RAIRO-Oper. Res. 53 (2019) 807–827
https://doi.org/10.1051/ro/2017065

RAIRO Operations Research
www.rairo-ro.org

ALLOCATING NODES TO HUBS FOR MINIMIZING THE HUBS PROCESSING RESOURCES: A CASE STUDY

Ali Balma[1,2,*] and Mehdi Mrad[2,3]

Abstract. This paper addresses the problem of allocating the terminal nodes to the hub nodes in a telecommunication network. Since the flow processing induces some undesirable delay, the objective is to minimize the total flow processed by the hubs. This study focuses on a real life network of the tunisian operator *Tunisie Telecom* whose operations managers are concerned by the quality of service. We provide three compact formulations that give optimal solutions for networks of large size. In particular, the last two are obtained by applying the Reformulation-Linearization Technique to a nonlinear formulation of the problem. The latter formulation derived within this approach is the most computationally effective, as pointed out by the computational experiments conducted on the real life network of *Tunisie Telecom* with 110 nodes and 5 hubs. Finally, we discuss and compare between the single allocation and double allocation configurations.

Mathematics Subject Classification. 90B18, 90B20, 90B80, 90B90, 90C11, 90C30.

Received August 28, 2016. Accepted September 7, 2017.

1. Introduction

Telecommunications networks are generally deployed in two layers [22]

- a backbone layer made up of high capacity nodes and edges. These nodes, the so-called hubs, are strongly interconnected and are in charge of collecting and processing flows emanating from and arriving to the terminal nodes.
- an access layer composed of the terminal nodes from which the flows originate and at which they end. A demand flow exists between each pair of terminal nodes. These nodes are not connected directly to each other but their flow must pass through the hubs of the backbone layer. Each terminal node must be connected to a single hub by a link that carries all flows related to that node, the so-called single allocation mode.

Generally, the backbone network has its own transport technology adapted to the high bandwidth and security requirements. However, the transport technology of the access network is specific to, relatively, less bandwidth.

Keywords. Hub allocation, Non-linear programming, Combinatorial optimization, Graphs and Networks, Reformulation-Linearization Technique, Telecommunications.

[1] Université de Tunis, Ecole Nationale Supérieure d'Ingénieurs de Tunis, 5 av. Taha Hussein, 1008 Tunis, Tunisia.

[2] Université de Tunis, BADEM, Tunis Business School, PO.Box 65, Bir El Kassaa, 2059, Tunisia.

[3] Department of Industrial Engineering, King Saud University, College of Engineering P.O. Box 800, Riyadh 11421, Saudi Arabia.

*Corresponding author: ali.balma@gmail.com

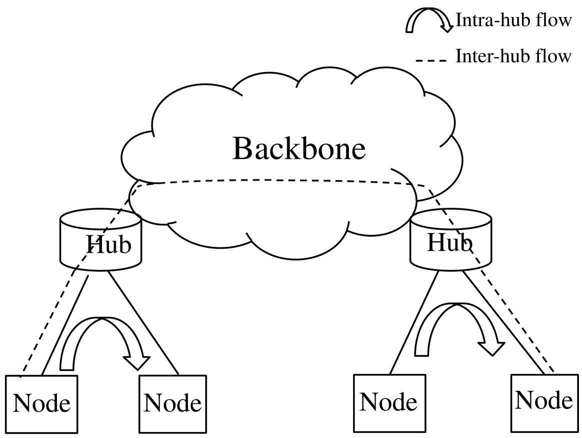

Figure 1. Intra-hub and inter-hub flows.

Therefore, the hubs are in charge of adapting the signals originating from the terminal nodes to the transport technology of the backbone. They process the flow when it must be transported through the backbone network. On another hand, the flow between two nodes that are connected to the same hub is transferred from the source to the destination transparently without being processed. In this last case, the processing resources of the hub are saved. The flow that takes place behind the same hub is called intra-hub traffic, while the flow transported through the backbone network is called inter-hub traffic (see Fig. 1).

