \forall j \in J, S \subset P, k \in S. \tag{12}$$

Moreover the GSI can be re-written as follows:

$$\sum_{l,p \in S} x_{lp}^j \leq \sum_{p \in S} z_{pj} - \left\lceil \frac{\sum_{p \in P} f_p}{C} \right\rceil, \qquad \forall j \in J, S \subset P, k \in S. \tag{13}$$

When a single vehicle has to be used, as it occurs in the *OP*, the equivalence between the two set of constraints is straightforward given that the condition $\left\lceil \frac{\sum_{p \in P} f_p \cdot}{C} \right\rceil \leq 1$ is verified.

Lifted cover inequalities. The capacity Constraints (10) added to strengthen the *FIFLOR* formulation, using to the complementary variables $\gamma_j = 1 - y_j, \forall j \in J$, can be re-written as:

$$\sum_{j \in J} n_j \gamma_j \leq \beta, \tag{14}$$

where $\beta = \sum_{j \in J} n_j - \left\lceil \frac{\sum_{p \in P} f_p}{C} \right\rceil$. Given a fractional solution to the *FIFLOR* model, indicated as $(\bar{x}, \bar{y}, \bar{z})$, and a cover $C \subseteq J$ (given by $j \in J$ such that $\overline{\gamma_j} = 0$), then the widely used lifted cover inequalities (LCIs) can be used:

$$\sum_{j \in C} \overline{\gamma_j} + \sum_{j \in J \setminus C} \alpha_j \overline{\gamma_j} \leq |C| - 1. \tag{15}$$

Each knapsack sub-problem is solved by CPLEX and variables are used following the increasing index sequence.

Aggregated GSI. Shrinking all the potential intermediate facility location nodes into a single dummy node, we obtain the following aggregated generalized subtour inequalities (AGSI):

$$\sum_{j \in J} \sum_{l \in P \setminus S} \sum_{p \in S} x_{lp}^{j} \geq \left\lceil \sum_{j \in J} \sum_{p \in S} \frac{f_p}{C} \right\rceil, \qquad \forall S \subset P. \tag{16}$$

Indeed, given the fact that no customer can remain unserviced, then $\sum_{j \in J} \sum_{p \in S} z_{pj} = 1$, and consequently the right hand side of the constraint can be rounded-up. Given a fractional solution to the *FIFLOR* model, $(\bar{x}, \bar{y}, \bar{z})$, the exact separation of these constraints can be performed using the following ILP model:

$$SEP\ 1\colon \min \sum_{j \in J} \sum_{l \in P \setminus S} \sum_{p \in S} \overline{x_{il}^{J}}\, x_{il} - RHS \tag{17}$$

$$s.t. \tag{18}$$

$$x_{ij} \geq \zeta_i, \qquad \forall i \in P, j \in J, \tag{19}$$

$$x_{ij} \geq \zeta_l - \zeta_i, \qquad \forall i, l \in P, \tag{20}$$

$$RHS \leq \frac{\sum_{p \in P} f_p}{C} + (1 - \varepsilon), \tag{21}$$

$$RHS \geq 0\ (integer), \zeta_i \in \{0,1\}, \qquad \forall i \in P. \tag{22}$$

The usage of the exact separation for these constraints is due to the round-up of the right hand side of constraints (16). Indeed, if we relax the rouding-up, then the AGSI can be obtained as a sum of the GSI over the set of the potential facilities, as done by Adulyasak et al. (2013). A similar, but quadratic separation, is proposed by Avella et al. (2014).

R-cuts constraints. Reachability cut constraints (R-cuts) have been defined by Lysgaard (2006) and their lifting, based on the relaxation of the pairwise conflicting nodes, has been proposed by Avella et al. (2013). Starting from the last ones, we used two types of *R*-cuts, classifying them as R1-cuts and R2-cuts, depending on the number of nodes involved.

Let *C(j, S)* and $C(S_1, S_2)$ be the set of the arcs belonging to the cut between the facility *j* and the node subset *S* and the set of the arcs belonging to the cut between two node subsets, S_1 and S_2, respectively. Moreover, indicating by *T* a generic cut, let *X(T)* be the sum of the x_{il}^{j} variables belonging to the cut. Then the following inequalities are valid for the FIFLOR problem:

R1 – cuts

$$X(C(j,S)) \geq z_{pj}, \qquad \forall S \subseteq P, p \in S, j \in J. \tag{23}$$

R2 – cuts

$$X(C(j,S_1 \cup S_2) \cup C(S_1,S_2) \cup C(S_2 \cup S_1)) \geq z_{ij} + z_{lj}, \qquad \forall S_1,S_2 \subseteq P, \; i \in S_1, l \in S_2, j \in J. \tag{24}$$

The explanation of R1- cuts is straightforward. Indeed they impose that, if a subset of nodes S is assigned to a facility j, then there must be at least one vehicle serving it.

Concerning instead R2-cuts, for the sake of the simplicity, we explain these inequalities referring to the graphical representation displayed in Figure 2.

Given a set of customers $S = S_1 \cup S_2$, assigned to a facility j^*, they consider the following three node cuts: the first with origin in the facility node and destinations in sets S_1 and S_2; the second with origin in S_1 and destination in S_2; the third with origin in S_2 and the destination in S_1. These cuts, represented by dotted black lines are indicated as T_1, T_2 and T_3 respectively, in Figure 2. On this basis, Inequalities (24) impose that either the customers of S_1 and S_2 are serviced by two different vehicles dispatched from facility j^* (Figure 2a) or the customers are serviced by a single vehicle (Figure 2b) dispatched from the same facility. In other words, interpreting these inequalities in terms of vehicle flow variables x^j_{il}, they impose that the total flow traversing the three cuts has to be equal to 2, either because two unit of flows leave the facility j^*or because one unit of flow leaves facility $j^*$to reach consecutively the two subsets S_1 and S_2. The routes performed by the vehicles in the two cases are represented by thick black arrows in Figure 2.

The separation of R1-cuts and R2-cuts can be obtained just applying the max-flow min-cut separation to the above described node cuts.

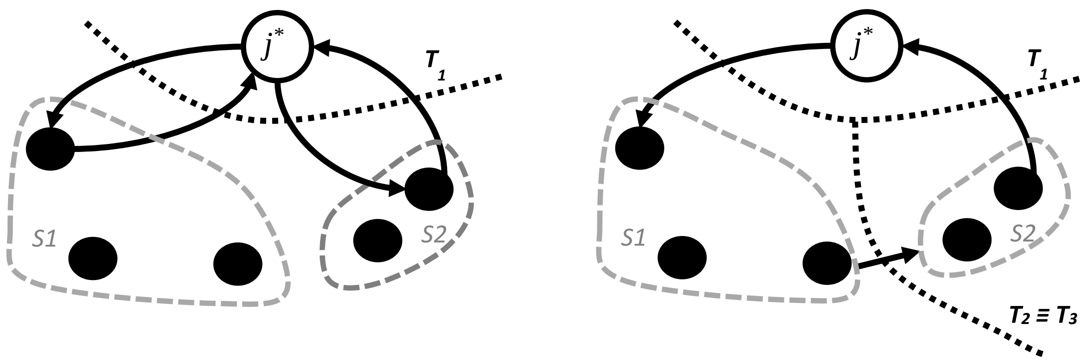


Figure 2. Representation of the R2-cuts: a) two different vehicles serve the customer subsets S_1 and S_2; b) one vehicle serves consecutively the customer subsets S_1 and S_2.

4.2 Heuristic for the *UB* computation

The heuristic for the *UB* computation is based on three steps:

Step 1. Solution of the linear relaxation of *FIFLOR* formulation;

Step 2. Commodity-to-facility assignment;

Step 3. Unfeasibility management.

The solution of the *FIFLOR* formulation relaxation, indicated in the following as $(\widehat{x^J_{il}}, \widehat{y_J}, \widehat{z_{pJ}})$, is obtained by an ILP solver.

The commodity-to-facility assignment is performed by sequentially rouding-up the $\widehat{z_{pJ}}$ variables. In particular the paths, i.e., the commodities, are sorted using the following three criteria:

1 Decreasing values of the $\widehat{z_{pj}}$ values for each, j, $j \in J_p^*$, with J_p^* the set of open facilities that can intercept the path p, i.e., the ones characterized by $\widehat{z_{pj}} > 0$. This pushes towards the assignment of a commodity p to the facility j with the highest $\widehat{z_{pj}}$ value. Hence, it tries to reproduce the solution of the linear relaxation.

2 Increasing values of the number of open facilities that can intercept a path. This attempts to push towards a solution where the assignment of the paths intercepted by the highest number of located facilities is performed lastly. This, as explained later, should guarantee that the paths that can be intercepted by more than one open facility can be re-assigned without additional location costs, when the solution would be unfeasible otherwise due to capacity constraints.

3 Decreasing values of the demands f_p, $p \in P$, to push towards solutions where we first assign the highest demand paths to the open facilities, and then we consider the lowest demand paths. It is intuitive that, in this way, it should be easier to satisfy facility and vehicle capacity constraints.

Once this sorting has been performed, at first we try to assign each path p, $p \in P$, to the facility associated with the highest $\widehat{z_{pj}}$value. If the assignment is feasible, i.e., there is at least one vehicle with enough residual capacity, we insert the related customer d_p in a route built by the Clarke and Wright algorithm. If no vehicle of the facility with the highest $\widehat{z_{pj}}$ value has still sufficient capacity to serve the customer, then the assignment to the successive facility is attempted.

A flow diagram of the ***Step 2*** is displayed in Figure 3, where we note *PJ** the path-node matrix extracted from the *PN* matrix considering just the selected intermediate facilities, and *SP* the set of the serviced paths, i.e., the ones for which a feasible commodity-to-facility assignment can be obtained at the end of the procedure. Given the facility and the vehicle capacity constraints, it may occur, however, that a commodity cannot be assigned to any of the open facilities. In this case the commodity is inserted in the set of unserviced paths, referred to as *UP*. In order to determine an unserviced commodity-to-facility assignment, we have to open additional facilities. To determine which ones, we use a simple covering heuristic based on the *PN* matrix. This heuristic, referred to as *H2* in Boccia *et al*., 2009, minimizes the number of facilities to be opened to intercept all the flows traveling on a network. In particular, at each iteration, it opens an additional facility choosing the one intercepting the highest number of unserviced paths (i.e. the one able to serve the highest number of customers). If a feasible assignment for all the unserviced customers is determined, then ***Step 2*** is repeated in order to build the routes. If, instead, a customer cannot be serviced by any of the additional facilities, i.e., all the facilities able to intercept it have insufficient residual capacity, then the solution is unfeasible and no *UB* is obtained.

A flow diagram of the ***Step 3*** is given in Figure 4, where we note *UPN* the path-node incidence matrix related to the unserviced paths *UP*.

The proposed heuristic, obviously, can be used either each time a different solution of the *FIFLOR* linear relaxation is determined or with a given frequency with respect to the B&C algorithm iterations.

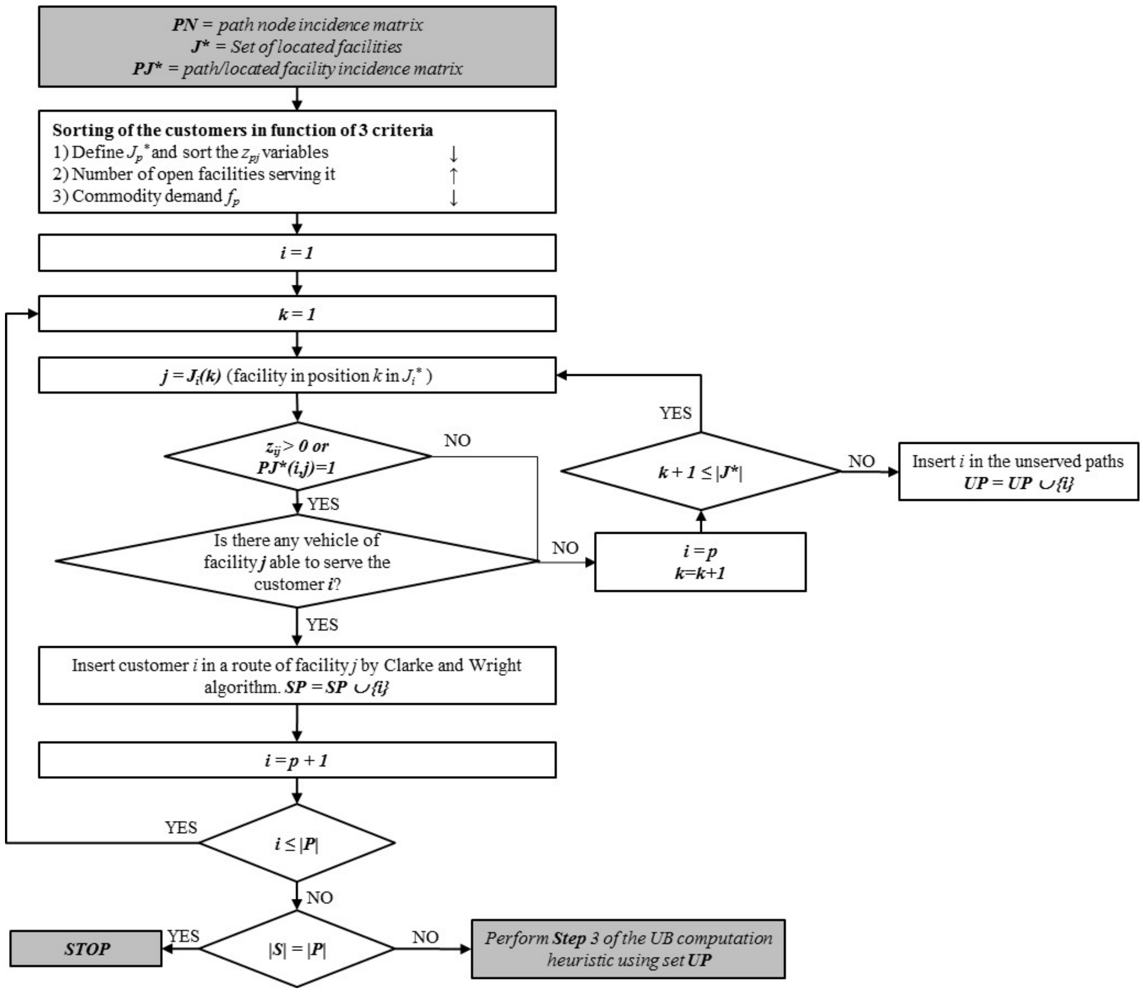


Figure 3. Flow diagram of the Step 2 of the heuristic for the UB computation

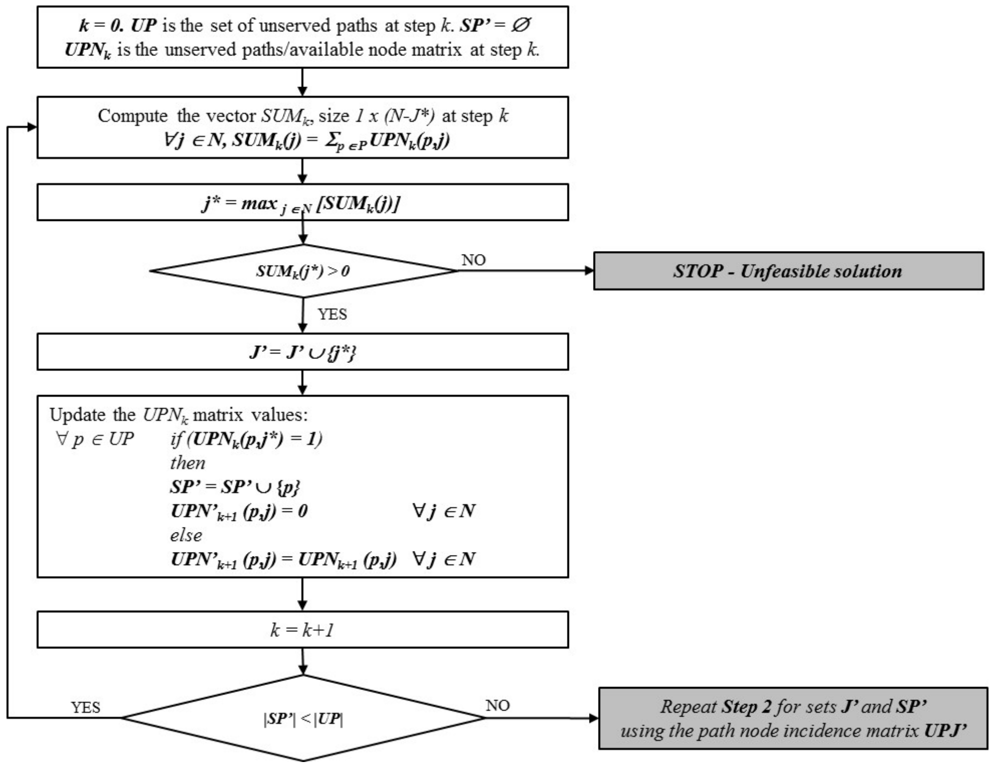


Figure 4. Flow diagram of the Step 3 of the heuristic for the UB computation.

5. Experimental Results

We report in this section the computational results of the experimentations performed to test and validate the proposed methodology. Computations were run on an Intel® Core™ i7, 870, 2.93 GHz, 4GB RAM, Windows Vista™ 64 bit. As said above, CPLEX 12.6 (default settings) was used as LP solver and to manage the branching of the proposed B&C algorithm. The maximum computation time imposed for each instance was 1 hour.

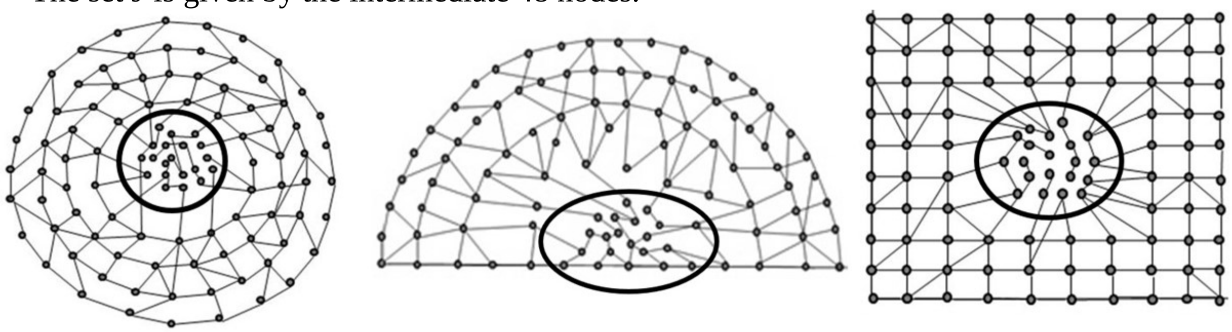
Experimentations were conducted using six sets of test instances differing in network topology, number of commodities or cost parameter setting (42 instances). Hence, in the following, we first describe the instances (*Section* 5.1). We then discuss the results we obtained (*Section* 5.2), according to a twofold perspective. The first concerns the comparison between the proposed B&C algorithm and CPLEX performance. The second focuses on the effect of the instance structure and parameters on the B&C algorithm performance.

5.1 Test instances

The proposed algorithm has been tested on randomly generated test instances. These instances are based on three different network topologies (Figure 5), which recall the structures of several European and North America urban areas:

- ***Circle network (*CN*)*** composed of 105 nodes and 346 directed links (illustrated as undirected in Figure 5a for simplicity's sake). The 25 nodes on the outmost level and the 20 nodes on the innermost level (within thick line circle) constitute the sets *O* and *D* respectively. The set *J* is given by the intermediate 60 nodes.
- ***Semi-circle network (*SCN*)*** composed of 100 nodes and 342 directed links (Figure 5b). The 25 nodes on the outmost level and the 20 nodes on the innermost level constitute the sets *O* and *D* respectively. The set *J* is given by the intermediate 60 nodes.

- ***Grid network (*GN*)*** composed of 104 nodes and 394 directed links (Figure 5c). The 36 nodes on the outmost level and the 20 nodes on the innermost level constitute the sets *O* and *D*, respectively. The set *J* is given by the intermediate 48 nodes.



(a) (b) (c)

Figure 5. The three network topologies used to generate test instances.

It is important to highlight that, as can be observed in Figure 5, the networks are composed of more than three layers. However the $3/R/\bar{T}$ structure can be easily recognized considering that the nodes of the intermediate levels are the potential facility locations and just the selected ones will constitute the second layer of the distribution system.

Seven instances were generated for each network topology, differing in the number of paths (commodities), varying from 20 to 50 (step size 5). In particular the set P of each instance has been built by randomly selecting the paths to be intercepted among all the possible o-d paths between the origin and destination nodes of the networks. Given that all instances have 20 destination nodes, the origin and destination nodes are replicated when $|P|$ is higher than $|O|$ or $|D|$, or both, in order to maintain the one-to-one relation between *O*, *D* and *P*.