One major concern of telecommunication network managers is the delay induced by the processing of flow in the hubs. Indeed, the inter-hub traffic must be processed by the hubs, inducing thereby a certain latency. In order to reduce this latency, the terminal nodes must be connected to the hubs in such a way that the use of the processing resources of the hubs is minimized; that is, minimizing the inter-hub traffic.

The problem studied in this paper originates from a real world operator *Tunisie Telecom*, which is the historical and major telecommunication operator of Tunisia. It is a state owned company which capital is detained, jointly by the tunisian state and by TECOM-DIG private consortium, respectively with 65% and 35% of capital share. It employs 8000 people and realizes approximately 1 billion US $ of annual turnover. It manages a multiservice telecommunication network consisting of fixed and mobile telephony, data communications and Internet. Although it offers fixed and mobile telephony, it remains the major market player in the fixed telephony service with almost 90% of market share.

The fixed and mobile telephony network complies with the nowadays layered structure. Its network is constituted by an IP backbone, based on IP transportation technology, and many terminal nodes that exchange voice calls. In order to be transported to their destinations through the backbone network, there are some special equipment located in the backbone border that are in charge of converting the voice calls into IP packets. These equipments, the so-called Mediagateways, play the role of hubs. Each hub possesses voice processing boards that achieve the IP transformation. This IP conversion is performed only when the calls take place between two nodes connected to two different hubs (inter-hub traffic), otherwise, there is no need to convert voice into IP packets (intra-hub traffic).

The exchange equipment may seem obsolete as most nowadays operators are migrating toward the all IP networks. In all IP networks, any communication takes place over IP all the way between the end users' terminals and no intermediate conversion equipment is needed in the operator's network. Nevertheless, the migration process cannot be achieved overnight and the exchanges must interwork with the newly installed all IP devices for yet some years. It is one of the roles of the Mediagateways to assure the adaptation.

The operator managers aim to establish a connection plan between nodes and hubs that minimizes the use of the IP conversion resources. Thus reducing delays due to the IP transformation. With such a manner of connecting nodes to hubs, the total traffic injected into the backbone network is minimized. In fact, the backbone is a large platform of many types of traffic and the peripheral networks must be deployed in a way that minimizes the inter-hub flow, thus preventing a possible congestion in the backbone.

In the present paper, we study the problem of minimizing the total flow processed by the resources installed on the hubs. In addition, rather than assuming that any node is a possible candidate for being a hub, we consider that the hub nodes are known (the so-called Point-Of-Presence or POP) and the terminal nodes must be connected to the hubs while minimizing the total flow processed by the hubs. Also, any given node is located as close as possible to the customers and the hubs are situated in the operators premises. The node and the hub are related by an access network. Any given node can be connected to a subset of hubs that are determined through the topology of the access network.

Any node is linked to at least two hubs, among the possible ones, through two links, for security reasons. One link carries the traffic in the normal state, is active and the other is in a standby mode. When a failure occurs on the active link or on the current hub, then the node switches over the traffic to the standby link that becomes active, in order to avoid service disruption.

The connection plan is not static and must be adpated to the changing traffic pattern. In fact, the peak of traffic changes over time and this connection plan must be updated regularly in order to meet the customers' needs, particularly at the busy hour for the real time voice over IP communications. According to the traffic pattern of *Tunisie Telecom*, the periodicity of the plan update is less than three months, which makes the problem situated at the operational level.

Moreover, the operator managers monitor some performance indicators about calls' rejection due to resources unaivailability. Once such a problem occurs, the connection plan is modified in order to relieve the overloaded resources. This imposes having a rapid running optimizing tool.

We call the described problem as Allocating Nodes to Hubs Problem and is denoted, hereinafter, by ANHP. It consists of determining the optimal allocation of nodes to hubs while minimizing the total traffic processed by the hubs. The aim of the present work is to provide the network managers of *Tunisie Telecom* with an optimization fast running tool giving rapidly the optimal solution.

Although our approach is directed at solving a particular problem, the methodology can be applied to solve any other hub allocation problem, for instance, connecting the customers access equipments to the WDM network [17]. Recent applications deal with data centers design and virtual machines placement in the Cloud Computing network [19].

The present paper is organized as follows. In Section 2, we give a literature survey about hub location and allocation problem. In Section 3, we describe formally the problem and present three mathematical formulations. The first formulation is elaborated straightforwardly whereas the last two formulations are devised from the so-called Reformulation-Linearization Technique (RLT) [23, 24]. In Section 4, we compare theoretically the given formulations in terms of their sizes and the quality of their linear relaxations. In addition, we investigate and compare the double allocation mode to the single allocation mode in Section 5. In Section 6, we provide computational results for the devised formulations and compare their computational performances on both real life network and randomly generated data.

2. Related works

The present problem is a variant of the hub location problem. A detailed review of its different variants is given in [3, 7, 9]. Although it was first stated in the early nineties, it regained attention in the recent years due to its many applications in real life problems such as air transportation, telecommunication or logistics.

In [10–13, 26], the authors consider the capacitated version of the problem where the hubs have limited capacity of processing flows. They take into account the distances between the nodes in the objective function. The authors of [20] solve the combined version of the hub location and the network design problems with a Benders decomposition approach. A Benders approach is also devised for the uncapacitated hub location problem in [5]. A Lagrangian relaxation approach is applied to the uncapacitated problem in [21].

The authors of [22] address the same version of the problem of allocating the terminal nodes to the hubs while minimizing the total costs composed of the costs of the capacity on the hubs and the cost of connecting the nodes to the hubs. However, inter-hub traffic costs have not been taken into account.

In [1, 11], the authors tackle the problem of designing the capacities to install on the hubs, while considering the costs on the inter-hub traffic. In [11], the authors strengthen the formulation of [13], whereas the authors of [1] use Dantzig-Wolfe decompositon technique.

The authors of [27] consider the problem of hub location with multimodal transportation offered by the hubs. This variant is first introduced by [2]. The problem is solved with a heuristic that combines linear relaxation, Lagrangian relaxation and Branch-and-Bound. In [8], the author considers the problem of the single allocation hub location problem and solves it with a heuristic combining simulated annealing and Tabu lists.

The p-hub single allocation problem has been studied in [6, 14, 18]. This problem addresses the location of p hubs and allocates terminals to them in order to minimize the maximum travel time between any pair of terminals.

This paper is an extension of [4]. Therein, a compact formulation was given for the problem that turned out to be impractical for large networks. Indeed, the problem cannot be solved exactly in a reasonable time for networks with more than 70 nodes.

In the present paper, we do not adopt sophisticated decomposition techniques that are time investing or heuristics. Rather, we give compact formulations for ANHP that outperform the one given in [4] and provide optimal solutions in reasonable running times for network size beyond 100 nodes. The operator managers expressed the need for a rapid and efficient tool that allows them to determine and update the allocation of the nodes to the hubs through optimal basis.

The problem ANHP can be formally described as follows: given a set N of nodes and a set H of hubs that are fully connected through the backbone network (a complete graph). Each pair $(i, j) \in N^2$ of nodes interchanges nonzero flow denoted by d_{ij} . The nodes of N cannot be connected directly to each other but through the hub nodes. A hub $h \in H$ collects the flows from the nodes and reroutes them to their destination either directly or through another hub. For this latter case, the flows are processed and adapted to the transport technology of the backbone network, then sent to the destination nodes *via* another hub. However for the first case where the nodes are connected to the same hub, no conversion is needed and the flow is directly and transparently sent to the destination nodes.

The processing resources used for the signal conversion in the hubs may induce some delays that are undesirable for real-time applications. Consequently, the problem the network managers face is to assign the nodes to the hubs such that the use of the processing and conversion resources located in the hubs is minimized, taking advantage of the fact that these resources are not used for the intra-hub traffic.

In addition, a given node cannot be connected to any hub but to a subset of the hubs, because of access network topology considerations. We denote by $H_i \subset H$ a subset of hubs to which a given node $i \in N$ can be connected ($|H_i| \geqslant 2,\ \forall i \in N$ as described in the previous section for security reasons). Conversely, we denote by $S_h \subset N$ as the subset of the nodes that can be allocated to the hub $h \in H$.