The problem setting used in the generation of the instances is the following:

- d_{pj}: the value of the assignment cost has been computed as a function of the shortest path between the origin of the path *p* and the potential facility location *j*;
- c_{ij}: the value of the routing cost has been computed as a function of the Euclidean distance between node *i* to j, $(i,j) \in E$;
- *PN*: as explained above, the generic element has been set equal to *1*, not only if the node is on the *o-d* path associated to a commodity, but also if the distance between a node *i* of the path and a node *j* (potential location) not belonging to the path, is lower than or equal to a pre-fixed value Δ. The value of Δ has been properly chosen in order to have a non-sparse path node incidence matrix.

Vehicle capacity has been fixed equal to 200 and the facility capacity to 600 (i.e., the maximum number of vehicles for each facility equal to 3). Customer demands have been randomly and uniformly generated in the range 25 - 50.

Two different settings have been defined for the cost parameters, differing in the relevance of the three objective function components:

- ***Set 1 (S1): assignment cost >> location cost >> routing cost.*** With reference to the *CL* context, this set represents a situation where the facilities to be located are characterized by low installation costs. This means that we use pre-existing structures to locate facilities where no freight storage is foreseen and just cross docking operations are performed. Hence, the assignment cost represents the main cost component of the objective function.
- ***Set 2 (S2): location cost >> assignment cost >> routing cost.*** This set represents the case where the facilities to be located provide also additional freight storage/management services. Hence higher financial investments and installation costs have to be considered and the location cost represents the main component of the objective function.

In both sets, the facility location costs are not the same for all the potential locations, as they have been randomly generated within pre-fixed ranges. Moreover, we assume that the routing cost component is always the least relevant one. This choice is motivated by the fact that, as explained above, the final distribution is performed on relatively small distances (within the city center) and using environmental friendly vehicles.

In the following each instance will be identified by the following *Id*: *Instance Set_Network Topology_ # of commodities.* Hence, for example, S1_CN_20, indicates an instance of the Set *1* on the circle network topology with 20 commodities. The data of the 42 generated instances are available at http://wpage.unina.it/claudiosterle.

5.2 B&C algorithm performance

For the sake of brevity, we report a synthesis of the performed experimentation only. More detailed results can be found in the Appendix and at http://wpage.unina.it/claudiosterle. These results measure the performance of the proposed solution method in terms of number of cuts generated, at the root note and during the *B&C* algorithm, and in terms of explored branch and bound tree nodes.

The synthesis of the results obtained is displayed in Tables 1-6. We provide two tables for each network topology, Tables 1 and 4 for CN, Tables 2 and 5 for SCN, and Tables 3 and 6 for GN. Tables 1-3 deal with the comparison between the solutions obtained at the root node by CPLEX (using GSI) and the proposed B&C algorithm using all the presented cuts. Tables 4-6 concern the comparison between the solutions obtained by CPLEX (using GSI), the proposed B&C algorithm and the B&C algorithm integrated with the *UB* heuristic, referred to in the following as B&C+UB. In all cases, when the final gap is not greater than 1%, the corresponding value is highlighted in bold. In the tables, for each instance of a given set and topology, we report the lower bound (LB), the upper bound (UB), the percentage gap (GAP) and the computation time (in seconds).

We discuss the results specifically referring to each network topology and parameter setting. This makes it easier to derive not only the outcomes related to the performance of the proposed approach, but also the ones related to the effect of the network structure and parameter setting. Let us first discuss the results at the root node (Tables 1-3).

Circle Networks. Concerning the set S1, three instances (S1_CN_20, S1_CN_25 and S1_CN_30) are solved by CPLEX with a gap lower than 1%. The usage of the proposed cuts allows to optimally solve 5 instances (the previous ones and S1_CN_35 and S1_CN_40). With reference to set S2, no instance can be solved with an acceptable gap by CPLEX, whereas the usage of the proposed cuts allows to solve one instance (S2_CN_20) with a gap equal to 0.45%. All the other instances, with both methods, present high percentage gaps, but the ones obtained by the introduction of the proposed cuts are generally significantly lower than the ones of CPLEX.

	Circle Networks							
	CPLEX				B&C			
Id	**LB**	**UB**	**GAP**	**Time**	**LB**	**UB**	**GAP**	**Time**
S1_CN_20	2531.74	2538.96	**0.29**	2.60	2538.96	2538.96	**0.00**	2.67
S1_CN_25	2968.06	2970.34	**0.08**	2.62	2970.34	2970.34	**0.00**	4.78
S1_CN_30	3566.90	3571.91	**0.14**	7.59	3567.57	3567.57	**0.00**	24.48
S1_CN_35	4089.35	14837.89	262.84	8.31	4092.97	4092.97	**0.00**	39.38
S1_CN_40	4390.85	15800.48	259.85	26.17	4401.63	4401.63	**0.00**	134.55
S1_CN_45	4528.26	18766.50	314.43	35.44	4529.18	18766.50	314.35	180.36
S1_CN_50	5195.87	19306.45	271.57	43.10	5196.27	19306.45	271.54	293.67
S2_CN_20	71169.30	110441.58	55.18	8.62	71223.46	71542.99	**0.45**	44.63
S2_CN_25	84442.34	486135.32	475.70	15.76	84556.50	106639.89	26.12	49.49
S2_CN_30	97229.57	674855.17	594.08	31.91	97315.83	173372.89	78.15	251.86
S2_CN_35	110287.01	727020.13	559.21	53.61	111764.90	727020.13	550.49	356.12
S2_CN_40	115940.89	784050.34	576.25	72.37	116171.37	784050.34	574.91	955.69
S2_CN_45	121938.78	868823.53	612.51	63.40	122239.19	265006.83	116.79	1511.42
S2_CN_50	138946.68	880965.14	534.03	218.33	144179.63	880965.14	511.02	1821.67

Table 1. CPLEX and B&C algorithm results on CN topology at root node.

Semicircle Networks. Concerning set S1, no instance is optimally solved by CPLEX. Instead, the usage of the proposed cuts allows to optimally solve two instances (S1_SCN_20, S1_SCN_30). Moreover for another instance (S1_SCN_45) a gap equal to 0.52% is obtained. With reference to set S2, no instance can be solved with an acceptable gap by PLEX, whereas the usage of the proposed cuts allows to optimally solve one instance (S2_SCN_20). All the other instances, with both methods, present high percentage gap, without significant differences.

	Semicircle Networks							
	CPLEX				B&C			
Id	**LB**	**UB**	**GAP**	**Time**	**LB**	**UB**	**GAP**	**Time**
S1_SCN_20	1949.06	8566.12	339.50	3.13	1957.90	1957.90	**0.00**	5.70
S1_SCN_25	2622.24	10802.93	311.97	4.26	2630.78	10802.93	310.64	10.85
S1_SCN_30	2900.45	12476.70	330.16	5.38	2908.24	2908.24	**0.00**	41.36
S1_SCN_35	3234.99	14227.69	339.81	8.12	3239.40	14227.69	339.21	115.63
S1_SCN_40	3250.18	16091.63	395.10	26.18	3260.87	16091.63	393.48	426.14
S1_SCN_45	3817.24	16754.53	338.92	32.04	3821.50	3841.20	**0.52**	258.56
S1_SCN_50	4177.38	18724.60	348.24	49.92	4180.66	18724.60	347.89	832.60
S2_SCN_20	62809.66	119110.01	89.64	6.12	62832.49	62834.61	**0.00**	8.61
S2_SCN_25	79814.96	524725.39	557.43	8.90	79851.62	524725.39	557.13	116.04
S2_SCN_30	89303.88	602765.80	574.96	24.99	89418.86	206822.21	131.30	212.73
S2_SCN_35	100356.34	727706.23	625.12	35.50	101627.38	727706.23	616.05	495.00
S2_SCN_40	101750.04	830403.28	716.12	96.23	102530.81	830403.28	709.91	1211.49
S2_SCN_45	109137.50	816635.38	648.26	42.71	109228.01	816635.38	647.64	981.95
S2_SCN_50	117977.13	854405.67	624.21	83.94	128745.93	854405.67	563.64	669.45

Table 2. CPLEX and B&C algorithm results on SCN topology at root node

Grid Networks. Concerning set S1, four instances are optimally solved by CPLEX (S1_GN_20, S1_GN_25, S1_GN_30 and S1_GN_35). The usage of the proposed cuts allows to optimally solve two additional instances (S1_GN_40 and S1_GN_50). With reference to set S2, no instance can be solved with an acceptable gap by both methods. However, with few exceptions, the gap obtained by the introduction of the proposed cuts is significantly better compared to CPLEX.

	Grid Networks							
	CPLEX				B&C			
Id	**LB**	**UB**	**GAP**	**Time**	**LB**	**UB**	**GAP**	**Time**
S1_GN_20	2351.58	2351.58	**0.00**	1.71	2351.58	2351.58	**0.00**	2.44
S1_GN_25	2851.74	2852.13	**0.01**	3.10	2852.13	2852.13	**0.00**	2.76
S1_GN_30	2917.83	2917.83	**0.00**	4.57	2917.83	2917.83	**0.00**	2.80
S1_GN_35	3405.59	3405.59	**0.00**	5.80	3405.59	3405.59	**0.00**	6.76
S1_GN_40	3752.73	15248.96	306.34	29.53	3755.09	3755.09	**0.00**	37.07
S1_GN_45	4072.48	16848.93	313.73	39.20	4077.15	16848.93	313.25	292.41
S1_GN_50	4575.53	17384.04	279.93	43.90	4579.54	4580.24	**0.02**	266.67
S2_GN_20	73369.17	110581.55	50.72	9.60	73724.42	137254.14	86.17	4.96
S2_GN_25	84749.92	167563.18	97.71	10.14	84776.74	555516.08	555.27	30.70
S2_GN_30	91408.95	642128.83	602.48	21.84	91437.91	154285.29	68.73	181.53
S2_GN_35	108901.14	617690.66	467.20	45.19	112630.86	160846.30	42.81	248.74
S2_GN_40	112379.75	762632.39	578.62	70.38	112436.67	216865.91	92.88	758.54
S2_GN_45	122701.48	772707.15	529.75	118.71	123283.76	228858.69	85.64	1011.99
S2_GN_50	135075.72	720219.06	433.20	162.92	139999.81	720219.06	414.44	1314.75

Table 3. CPLEX and B&C algorithm results on GN topology at root node.

Let us now consider the comparison between CPLEX, B&C and B&C+UB in terms of quality of solution expressed by the optimality gap. Computation times are discussed after.

Circle Networks. CPLEX determines the optimal solution of all the instances of set S1, except S1_CN_45. This instance is optimally solved by the other two methods. With reference to set S2, CPLEX determines solutions with a gap lower than or equal to 0.55% for four instances (S2_CN_20, S2_CN_25, S2_CN_30 and S2_CN_40), but it is not able to determine an acceptable solution for the other instances. The other two methods determine the optimal solution also for instance S2_CN_35. The instance S2_CN_50 is solved with a gap equal to 0.55% just by the B&C+UB algorithm, which is also the only one able to determine an acceptable solution for S2_CN_45 (gap equal to 11.03%).

	Circle Networks											
	CPLEX				B&C				B&C+UB			
Id	**LB**	**UB**	**GAP**	**Time**	**LB**	**UB**	**GAP**	**Time**	**LB**	**UB**	**GAP**	**Time**
S1_CN_20	2538.96	2538.96	**0.00**	2.62	2538.96	2538.96	**0.00**	2.64	2538.96	2538.96	**0.00**	2.60
S1_CN_25	2970.34	2970.34	**0.00**	2.62	2970.34	2970.34	**0.00**	4.78	2970.34	2970.34	**0.00**	2.67
S1_CN_30	3567.57	3567.57	**0.00**	87.41	3567.57	3567.57	**0.00**	22.43	3567.57	3567.57	**0.00**	14.54
S1_CN_35	4091.95	4092.97	**0.02**	3600.06	4092.97	4092.97	**0.00**	40.08	4092.97	4092.97	**0.00**	40.16
S1_CN_40	4401.63	4401.63	**0.00**	1512.08	4401.63	4401.63	**0.00**	135.15	4401.63	4401.63	**0.00**	140.01
S1_CN_45	4528.74	5628.29	24.28	3600.29	4531.09	4531.09	**0.00**	2600.28	4531.09	4531.09	**0.00**	1563.35
S1_CN_50	5196.76	5196.76	**0.00**	953.20	5196.76	5196.76	**0.00**	786.31	5196.76	5196.76	**0.00**	381.08
S2_CN_20	71244.63	71244.63	**0.00**	107.26	71244.63	71244.63	**0.00**	56.51	71244.63	71244.63	**0.00**	56.43
S2_CN_25	84556.11	84577.30	**0.03**	3600.04	84570.15	84570.15	**0.00**	2230.19	84570.15	84570.15	**0.00**	1848.25
S2_CN_30	97479.66	97594.71	**0.12**	3600.06	97536.45	97543.71	**0.01**	1593.71	97536.45	97543.71	**0.01**	819.23
S2_CN_35	111014.22	727020.13	554.89	3600.11	111828.10	111843.87	**0.01**	3600.08	111843.87	111843.87	**0.00**	2698.62
S2_CN_40	115954.86	116588.82	**0.55**	3600.39	116189.45	116588.82	**0.34**	3606.32	116262.53	116375.49	**0.10**	3607.45
S2_CN_45	122448.66	868823.53	609.54	3601.01	122676.92	265006.83	116.02	3600.13	122676.92	136204.83	11.03	3609.53
S2_CN_50	139094.27	880965.14	533.36	3601.14	144405.68	880965.14	510.06	3600.05	144408.57	145201.07	**0.55**	3605.50

Table 4. Results of CPLEX, B&C and B&C + UB algorithms on CN topology.

Semicircle Networks. CPLEX determines the optimal solution of all the instances of set S1, except S1_SCN_45 (gap equal to 17.14%). This instance is optimally solved by the other two methods. With reference to set S2, CPLEX determines solutions with a gap lower than or equal to 0.44% for three instances (S2_SCN_20, S2_SCN_25 and S2_SCN_40). The other two methods optimally solve instance S2_SCN_35 and provide a solution with a gap equal to 1.74% for instance S2_SCN_30. Instance S2_SCN_45 is solved only by the B&C+UB algorithm. Instance S2_SCN_50 cannot be solved by CPLEX and B&C algorithm, whereas the B&C+UB algorithm returns a solution with a gap equal to 15.60%.

	Semicircle Networks											
	CPLEX				B&C				B&C+UB			
Id	**LB**	**UB**	**GAP**	**Time**	**LB**	**UB**	**GAP**	**Time**	**LB**	**UB**	**GAP**	**Time**
S1_SCN_20	1957.90	1957.90	**0.00**	8.70	1957.90	1957.90	**0.00**	5.61	1957.90	1957.90	**0.00**	5.66
S1_SCN_25	2638.73	2638.73	**0.00**	70.36	2638.73	2638.73	**0.00**	48.89	2638.73	2638.73	**0.00**	13.61
S1_SCN_30	2908.24	2908.24	**0.00**	10.02	2908.24	2908.24	**0.00**	41.55	2908.24	2908.24	**0.00**	42.80
S1_SCN_35	3241.78	3241.78	**0.00**	1048.09	3241.78	3241.78	**0.00**	256.55	3241.78	3.241.778	**0.00**	148.38
S1_SCN_40	3260.57	3267.63	**0.22**	3600.84	3264.83	3264.83	**0.00**	3176.71	3264.83	3264.83	**0.00**	2923.48
S1_SCN_45	3823.35	4478.83	17.14	3600.15	3829.98	3829.98	**0.00**	1226.34	3829.98	3833.74	**0.10**	1225.82
S1_SCN_50	4181.87	4182.90	**0.02**	3601.86	4182.90	4182.90	**0.00**	1816.52	4182.90	4182.90	**0.00**	1535.59
S2_SCN_20	62834.61	62834.61	**0.00**	83.31	62834.61	62834.61	**0.00**	8.18	62834.61	62834.61	**0.00**	8.11
S2_SCN_25	79858.39	79870.74	**0.02**	3600.12	79858.39	79858.39	**0.00**	239.95	79858.39	79858.39	**0.00**	115.20
S2_SCN_30	89349.66	110885.76	24.10	3600.08	89421.66	90979.64	1.74	3600.06	89423.75	90979.64	1.74	3601.23
S2_SCN_35	101159.12	123196.02	21.78	3600.03	101657.70	101661.88	**0.00**	3600.04	101661.88	101661.88	**0.00**	3163.02
S2_SCN_40	102295.80	102741.44	**0.44**	3600.39	102531.80	102544.09	**0.01**	3603.26	102544.09	102544.09	**0.00**	3515.35
S2_SCN_45	109142.28	816635.38	648.23	3600.41	109228.01	816635.38	647.64	3600.02	102531.80	102544.09	**0.01**	3600.82
S2_SCN_50	118349.19	854405.67	621.94	3600.30	128747.65	854405.67	563.63	3680.40	128747.16	148830.64	15.60	3600.16

Table 5. Results of CPLEX, B&C and B&C + UB algorithms on SCN topology.

Grid Networks. All the instances of set S1 are optimally solved by CPLEX and the proposed algorithms. With reference to set S2, CPLEX determines solutions with a gap lower than or equal to 0.28% for three instances (S2_GN_20, S2_GN_25 and S2_GN_30), but it is not able to determine an acceptable solution for the other instances. Also the other two algorithms cannot solve these instances either. The B&C+UB algorithm is able, however, to determine solutions with the following gaps: 4.60%, 20.59%, 15.96% and 12.73% for S2_GN_35, S2_GN_40, S2_GN_45 and S2_GN_50, respectively.

	Grid Networks											
	CPLEX				B&C				B&C+UB			
Id	**LB**	**UB**	**GAP**	**Time**	**LB**	**UB**	**GAP**	**Time**	**LB**	**UB**	**GAP**	**Time**
S1_GN_20	2351.58	2351.58	**0.00**	3.98	2351.58	2351.58	**0.00**	2.67	2351.58	2351.58	**0.00**	3.11
S1_GN_25	2852.13	2852.13	**0.00**	4.55	2852.13	2852.13	**0.00**	2.18	2852.13	2852.13	**0.00**	1.26
S1_GN_30	2917.83	2917.83	**0.00**	4.08	2917.83	2917.83	**0.00**	2.81	2917.83	2917.83	**0.00**	2.88
S1_GN_35	3405.59	3405.59	**0.00**	6.63	3405.59	3405.59	**0.00**	8.02	3405.59	3.405.593	**0.00**	3.44
S1_GN_40	3755.09	3755.09	**0.00**	3109.48	3755.09	3755.09	**0.00**	54.72	3755.09	3755.09	**0.00**	56.17
S1_GN_45	4076.99	4077.97	**0.02**	3600.28	4077.97	4077.97	**0.00**	538.98	4077.97	4077.97	**0.00**	214.52
S1_GN_50	4578.23	4580.24	**0.04**	3601.53	4580.24	4580.24	**0.00**	367.44	4580.24	4580.24	**0.00**	367.16
S2_GN_20	73650.71	73734.69	**0.11**	3600.01	73734.69	73734.69	**0.00**	18.16	73734.69	73734.69	**0.00**	20.84
S2_GN_25	84764.97	85001.69	**0.28**	3600.37	84785.43	84785.43	**0.00**	63.45	84785.43	84785.43	**0.00**	63.39
S2_GN_30	91425.75	91457.30	**0.03**	3601.06	91448.29	91448.29	**0.00**	349.46	91448.29	91448.29	**0.00**	349.42
S2_GN_35	109683.38	157241.81	43.36	3607.75	112789.67	139051.47	23.28	3600.09	112788.97	117973.8	4.60	3600.04
S2_GN_40	112398.15	762632.39	578.51	3600.17	112438.56	216865.91	92.88	3600.67	112438.56	135586.25	20.59	3600.79
S2_GN_45	122826.11	772707.15	529.11	3600.30	123354.85	228858.69	85.53	3600.01	123386.83	143084.03	15.96	3600.66
S2_GN_50	135132.73	720219.06	432.97	3600.06	140810.22	267396.36	89.90	3600.43	140983.65	158934.41	12.73	3600.43

Table 6. Results of CPLEX, B&C and B&C + UB algorithms on GN topology.