3. Problem formulations

3.1. Partition based formulation

We give a formulation based on partitioning the nodes depending on whether they are connected to the same hub or not. For that purpose, we define the following decision variables:

- p_{ij} takes the value 1 when the nodes i and j are connected to two different hubs, and takes 0 otherwise.
- x_i^h takes 1 when the node i is connected to the hub h, and takes 0 otherwise.

The formulation with these decision variables, that we denote by (PF), is as follows

$$\min z_2 = \sum_{i\in N}\sum_{j\in N: i<j} p_{ij}(d_{ij} + d_{ji}) \tag{3.1}$$

subject to

$$\sum_{h\in H_i} x_i^h = 1 \quad \forall i \in N. \tag{3.2}$$

$$p_{ij} \geqslant x_i^h - x_j^h, \forall h \in H, i \in S_h, j \in S_h : i < j. \tag{3.3}$$

$$p_{ij} \geqslant x_j^h - x_i^h, \forall h \in H, i \in S_h, j \in S_h : i < j. \tag{3.4}$$

$$p_{ij} \geqslant x_i^h, \forall h \in H, i \in S_h, j \notin S_h. \tag{3.5}$$

$$x_i^h \in \{0,1\} \quad \forall h \in H, i \in S_h. \tag{3.6}$$

$$p_{ij} \in \{0,1\} \quad \forall i \in N, j \in N. \tag{3.7}$$

The objective function (3.1) aims to minimize the inter-hub flow. The constraints (3.2) stipulate that any given node i can be connected to only one hub, according to the single allocation mode. The constraints (3.3) and (3.4) state that p_{ij} takes the value 1 when both nodes i and j are assigned to two different hubs. The constraints (3.5) force p_{ij} to take the value of x_i^h when the node i can be connected to h ($i \in S_h$) but not the node j ($j \notin S_h$).

The experiments showed that it is computationally advantageous to add the constraints (3.8) as they strengthen the formulation (PF).

$$p_{ij} \geqslant x_j^h, \forall h \in H, j \in S_h, i \notin S_h. \tag{3.8}$$

In fact, for instance, assume that node $i = 3$ can be connected to hub 1 and other hubs except hub 2. Similarly, node $j = 5$ can be allocated to hub 2 and other hubs except hub 1. Hence, from (3.5), we have $p_{35} \geqslant x_3^1$. In addition $p_{35} \geqslant x_5^2$, for the same node pair $i = 3, j = 5$, according to (3.8). This inequality makes the formulation (PF) stronger.

It is noteworthy that the variables p_{ij} can be relaxed to be continuous while x_i^h are kept binary. In fact, from the constraints (3.3), (3.4) and (3.5), we deduce that the values of p_{ij} are either 0 or 1, since $x_i^h \in \{0,1\}$, $\forall i \in N$. It is easy to notice that the smallest values of p_{ij} that lead to the minimal value of the objective function, are either 0 or 1, when the variables x_i^h are constrained to be binary.

Although giving the optimal solution for more than 70 nodes in a reasonable time, the problem remains challenging for more than 110 nodes, as shown in the computational experiments' section. Hence, we give an alternative and disaggregated formulation that gives tighter bounds and speeds up the exact resolution on a commercial solver. It is obtained by applying the so-called Reformulation-Linearization Technique (RLT).

3.2. RLT based formulation

The RLT [23, 24] is an approach for recasting difficult optimization problems and devising new tight linear representation of the original problem. It departs from either a linear or a nonlinear formulation of a problem and leads to a new tight polyhedral relaxation of the problem. It consists essentially of two phases: the Reformulation phase creates additional redundant nonlinear constraints by multiplying the original constraints by the variables x and their complements $(1-x)$. Then, the Linearization phase consists of variable substitution of the quadratic terms, obtained through the Reformulation step, by new continuous variables. Such a technique was successfully applied to many various difficult problems, such as the traveling salesman problem [25], the prize collecting Steiner tree problem [16], the shortest path problem with negative cycles, the minimum-weight connected subgraph problem [15] and the design of local access transport area network [22].

First, we formulate the problem into a nonlinear program and then we apply the two phases described above.