For all the network topologies and for both sets, the computation time of CPLEX are significantly higher than the ones of the other two methods and, in most cases, equal to the maximum imposed computation time. The computation time of the B&C+UB algorithm is generally lower than the one of the B&C algorithm. Finally, we note that in all the cases where the B&C and the B&C+UB algorithms reach the maximum computation time, they outperform CPLEX in terms of solution quality

The main conclusions of our experimentation can then be summarized as follows:

1. CPLEX (using GSI) is able to solve 29 *FIFLOR* instances out of 42, with a maximum gap of 0.55% and an average computation time of 1855.91 seconds. For 14 instances these solutions are obtained after 1 hour of computation time. The proposed B&C is able to solve 33 instances with a gap not greater than 0.34% and an average computation time of 912.42 seconds. The B&C+UB algorithm solves 35 instances obtaining a gap not greater than 0.55%, due to the gap of the two additional solved instances. The average computation time on the same 33 instances solved by the B&C algorithm is 756.16 seconds. The two additional instances are solved after 1 hour computation time. Let us consider now the unsolved instances. For all of them, only B&C+UB yields a solution with gaps of 11.03% (S2_CN_45); 1.74% (S2_SCN_25); 15.60% (S2_SCN_50); 4.60% (S2_GN_35); 20.59% (S2_GN_40); 15.96% (S2_GN_45); 12.73% (S2_GN_50).
2. The usage of the *UB* heuristic is fundamental, as CPLEX is not able to determine a good feasible solution for the most difficult instances.
3. All the set *S1* instances were solved near optimality within the imposed time limit. The same does not occur for the *S2* instances, which seem to be more difficult to solve. This could be explained by the low quality of the *UB* obtained when location costs are significantly higher than the other two cost components. The B&C+UB algorithm mitigates this effect and is able to provide solutions for additional instances. This is particularly evident considering the set *S2* instances on the GN topology, where the low quality of the *UB* could also be explained by the structure of the network, allowing many symmetric solutions in terms of assignment costs.
4. *AGSI* cuts are the most effective ones. By contrast, *R-cuts* are not very effective in terms of quality of solution, since they generally allow to achieve slight improvements of the *LB* values only. However, when applied, they provide a great reduction in the number of the branch-and-bound explored nodes. Moreover, they allow to find other violated *GSI* and *AGSI* in several instances.

6. Conclusions and Perspectives

We defined a new multi-commodity *LRP* and proposed an original ILP formulation based on a flow-intercepting approach. The problem is referred to as flow intercepting facility location routing problem (*FIFLOR*) since it merges and tackles simultaneously the flow intercepting facility location and the vehicle routing problems. The problem has been addressed by a branch-and-cut algorithm exploiting valid inequalities derived and adapted from the literature.

This work can represent an important starting point for further developments for solution methods for multi-commodity *LRP*s. Indeed, using the flow intercepting approach leads to the development of path generation-based procedures in order to deal with large size instances involving a great number of commodities, or integrating additional practical constraints.

FIFLOR can be applied to various City Logistics settings. Indeed it allows to easily and effectively take into account the strategic and tactical decisions arising in the design of a *single-tier* freight distribution system. It is for this reason that we experimented and validated the proposed approach on network instances with different topologies and parameter settings, built with the aim of representing different urban area structures and scenarios.

Three extensions deserve to be further investigated. The first concerns the design of two-tier freight distribution systems (Crainic et al., 2009). The second concerns the adaptation of *FIFLOR* to tackle the park and ride facility location problem, discussed in Horner and Groves (2007). In this case, great attention should be given to the individual paths of the network users which may significantly change following the selected facility locations. Finally, the last extension could concern the usage of *FIFLOR* to approach the hub-location problem (Nagy and Salhi, 1998 and Rieck et al. 2014), which is a many-to-many location routing problem and hence it is inherently multi-commodity.

Acknowledgments

Partial funding for this project has been provided by the Natural Sciences and Engineering Council of Canada (NSERC) through its Discovery Grants program. We also gratefully acknowledge the support of the Fonds de recherche du Québec through their infrastructure grants. While working on this project, the second author was Adjunct Professor with the Department of Computer Science and Operations Research, Université de Montréal.
The research activity of the third and forth author was partially funded by the Department of Electrical Engineering and Information Technology of University "Federico II" of Naples, within the research project on "Optimization Models and Methods for ITS".

References

[1] Adulyasak, Y., Cordeau, J.F., Jans, R.. Formulations and branch-and-cut algorithms for multi-vehicle production and inventory routing problems. *INFORMS Journal on Computiing*, 26 (1), 103-120, (2013).

[2] Avella, P., Boccia, M., Vasilyev, I.. Lifted and local reachability cuts for the vehicle routing problem with time windows. *Computers and Operations Research*, 40 (8), 2004-2010, (2013).

[3] Avella, P., Boccia, M., Wolsey, L.A.. Single-item reformulations for a vendor managed inventory routing problem: computational experience with benchmark instances. *Networks*, 10.1002/net.21586, 2014.

[4] Bektas, T., Crainic T.G., Woensel, T.. From managing urban freight to smart City Logistics networks. In Pardalos P. and Gakis K. (ed.), *Networks Design and Optimization for Smart Cities*. World Scientific Publishing, (2016), to appear.

[5] Berman, O., Larson, R.C., Fouska, N.. Optimal locating of discretionary facilities. *Transportation Science*, 26 (3), 201–211, (1992).

[6] Berman, O., Hodgson, M.J., Krass, D.. Flow interception problems. In Drezner Z. (ed.), *Facility Location: A survey of Applications and Methods*. Springer New York: 389 – 426, (1995).

[7] Boccia, M., Crainic, T., Sforza, A., Sterle, C.. Location-routing models for designing a two-echelon freight distribution system. *Technical report CIRRELT-2011-06*. Université de Montréal, (2011).

[8] Boccia, M., Crainic, T.G., Sforza, A., Sterle, C.. A metaheuristic for two-echelon location-routing problem. Experimental Algorithms. *Lecture Notes in Computer Science*, 6049, 288-301, (2010).

[9] Boccia, M., Sforza, A., Sterle, C.. Flow Intercepting Facility Location: Problems, Models and Heuristics. *Journal of Mathematical Modelling and Algorithms*, 8 (1), 35-79, (2009).

[10] Burks, R.. An adaptive tabu search heuristic for the location-routing pickup and delivery problem with time windows with a theater distribution application. *PhD thesis*, Graduate School of Engineering and Management, Air Force Institute of Technology, Ohio, (2006).

[11] Crainic, T.G., Ricciardi, N., Storchi, G.. Advanced freight transportation systems for congested urban areas. *Transportation Research Part C: Emerging Technologies*, 12 (2), 119-137, (2004).

[12] Crainic, T.G., Ricciardi, N., Storchi, G.. Models for evaluating and planning city logistics systems. *Transportation Science*, 43 (4), 432-454, (2009).

[13] Cuda, R., Guastaroba, G., Spranza, M.G.. A survey on two-echelon routing problems. *Computers & Operations Research*, 55, 185-199, (2015).

[14] De Cillis, F., Sforza, A.,Sterle, C. Optimal location of flow intercepting facilities to improve security in urban areas. *International Journal of System of Systems Engineering*, 4(3-4), 222–242, (2013).

[15] Dial, R.B.. A path-based user-equilibrium traffic assignment algorithm that obviates path storage and enumeration. *Transportation Research Part B: Methodological*, 40 (10), 917–936, (2006).

[16] Drexl, M., Schneider, M.. A survey of variants and extensions of the location-routing problem. *European Journal of Operational Research*, 241 (2), 283-308, (2015).

[17] Fischetti, M., Salazar-Gonzalez, J. J., Toth., P.. Solving the orienteering problem through branch-and-cut. *INFORMS Journal on Computing*, 10 (2), 133-148, (1998).

[18] Gendreau, M., Laporte, G., Parent, I. Heuristics for the location of inspection station on a network. *Naval Research Logistics*, 47 (4), 287–303, (2000).

[19] Gentili, M., Mirchandani, P.B.. Locating active sensors on traffic networks. *Annals of Operations Research*, 35 (1), 229–257, (2005).

[20] Hamidi, M., Farahamand, K., Sajjadi, S.. Modeling a four-layer location routing problem. *International Journal of Industrial Engineering Computations*, 3, 43-52, (2012)

[21] Hamidi, M., Farahamand, K., Sajjadi,S., Nygard, K.. A heuristic algorithm for a multi-product four-layer capacitated location-routing problem. *International Journal of Industrial Engineering Computations*, 5 (1), 87–100, (2014).

[22] Hodgson, M.J.. The location of public facilities intermediate to the journey to work. *European Journal of Operations Research*, 6 (2), 199–204, (1981).

[23] Horner, M.W., Groves, S.. Network flow-based strategies for identifying park-and-ride facility locations. *Socio-economic Planning Sciences*, 41 (3), 255-268, (2007).

[24] Kim, J.G., Kuby, M.. A network transformation heuristic approach for the deviation flow refueling location model. *Computer & Operations Research*, 40 (4), 112–1131, (2013).

[25] Laporte, G.. Location-routing problems. In B.L. Golden & A.A. Assad (Eds), *Vehicle routing: Methods and studies*. Amsterdam: North-Holland, (1988).

[26] Lysgaard, J.. Reachability cuts for the vehicle routing problem with time windows. *European Journal of Operational Research*, 175 (1), 210–223, (2006).

[27] Maranzana, F.E.. On the location of supply points to minimize transport costs. *Operational Research Quarterly*, 15, 261-270, (1964).

[28] Mancini, S., Gonzalez-Feliu, J., Crainic, T.G.. Planning and optimization methods for advanced urban logistics systems at tactical level. In K. Gonzalez-Feliu, F. Semet, J.L. Routhier (Eds.). *Sustainable Urban Logistics: Concepts, Methods and Information Systems*. Springer, Berlin Heidelberg, 145-164, (2014).

[29] Morana J.. Sustainable supply chain management in urban logistics. In K. Gonzalez-Feliu, F. Semet, J.L. Routhier (Eds.). *Sustainable Urban Logistics: Concepts, Methods and Information Systems*. Springer, Berlin Heidelberg, 21-35, (2014).

[30] Nagy, G., Salhi, S.. The many-to-many location-routing problem. *TOP*, 6, 261–275, (1998).

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

[32] Perl, J., Daskin, M.S.. A warehouse location-routing problem. *Transportation Research Part B*, 19 (5), 381-396, (1985).

[33] Prodhon, C., Prins, C.. A survey of recent research on location-routing problems. *European Journal of Operational Research*, 238 (1), 1-17, (2014).

[34] Rahmani, Y., Ramdane Cherif, W., Oulamara, A.. A Local Search approach for the Two–Echelon Multi-products Location–Routing problem with Pickup and Delivery. *IFAC-Papers On Line*, 48 (3), 193–199, (2015a).

[35] Rahmani, Y., Ramdane Cherif, W., Oulamara, A.. The two-echelon multi-products location-routing problem with pickup and delivery: formulation and heuristic approaches. *International Journal of Production Research*, DOI: 10.1090/00207543.2015.1040127, (2015b).

[36] Rath, S., Gutjahr, W.J.. A math-heuristic for the warehouse location-routing problem in disaster relief. *Computers & Operations Research*, 42, 25-39, (2014).