The total flow processed by a given hub h and issued from a node i is nonzero when the nodes i and j are not connected simultaneously to the same hub h; that is equal to $\sum_{j\in N} x_i^h(1-x_j^h)d_{ij}, \forall h \in H, \forall i \in N$. Then the nonlinear formulation for the problem, denoted by (NLF), is the following:

$$\min z_3 = \sum_{h\in H}\sum_{i\in N}\sum_{j\in N} x_i^h(1-x_j^h)d_{ij} \tag{3.9}$$

subject to

$$\sum_{h\in H_i} x_i^h = 1, \quad \forall i \in N. \tag{3.2}$$

$$x_i^h \in \{0,1\},\ \forall h \in H, i \in S_h. \tag{3.6}$$

The nonlinear formulation (NLF) is simple and has only one set of constraints (3.2) stipulating that any node i must be connected to a single hub h. The objective function (3.9) computes the total flow emanating from the node i and processed by the hub h.

With reference to (3.9) and since $x_i^h = 0\ \forall i \notin S_h$, then the total inter-hub flow is the following

$$z_3 = \sum_{h\in H}\sum_{i\in S_h} x_i^h \left(\sum_{j\in N}(1-x_j^h)d_{ij}\right).$$

That is

$$z_3 = \sum_{h\in H}\sum_{i\in S_h} x_i^h \left(\sum_{j\in N} d_{ij} - \sum_{j\in N} x_j^h d_{ij}\right).$$

Likewise $x_j^h = 0\ \forall j \notin S_h$ which leads to

$$z_3 = \sum_{h\in H}\sum_{i\in S_h} x_i^h \left(\sum_{j\in N} d_{ij} - \sum_{j\in S_h} x_j^h d_{ij}\right).$$

After rearranging, we get

$$z_3 = \sum_{h\in H}\sum_{i\in S_h} x_i^h \left(\sum_{j\in N} d_{ij}\right) - \sum_{h\in H}\sum_{i\in S_h}\sum_{j\in S_h} x_i^h x_j^h d_{ij}.$$

By taking into account (3.2), we have $\sum_{h\in H}\sum_{i\in S_h} x_i^h(\sum_{j\in N} d_{ij}) = \sum_{i\in N}\sum_{j\in N} d_{ij}$. Hence, we obtain finally

$$z_3 = \sum_{i\in N}\sum_{j\in N} d_{ij} - \sum_{h\in H}\sum_{i\in S_h}\sum_{j\in S_h} x_i^h x_j^h d_{ij}. \tag{3.10}$$

In order to apply the RLT to (NLF), we perform the following operations:

Reformulation phase. Since for all $h \in H$ and $i \in S_h$, $x_i^h \geqslant 0$ and $1 - x_i^h \geqslant 0$, we construct additional sets of constraints by the following:

$$x_i^h(1-x_j^h) \geqslant 0,\ \forall h \in H, i, j \in S_h. \tag{3.11}$$

$$x_j^h(1 - x_i^h) \geqslant 0,\ \forall h \in H, i, j \in S_h. \tag{3.12}$$

$$(1 - x_i^h)(1 - x_j^h) \geqslant 0,\ \forall h \in H, i, j \in S_h. \tag{3.13}$$

After developing, the previous inequalities become, respectively

$$x_i^h \geqslant x_i^h x_j^h,\ \forall h \in H, i, j \in S_h. \tag{3.14}$$

$$x_j^h \geqslant x_i^h x_j^h,\ \forall h \in H, i, j \in S_h. \tag{3.15}$$

$$1 + x_i^h x_j^h \geqslant x_i^h + x_j^h,\ \forall h \in H, i, j \in S_h. \tag{3.16}$$

Linearization phase. We define new decision variable $w_{ij}^h \geqslant 0$ such that $w_{ij}^h = x_i^h x_j^h$. This variable indicates whether both nodes i and j are connected to the same hub h or not. After substituting in the objective function (3.9), we obtain

$$\min z_3 = \sum_{i \in N} \sum_{j \in N} d_{ij} - \sum_{h \in H} \sum_{i \in S_h} \sum_{j \in S_h} w_{ij}^h d_{ij}. \tag{3.17}$$

After substituting $w_{ij}^h = x_i^h x_j^h$ in the inequalities (3.14)−(3.16), we get the new additional constraints to (3.2) as follows

$$x_i^h \geqslant w_{ij}^h,\ \forall h \in H, i, j \in S_h. \tag{3.18}$$

$$x_j^h \geqslant w_{ij}^h,\ \forall h \in H, i, j \in S_h. \tag{3.19}$$

$$1 + w_{ij}^h \geqslant x_i^h + x_j^h,\ \forall h \in H, i, j \in S_h. \tag{3.20}$$

Finally, the new formulation constituted by the objective function (3.17), the constraints (3.2), (3.18)-(3.20) and (3.6) is denoted by (RLTF). Also, the decision variables w_{ij}^h must be binary; that is,

$$w_{ij}^h \in \{0, 1\},\quad \forall h \in H, i, j \in S_h. \tag{3.21}$$

It is noteworthy that $w_{ij}^h = 0,\quad \forall i \notin S_h$, or $j \notin S_h$.

The constraints (3.18) and (3.19) impose the variables x_i^h and x_j^h to take the value 1 when both nodes i and j are connected to the same hub h ($w_{ij}^h = 1$). Conversely, the constraints (3.20) ensure that the variables w_{ij}^h take 1 when the nodes i and j are connected simultaneously to the same hub h.

This new formulation can be interpreted as being relative to the problem that maximizes the intra-hub traffic. In fact, since $\sum_{i \in N} \sum_{j \in N} d_{ij}$ is a constant quantity indicating the overall total flow, then $\min z_3$ is equivalent to maximize $\sum_{i \in S_h} \sum_{j \in S_h} w_{ij}^h d_{ij}$. The variable w_{ij}^h indicates whether both nodes i and j are connected to the same h, thus, not requiring the hub processing resources. Consequently, maximizing the total intra-hub traffic yields minimizing the use of the hubs processing resources.

This formulation (RLTF) is expected to be stronger than (PF), but at the expense of more variables and constraints. Next, we introduce a new formulation obtained by performing a partial RLT on the nonlinear formulation (NLF) yielding to a much more compact new mixed integer linear program by a factor of $|N|$.

3.3. Partial RLT based formulation

Instead of totally applying the RLT, we perform a partial Reformulation phase and a modified Linearization phase on the nonlinear formulation of the problem (NLF). With this new approach we provide a formulation with a lower size than (RLTF)

Reformulation phase. We multiply $x_j^h \geqslant 0$ by $(1 - x_i^h) \geqslant 0$ and obtain

$$x_j^h(1 - x_i^h) \geqslant 0,\ \forall h \in H, i, j \in S_h.$$

Thus

$$x_j^h \geqslant x_i^h x_j^h,\ \forall h \in H, i, j \in S_h. \tag{3.15}$$

Linearization phase. We have

$$\sum_{j\in N} x_i^h(1 - x_j^h)d_{ij} = x_i^h \sum_{j\in N} d_{ij} - \sum_{j\in S_h} x_i^h x_j^h d_{ij},\ \forall h \in H, i \in S_h.$$

We define a new continuous decision variable $f_i^h \geqslant 0$ as the total amount of flow from node i for all other nodes not connected to the hub h. It is written as follows

$$f_i^h = x_i^h \sum_{j\in N} d_{ij} - \sum_{j\in S_h} x_i^h x_j^h d_{ij},\ \forall h \in H, i \in S_h.$$

For sake of clarity, we denote by $O_i = \sum_{j\in N} d_{ij}$.

From (3.15) we have

$$-\sum_{j\in N} x_i^h x_j^h d_{ij} \geqslant -\sum_{j\in S_h} x_j^h d_{ij},\ \forall h \in H, i \in S_h.$$

We add $O_i x_i^h$ to the two members of the previous inequality and obtain

$$O_i x_i^h - \sum_{j\in N} x_i^h x_j^h d_{ij} \geqslant O_i x_i^h - \sum_{j\in S_h} x_j^h d_{ij},\ \forall h \in H, i \in S_h.$$

That is

$$f_i^h \geqslant O_i x_i^h - \sum_{j\in S_h} x_j^h d_{ij},\ \forall h \in H, i \in S_h.$$