[37] Rieck, J., Ehrenberg, C., Zimmermann, J.. Many-to-many location-routing with inter-hub transport and multi-commodity pickup-and-delivery. *European Journal of Operational Research*, 236, 863–878, (2014).

[38] Selmic, M., Teodorovic, D., Vukadinovic, K.. Locating inspection facilities in traffic networks: an artificial intelligence approach. *Transportation Planning and Technology*, 33 (6), 481–93, (2010).

[39] Sterle., C., Sforza, A., Amideo, A.E.. Multi-period location of flow intercepting portable facilities of an intelligent transportation system. *Socio-Economic Planning Sciences*, 10.1016/j.seps.2016.01.001, (2016)

[40] Taniguchi, E., Norikate, M., Yamada, T., Izumitani, T.. Optimal size and location planning of public logistic terminals. *Transportation Research Part E: Logistics and Transportation Review*, 35 (3), 207-222, (1999).

[41] Upchurch, C., Kuby, M., Lim, S.. A capacitated model for location of alternative-fuel stations. *Geographical Analysis*, 41 (1), 85–106, 2009.

[42] Wen, M., Laporte, G., Madsen, O.B.G., Norrelund, A.V., Olsen, A.. Locating replenishment stations for electric vehicles: application to Danish traffic data. *Journal of the Operational Research Society*, 65, 1555–1561, (2014).

[43] Yang, H., Yang, C., Liping, G.. Models and algorithms for the screen line-based traffic-counting location problems. *Computer & Operations Research*, 33 (3), 836–858, (2006).

[44] Zeng, W., Castillo, I., Hodgson, M.J.. A generalized model for locating facilities on a network with flow-based demand. *Network and Spatial Economics*, 10 (4), 579–611, (2010).

Appendix

The results of the experimentation are summarized in Tables A1-A6. For each network topology and for each set, the following information is reported for each instance (identified by its *Id* in the first column):

- Columns 2 to 5 are related to the usage of the additional inequalities, cuts and UB heuristic. In particular, in Columns 2 to 4 we indicate if *VUB* and *CC*, *AGSI* and *R-Cuts*, respectively, are used (value 1) or not (value 0). We indicate in Column 5 the frequency of using the heuristic (set to 5);
- Columns 6 to 9 report the lower bound (*LB*) and the upper bound (*UB*) obtained, the related percentage gap (*GAP*) and computation time (*Time*). Percentage gap values lower than 1% are highlighted in bold.
- Columns 10 to 13 report the number of cuts (*GSI*, *AGSI*, *R-cuts*) added and the number of branch-and-bound tree nodes explored by the algorithm (*# Nodes*).

Note that, for each instance, the rows from 1 to 4 are related to the results obtained at the root node (indicated as 1 in the column *# Nodes*), whereas the rows from 5 to 7 are related to the results obtained within 1 hour computation time. In particular Row 5 is related to the solution obtained by CPLEX 12.6 using only the *GSI*; Row 6 is related to the solution obtained by the proposed B&C algorithm; Row 7 is related to the solution obtained by the proposed B&C algorithm integrated with the *UB* heuristic.

Circle Networks - Set 1

Id	VUB & CC	AGSI	RCUTS	HEUR	LB	UB	GAP	Time	#GSI	#AGSI	#RCUTS	# Nodes
S1_CN_20	0	0	0	0	2531.74	2538.96	**0.29**	2.60	62	0	0	1
	1	0	0	0	2538.96	2538.96	**0.00**	2.06	28	0	0	1
	1	1	0	0	2538.96	2538.96	**0.00**	2.20	28	0	0	1
	1	1	1	0	2538.96	2538.96	**0.00**	2.67	28	0	0	1
	0	0	0	0	2538.96	2538.96	**0.00**	2.62	62	0	0	1
	1	1	1	0	2538.96	2538.96	**0.00**	2.64	28	0	0	1
	1	1	1	5	2538.96	2538.96	**0.00**	2.60	28	0	0	1
S1_CN_25	0	0	0	0	2968.06	2970.34	**0.08**	2.62	61	0	0	1
	1	0	0	0	2970.34	2970.34	**0.00**	3.50	24	0	0	1
	1	1	0	0	2970.34	2970.34	**0.00**	3.61	24	1	0	1
	1	1	1	0	2970.34	2970.34	**0.00**	4.78	24	1	0	1
	0	0	0	0	2970.34	2970.34	**0.00**	2.62	61	0	0	1
	1	1	1	0	2970.34	2970.34	**0.00**	4.78	24	1	0	1
	1	1	1	5	2970.34	2970.34	**0.00**	2.67	22	0	0	1
S1_CN_30	0	0	0	0	3566.90	3571.91	**0.14**	7.59	85	0	0	1
	1	0	0	0	3565.63	13156.31	268.98	9.11	40	0	0	1
	1	1	0	0	3567.57	3567.57	**0.00**	18.01	65	31	0	1
	1	1	1	0	3567.57	3567.57	**0.00**	24.48	63	28	35	1
	0	0	0	0	3567.57	3567.57	**0.00**	87.41	95	0	0	76
	1	1	1	0	3567.57	3567.57	**0.00**	22.43	67	34	51	14
	1	1	1	5	3567.57	3567.57	**0.00**	14.54	67	0	0	7
S1_CN_35	0	0	0	0	4089.35	14837.89	262.84	8.31	100	0	0	1
	1	0	0	0	4088.46	14837.89	262.92	9.59	55	0	0	1
	1	1	0	0	4092.97	4092.97	**0.00**	35.38	79	48	0	1
	1	1	1	0	4092.97	4092.97	**0.00**	39.38	79	48	0	1
	0	0	0	0	4091.95	4092.97	**0.02**	3600.06	1895	0	0	117578
	1	1	1	0	4092.97	4092.97	**0.00**	40.08	79	48	0	1
	1	1	1	5	4092.97	4092.969	**0.00**	40.16	79	48	0	1
S1_CN_40	0	0	0	0	4390.85	15800.48	259.85	26.17	84	0	0	1
	1	0	0	0	4393.25	15800.48	259.65	23.64	53	0	0	1
	1	1	0	0	4401.63	4401.63	**0.00**	128.98	60	44	0	1
	1	1	1	0	4401.63	4401.63	**0.00**	134.55	60	44	0	1
	0	0	0	0	4401.63	4401.63	**0.00**	1512.08	152	0	0	1920
	1	1	1	0	4401.63	4401.63	**0.00**	135.15	60	44	0	1
	1	1	1	5	4401.63	4401.63	**0.00**	140.01	60	44	0	1
S1_CN_45	0	0	0	0	4528.26	18766.50	314.43	35.44	89	0	0	1
	1	0	0	0	4528.42	18766.50	314.42	23.10	46	0	0	1
	1	1	0	0	4529.18	18766.50	314.35	164.10	56	60	0	1
	1	1	1	0	4529.18	18766.50	314.35	180.36	56	60	0	1
	0	0	0	0	4528.74	5628.29	24.28	3600.29	1496	0	0	3141
	1	1	1	0	4531.09	4531.09	**0.00**	2600.28	274	150	0	2092
	1	1	1	5	4531.09	4531.09	**0.00**	1563.35	253	143	40	1265
S1_CN_50	0	0	0	0	5195.87	19306.45	271.57	43.10	72	0	0	1
	1	0	0	0	5195.28	19306.45	271.62	37.40	50	0	0	1
	1	1	0	0	5196.24	19306.45	271.55	264.82	68	36	0	1
	1	1	1	0	5196.27	19306.45	271.54	293.67	67	36	9	1
	0	0	0	0	5196.76	5196.76	**0.00**	953.20	187	0	0	539
	1	1	1	0	5196.76	5196.76	**0.00**	786.31	70	36	9	23
	1	1	1	5	5196.76	5196.76	**0.00**	381.08	70	36	9	23

Table A.1. Results on circle network topology with Set S1.

Circle Networks - Set 2

Id	VUB & CC	AGSI	RCUTS	HEUR	LB	UB	GAP	Time	#GSI	#AGSI	#RCUTS	# Nodes
S2_CN_20	0	0	0	0	71169.30	110441.58	55.18	8.62	169	0	0	1
	1	0	0	0	71168.90	154750.32	117.44	4.07	40	0	0	1
	1	1	0	0	71220.36	71462.97	**0.34**	17.10	93	38	0	1
	1	1	1	0	71223.46	71542.99	**0.45**	44.63	99	48	516	1
	0	0	0	0	71244.63	71244.63	**0.00**	107.26	326	0	0	978
	1	1	1	0	71244.63	71244.63	**0.00**	56.51	125	150	516	11
	1	1	1	5	71244.63	71244.63	**0.00**	56.43	125	150	516	11
S2_CN_25	0	0	0	0	84442.34	486135.32	475.70	15.76	260	0	0	1
	1	0	0	0	84539.70	106639.89	26.14	4.75	22	0	0	1
	1	1	0	0	84556.50	106639.89	26.12	41.33	42	37	0	1
	1	1	1	0	84556.50	106639.89	26.12	49.49	42	37	0	1
	0	0	0	0	84556.11	84577.30	**0.03**	3600.04	3377	0	0	206435
	1	1	1	0	84570.15	84570.15	**0.00**	2230.19	1269	59	0	184110
	1	1	1	5	84570.15	84570.15	**0.00**	1848.25	1167	59	0	112653
S2_CN_30	0	0	0	0	97229.57	674855.17	594.08	31.91	263	0	0	1
	1	0	0	0	97243.65	173372.89	78.29	15.26	109	0	0	1
	1	1	0	0	97309.32	173372.89	78.17	242.47	241	162	0	1
	1	1	1	0	97315.83	173372.89	78.15	251.86	239	163	36	1
	0	0	0	0	97479.66	97594.71	**0.12**	3600.06	15945	0	0	55939
	1	1	1	0	97536.45	97543.71	**0.01**	1593.71	1571	376	1,002	7954
	1	1	1	5	97536.45	97543.71	**0.01**	819.23	334	376	802	5389
S2_CN_35	0	0	0	0	110287.01	727020.13	559.21	53.61	320	0	0	1
	1	0	0	0	111678.97	727020.13	550.99	42.44	166	0	0	1
	1	1	0	0	111760.67	727020.13	550.52	338.62	168	121	0	1
	1	1	1	0	111764.90	727020.13	550.49	356.12	172	122	20	1
	0	0	0	0	111014.22	727020.13	554.89	3600.11	14709	0	0	119420
	1	1	1	0	111828.10	111843.87	**0.01**	3600.08	1948	295	20	84142
	1	1	1	5	111843.87	111843.87	**0.00**	2698.62	977	501	30	52812
S2_CN_40	0	0	0	0	115940.89	784050.34	576.25	72.37	282	0	0	1
	1	0	0	0	116075.57	784050.34	575.47	58.51	75	0	0	1
	1	1	0	0	116164.96	784050.34	574.95	911.02	306	176	0	1
	1	1	1	0	116171.37	784050.34	574.91	955.69	306	178	99	1
	0	0	0	0	115954.86	116588.82	**0.55**	3600.39	3380	0	0	5226
	1	1	1	0	116189.45	116588.82	**0.34**	3606.32	606	467	99	3542
	1	1	1	5	116262.53	116375.49	**0.10**	3607.45	615	501	126	3327
S2_CN_45	0	0	0	0	121938.78	868823.53	612.51	63.40	199	0	0	1
	1	0	0	0	122092.03	265006.83	117.05	90.26	162	0	0	1
	1	1	0	0	122239.19	265006.83	116.79	1500.77	446	166	0	1
	1	1	1	0	122239.19	265006.83	116.79	1511.42	449	166	46	1
	0	0	0	0	122448.66	868823.53	609.54	3601.01	3150	0	0	3382
	1	1	1	0	122676.92	265006.83	116.02	3600.13	643	501	76	277
	1	1	1	5	122676.92	136204.83	11.03	3609.53	639	501	76	271
S2_CN_50	0	0	0	0	138946.68	880965.14	534.03	218.33	426	0	0	1
	1	0	0	0	144164.37	880965.14	511.08	165.78	180	0	0	1
	1	1	0	0	144179.63	880965.14	511.02	1601.30	249	202	0	1
	1	1	1	0	144179.63	880965.14	511.02	1821.67	249	215	52	1
	0	0	0	0	139094.27	880965.14	533.36	3601.14	3955	0	0	896
	1	1	1	0	144405.68	880965.14	510.06	3600.05	738	230	52	50
	1	1	1	5	144408.57	145201.07	**0.55**	3605.50	805	227	52	45

Table A.2. Results on circle network topology with Set S2.

Semicircle Networks - Set 1												
Id	VUB & CC	AGSI	RCUTS	HEUR	LB	UB	GAP	Time	#GSI	#AGSI	#RCUTS	# Nodes
S1_SCN_20	0	0	0	0	1949.06	8566.12	339.50	3.13	75	0	0	1
	1	0	0	0	1957.90	1957.90	**0.00**	6.36	34	0	0	1
	1	1	0	0	1957.90	1957.90	**0.00**	4.36	32	2	0	1
	1	1	1	0	1957.90	1957.90	**0.00**	5.70	32	2	0	1
	0	0	0	0	1957.90	1957.90	**0.00**	8.70	77	0	0	11
	1	1	1	0	1957.90	1957.90	**0.00**	5.61	32	2	0	1
	1	1	1	5	1957.90	1957.90	**0.00**	5.66	32	2	0	1
S1_SCN_25	0	0	0	0	2622.24	10802.93	311.97	4.26	85	0	0	1
	1	0	0	0	2625.39	10802.93	311.48	4.36	48	0	0	1
	1	1	0	0	2630.78	10802.93	310.64	10.25	56	11	0	1
	1	1	1	0	2630.78	10802.93	310.64	10.85	56	11	0	1
	0	0	0	0	2638.73	2638.73	**0.00**	70.36	163	0	0	3934
	1	1	1	0	2638.73	2638.73	**0.00**	48.89	73	55	87	87
	1	1	1	5	2638.73	2638.73	**0.00**	13.61	63	10	0	5
S1_SCN_30	0	0	0	0	2900.45	12476.70	330.16	5.38	94	0	0	1
	1	0	0	0	2903.99	12476.70	329.64	6.18	59	0	0	1
	1	1	0	0	2908.24	2908.24	**0.00**	40.08	104	116	0	1
	1	1	1	0	2908.24	2908.24	**0.00**	41.36	104	116	0	1
	0	0	0	0	2908.24	2908.24	**0.00**	10.02	128	0	0	105
	1	1	1	0	2908.24	2908.24	**0.00**	41.55	104	116	0	1
	1	1	1	5	2908.24	2908.24	**0.00**	42.80	104	116	0	1
S1_SCN_35	0	0	0	0	3234.99	14227.69	339.81	8.12	85	0	0	1
	1	0	0	0	3235.32	14227.69	339.76	7.09	43	0	0	1
	1	1	0	0	3239.40	14227.69	339.21	114.00	95	77	0	1
	1	1	1	0	3239.40	14227.69	339.21	115.63	96	79	22	1
	0	0	0	0	3241.78	3241.78	**0.00**	1048.09	607	0	0	30900
	1	1	1	0	3241.78	3241.78	**0.00**	256.55	121	304	22	29
	1	1	1	5	3241.78	3241.778	**0.00**	148.38	99	257	22	10
S1_SCN_40	0	0	0	0	3250.18	16091.63	395.10	26.18	94	0	0	1
	1	0	0	0	3251.92	16091.63	394.83	19.88	54	0	0	1
	1	1	0	0	3259.96	16091.63	393.61	405.80	106	100	0	1
	1	1	1	0	3260.87	16091.63	393.48	426.14	106	100	204	1
	0	0	0	0	3260.57	3267.63	**0.22**	3600.84	746	0	0	6260
	1	1	1	0	3264.83	3264.83	**0.00**	3176.71	280	261	204	2278
	1	1	1	5	3264.83	3264.83	**0.00**	2923.48	271	261	204	1912
S1_SCN_45	0	0	0	0	3817.24	16754.53	338.92	32.04	109	0	0	1
	1	0	0	0	3820.07	16754.53	338.59	28.67	74	0	0	1
	1	1	0	0	3821.01	16754.53	338.48	242.58	100	58	0	1
	1	1	1	0	3821.50	3841.20	**0.52**	258.56	118	76	209	1
	0	0	0	0	3823.35	4478.83	17.14	3600.15	1499	0	0	2866
	1	1	1	0	3829.98	3829.98	**0.00**	1226.34	147	153	209	9
	1	1	1	5	3829.98	3833.74	**0.10**	1225.82	147	153	209	9
S1_SCN_50	0	0	0	0	4177.38	18724.60	348.24	49.92	145	0	0	1
	1	0	0	0	4177.42	18724.60	348.23	42.37	72	0	0	1
	1	1	0	0	4180.66	18724.60	347.89	734.22	127	134	0	1
	1	1	1	0	4180.66	18724.60	347.89	832.60	127	134	0	1
	0	0	0	0	4181.87	4182.90	**0.02**	3601.86	377	0	0	3550
	1	1	1	0	4182.90	4182.90	**0.00**	1816.52	184	106	0	878
	1	1	1	5	4182.90	4182.90	**0.00**	1535.59	145	71	0	724

Table A.3. Results on semi-circle network topology with Set S1.

Semicircle Networks - Set 2

Id	VUB & CC	AGSI	RCUTS	HEUR	LB	UB	GAP	Time	#GSI	#AGSI	#RCUTS	# Nodes
S2_SCN_20	0	0	0	0	62809.66	119110.01	89.64	6.12	144	0	0	1
	1	0	0	0	62818.51	433872.31	590.68	3.59	29	0	0	1
	1	1	0	0	62832.49	62834.61	**0.00**	8.78	38	14	0	1
	1	1	1	0	62832.49	62834.61	**0.00**	8.61	38	14	0	1
	0	0	0	0	62834.61	62834.61	**0.00**	83.31	358	0	0	8346
	1	1	1	0	62834.61	62834.61	**0.00**	8.18	38	14	0	1
	1	1	1	5	62834.61	62834.61	**0.00**	8.11	43	18	0	1
S2_SCN_25	0	0	0	0	79814.96	524725.39	557.43	8.90	179	0	0	1
	1	0	0	0	79836.00	524725.39	557.25	7.84	28	0	0	1
	1	1	0	0	79851.62	524725.39	557.13	111.33	95	88	0	1
	1	1	1	0	79851.62	524725.39	557.13	116.04	95	88	0	1
	0	0	0	0	79858.39	79870.74	**0.02**	3600.12	3604	0	0	188550
	1	1	1	0	79858.39	79858.39	**0.00**	239.95	515	247	0	2559
	1	1	1	5	79858.39	79858.39	**0.00**	115.20	78	184	0	105
S2_SCN_30	0	0	0	0	89303.88	602765.80	574.96	24.99	243	0	0	1
	1	0	0	0	89321.46	206822.21	131.55	19.19	46	0	0	1
	1	1	0	0	89418.86	206822.21	131.30	209.44	116	155	0	1
	1	1	1	0	89418.86	206822.21	131.30	212.73	116	155	0	1
	0	0	0	0	89349.66	110885.76	24.10	3600.08	15430	0	0	68922
	1	1	1	0	89421.66	90979.64	1.74	3600.06	7000	317	0	47819
	1	1	1	5	89423.75	90979.64	1.74	3601.23	6937	286	0	46927
S2_SCN_35	0	0	0	0	100356.34	727706.23	625.12	35.50	305	0	0	1
	1	0	0	0	101547.45	727706.23	616.62	10.21	125	0	0	1
	1	1	0	0	101620.68	727706.23	616.10	436.41	184	162	0	1
	1	1	1	0	101627.38	727706.23	616.05	495.00	186	165	100	1
	0	0	0	0	101159.12	123196.02	21.78	3600.03	16165	0	0	24692
	1	1	1	0	101657.70	101661.88	**0.00**	3600.04	2575	501	100	33286
	1	1	1	5	101661.88	101661.88	**0.00**	3163.02	2553	502	100	31488
S2_SCN_40	0	0	0	0	101750.04	830403.28	716.12	96.23	282	0	0	1
	1	0	0	0	102466.66	830403.28	710.41	76.89	164	0	0	1
	1	1	0	0	102530.81	830403.28	709.91	1170.06	213	266	0	1
	1	1	1	0	102530.81	830403.28	709.91	1211.49	213	266	0	1
	0	0	0	0	102295.80	102741.44	**0.44**	3600.39	2487	0	0	5656
	1	1	1	0	102531.80	102544.09	**0.01**	3603.26	1209	400	0	3784
	1	1	1	5	102544.09	102544.09	**0.00**	3515.35	562	472	0	63
S2_SCN_45	0	0	0	0	109137.50	816635.38	648.26	42.71	136	0	0	1
	1	0	0	0	109159.30	816635.38	648.11	23.64	46	0	0	1
	1	1	0	0	109228.01	816635.38	647.64	916.82	145	185	0	1
	1	1	1	0	109228.01	816635.38	647.64	981.95	145	185	0	1
	0	0	0	0	109142.28	816635.38	648.23	3600.41	3011	0	0	3605
	1	1	1	0	109228.01	816635.38	647.64	3600.02	569	500	0	2913
	1	1	1	5	102531.80	102544.09	**0.01**	3600.82	585	457	0	3296
S2_SCN_50	0	0	0	0	117977.13	854405.67	624.21	83.94	248	0	0	1
	1	0	0	0	128674.50	854405.67	564.01	76.23	118	0	0	1
	1	1	0	0	128745.93	854405.67	563.64	662.18	203	107	0	1
	1	1	1	0	128745.93	854405.67	563.64	669.45	203	107	0	1
	0	0	0	0	118349.19	854405.67	621.94	3600.30	3566	0	0	2298
	1	1	1	0	128747.65	854405.67	563.63	3680.40	718	500	30	942
	1	1	1	5	128747.16	148830.64	15.60	3600.16	439	420	26	711

Table A.4. Results on semi-circle network topology with Set S2.

Grid Networks - Set 1

Id	VUB & CC	AGSI	RCUTS	HEUR	LB	UB	GAP	Time	#GSI	#AGSI	#RCUTS	# Nodes
S1_GN_20	0	0	0	0	2351.58	2351.58	**0.00**	1.71	60	0	0	1
	1	0	0	0	2351.58	2351.58	**0.00**	1.08	15	0	0	1
	1	1	0	0	2351.58	2351.58	**0.00**	1.21	15	1	0	1
	1	1	1	0	2351.58	2351.58	**0.00**	2.44	15	1	0	1
	0	0	0	0	2351.58	2351.58	**0.00**	3.98	60	0	0	1
	1	1	1	0	2351.58	2351.58	**0.00**	2.67	15	1	0	1
	1	1	1	5	2351.58	2351.58	**0.00**	3.11	15	1	0	1
S1_GN_25	0	0	0	0	2851.74	2852.13	**0.01**	3.10	69	0	0	1
	1	0	0	0	2852.13	2852.13	**0.00**	1.84	26	0	0	1
	1	1	0	0	2852.13	2852.13	**0.00**	2.19	26	1	0	1
	1	1	1	0	2852.13	2852.13	**0.00**	2.76	26	1	0	1
	0	0	0	0	2852.13	2852.13	**0.00**	4.55	69	0	0	1
	1	1	1	0	2852.13	2852.13	**0.00**	2.18	26	1	0	1
	1	1	1	5	2852.13	2852.13	**0.00**	1.26	23	0	0	1
S1_GN_30	0	0	0	0	2917.83	2917.83	**0.00**	4.57	76	0	0	1
	1	0	0	0	2917.83	2917.83	**0.00**	2.32	21	0	0	1
	1	1	0	0	2917.83	2917.83	**0.00**	2.39	21	1	0	1
	1	1	1	0	2917.83	2917.83	**0.00**	2.80	21	1	0	1
	0	0	0	0	2917.83	2917.83	**0.00**	4.08	76	0	0	1
	1	1	1	0	2917.83	2917.83	**0.00**	2.81	21	1	0	1
	1	1	1	5	2917.83	2917.83	**0.00**	2.88	21	1	0	1
S1_GN_35	0	0	0	0	3405.59	3405.59	**0.00**	5.80	90	0	0	1
	1	0	0	0	3405.59	3405.59	**0.00**	4.70	36	0	0	1
	1	1	0	0	3405.59	3405.59	**0.00**	6.00	36	2	0	1
	1	1	1	0	3405.59	3405.59	**0.00**	6.76	36	2	0	1
	0	0	0	0	3405.59	3405.59	**0.00**	6.63	90	0	0	1
	1	1	1	0	3405.59	3405.59	**0.00**	8.02	36	2	0	1
	1	1	1	5	3405.59	3405.593	**0.00**	3.44	36	1	0	1
S1_GN_40	0	0	0	0	3752.73	15248.96	306.34	29.53	99	0	0	1
	1	0	0	0	3753.87	15248.96	306.22	21.58	45	0	0	1
	1	1	0	0	3755.09	3755.09	**0.00**	27.01	46	4	0	1
	1	1	1	0	3755.09	3755.09	**0.00**	37.07	46	4	0	1
	0	0	0	0	3755.09	3755.09	**0.00**	3109.48	676	0	0	2467
	1	1	1	0	3755.09	3755.09	**0.00**	54.72	46	4	0	1
	1	1	1	5	3755.09	3755.09	**0.00**	56.17	46	4	0	1
S1_GN_45	0	0	0	0	4072.48	16848.93	313.73	39.20	114	0	0	1
	1	0	0	0	4076.52	16848.93	313.32	22.65	52	0	0	1
	1	1	0	0	4077.15	16848.93	313.25	175.08	62	61	0	1
	1	1	1	0	4077.15	16848.93	313.25	292.41	62	61	0	1
	0	0	0	0	4076.99	4077.97	**0.02**	3600.28	203	0	0	3890
	1	1	1	0	4077.97	4077.97	**0.00**	538.98	65	65	0	7
	1	1	1	5	4077.97	4077.97	**0.00**	214.52	55	9	0	1
S1_GN_50	0	0	0	0	4575.53	17384.04	279.93	43.90	115	0	0	1
	1	0	0	0	4578.89	17384.04	279.66	23.41	49	0	0	1
	1	1	0	0	4579.54	4580.24	**0.02**	212.40	57	23	0	1
	1	1	1	0	4579.54	4580.24	**0.02**	266.67	57	23	0	1
	0	0	0	0	4578.23	4580.24	**0.04**	3601.53	830	0	0	2587
	1	1	1	0	4580.24	4580.24	**0.00**	367.44	57	23	0	3
	1	1	1	5	4580.24	4580.24	**0.00**	367.16	57	23	0	3

Table A.5. Results on grid network topology with Set S1.

Grid Networks - Set 2

Id	VUB & CC	AGSI	RCUTS	HEUR	LB	UB	GAP	Time	#GSI	#AGSI	#RCUTS	# Nodes
S2_GN_20	0	0	0	0	73369.17	110581.55	50.72	9.60	224	0	0	1
	1	0	0	0	73720.46	137254.14	86.18	2.62	22	0	0	1
	1	1	0	0	73724.42	137254.14	86.17	3.99	28	5	0	1
	1	1	1	0	73724.42	137254.14	86.17	4.96	28	5	0	1
	0	0	0	0	73650.71	73734.69	**0.11**	3600.01	2090	0	0	315152
	1	1	1	0	73734.69	73734.69	**0.00**	18.16	55	49	0	268
	1	1	1	5	73734.69	73734.69	**0.00**	20.84	12	49	0	268
S2_GN_25	0	0	0	0	84749.92	167563.18	97.71	10.14	210	0	0	1
	1	0	0	0	84762.37	555516.08	555.38	6.41	36	0	0	1
	1	1	0	0	84776.74	555516.08	555.27	28.77	57	39	0	1
	1	1	1	0	84776.74	555516.08	555.27	30.70	57	39	0	1
	0	0	0	0	84764.97	85001.69	**0.28**	3600.37	9375	0	0	123454
	1	1	1	0	84785.43	84785.43	**0.00**	63.45	73	105	0	118
	1	1	1	5	84785.43	84785.43	**0.00**	63.39	73	105	0	118
S2_GN_30	0	0	0	0	91408.95	642128.83	602.48	21.84	248	0	0	1
	1	0	0	0	91416.77	154285.29	68.77	18.82	46	0	0	1
	1	1	0	0	91437.91	154285.29	68.73	179.12	106	119	0	1
	1	1	1	0	91437.91	154285.29	68.73	181.53	106	119	0	1
	0	0	0	0	91425.75	91457.30	**0.03**	3601.06	8377	0	0	124448
	1	1	1	0	91448.29	91448.29	**0.00**	349.46	177	401	0	590
	1	1	1	5	91448.29	91448.29	**0.00**	349.42	177	401	0	590
S2_GN_35	0	0	0	0	108901.14	617690.66	467.20	45.19	445	0	0	1
	1	0	0	0	112542.02	160846.30	42.92	22.48	106	0	0	1
	1	1	0	0	112630.86	160846.30	42.81	244.67	293	80	0	1
	1	1	1	0	112630.86	160846.30	42.81	248.74	293	80	0	1
	0	0	0	0	109683.38	157241.81	43.36	3607.75	16265	0	0	20012
	1	1	1	0	112789.67	139051.47	23.28	3600.09	5138	506	502	19116
	1	1	1	5	112788.97	117973.8	4.60	3600.04	3883	500	502	7620
S2_GN_40	0	0	0	0	112379.75	762632.39	578.62	70.38	263	0	0	1
	1	0	0	0	112387.48	216865.91	92.96	50.57	69	0	0	1
	1	1	0	0	112436.67	216865.91	92.88	607.78	165	165	0	1
	1	1	1	0	112436.67	216865.91	92.88	758.54	165	165	0	1
	0	0	0	0	112398.15	762632.39	578.51	3600.17	2437	0	0	5187
	1	1	1	0	112438.56	216865.91	92.88	3600.67	814	498	0	1946
	1	1	1	5	112438.56	135586.25	20.59	3600.79	787	473	0	2306
S2_GN_45	0	0	0	0	122701.48	772707.15	529.75	118.71	292	0	0	1
	1	0	0	0	122832.30	228858.69	86.32	51.04	108	0	0	1
	1	1	0	0	123264.46	228858.69	85.66	901.94	383	122	0	1
	1	1	1	0	123283.76	228858.69	85.64	1011.99	384	126	284	1
	0	0	0	0	122826.11	772707.15	529.11	3600.30	2439	0	0	3992
	1	1	1	0	123354.85	228858.69	85.53	3600.01	1149	200	323	1523
	1	1	1	5	123386.83	143084.03	15.96	3600.66	1021	179	284	1827
S2_GN_50	0	0	0	0	135075.72	720219.06	433.20	162.92	381	0	0	1
	1	0	0	0	139878.91	720219.06	414.89	31.92	90	0	0	1
	1	1	0	0	139984.21	720219.06	414.50	1225.74	306	169	0	1
	1	1	1	0	139999.81	720219.06	414.44	1314.75	308	183	187	1
	0	0	0	0	135132.73	720219.06	432.97	3600.06	2966	0	0	1790
	1	1	1	0	140810.22	267396.36	89.90	3600.43	1029	498	451	883
	1	1	1	5	140983.65	158934.41	12.73	3600.43	906	578	402	862

Table A.6. Results on grid network topology with Set S2.