Finally, the RLT, partially applied on (NLF), leads to the following new mixed integer program, denoted by (PRLTF):

$$\min z_4 = \sum_{h\in H} \sum_{i\in S_h} f_i^h \tag{3.22}$$

subject to

$$f_i^h \geqslant O_i x_i^h - \sum_{j\in S_h} d_{ij} x_j^h,\quad \forall h \in H, i \in S_h, \tag{3.23}$$

$$\sum_{h\in H_i} x_i^h = 1,\quad \forall i \in N. \tag{3.2}$$

$$x_i^h \in \{0, 1\},\quad \forall h \in H, i \in S_h \tag{3.6}$$

$$f_i^h \geqslant 0,\quad \forall h \in H, i \in S_h \tag{3.24}$$

In this formulation the number of continuous variables is $O(|N||H|)$ and the binary variables is $O(|N||H|)$. The number of constraints is $O(|N||H| + |N|)$.

4. Formulations comparison

Besides enumerating the number of variables and constraints of the formulations, we show mathematically that the formulation (RLTF) is stronger than (PF) and (PRLTF). For that purpose, we show that the polyhedron representing the solutions of the linear relaxation of (RLTF) is contained in that of (PF) and (PRLTF).

Theorem 4.1. *Let $\bar{z}_3$ (respectively $\bar{z}_2$) be the value of the optimal solution of the relaxed formulation (RLTF) (respectively (PF)), then $\bar{z}_3 \geqslant \bar{z}_2$*

Proof. Since the constraints $\sum_{h\in H_i} x_i^h = 1$ exist for both formulations, then it suffices to check that any solution verifying the constraints (3.18)−(3.20) of (RLTF) is also feasible for (3.3)−(3.5) of (PF). Consider the variables t_{ij}^h that take the value 1 when the node $i \in S_h$ is connected to the hub h, and $j \in S_h$ connected to another hub. Then we have:

$$t_{ij}^h = x_i^h(1-x_j^h), \quad \forall h \in H, i, j \in S_h.$$

The decision variables p_{ij} of (PF) take the value 1 when the nodes i and j are connected to two different hubs. Then, for any pair of nodes $(i,j) \in N^2$, if $\exists h \in H$ such that $t_{ij}^h = 1$, then $p_{ij} = 1$. This can be expressed by

$$p_{ij} = \sum_{h\in H} t_{ij}^h \tag{4.1}$$

A disaggregated representation of (4.1) is $p_{ij} \geqslant t_{ij}^h,\ \forall h \in H, i \in S_h, j \in S_h$; that is, $p_{ij} \geqslant x_i^h(1-x_j^h)$

We add $x_j^h - x_i^h$ to both sides of the inequality above and by taking into account that $w_{ij}^h = x_i^h x_j^h$, we obtain

$$p_{ij} - x_i^h + x_j^h \geqslant x_j^h - w_{ij}^h$$

Since $w_{ij}^h \leqslant x_j^h$ from (RLTF), then $p_{ij} - x_i^h + x_j^h \geqslant 0$; that is, $p_{ij} \geqslant x_i^h - x_j^h$.

Likewise, we have $p_{ij} \geqslant x_j^h(1-x_i^h)$. By performing similarly as above we obtain $p_{ij} - x_j^h + x_i^h \geqslant x_i^h - w_{ij}^h$. With reference to (3.18) of (RLTF), we get the inequality (3.4) of (PF).

In addition, from $p_{ij} \geqslant x_i^h(1-x_j^h)$, we get $p_{ij} \geqslant x_i^h - w_{ij}^h$. But since $w_{ij}^h \leqslant x_i^h$ of the formulation (RLTF), we deduce that $0 \leqslant p_{ij} \leqslant 1$.

On another hand, for the case where $j \notin S_h$, we have obviously $x_j^h = 0$ and we obtain immediately $p_{ij} \geqslant x_i^h$ from $p_{ij} \geqslant x_i^h(1-x_j^h)$.

Finally, the feasible solution of (RLTF) verifies the formulation (PF). □

Theorem 4.2. *Let $\bar{z}_4$ (respectively $\bar{z}_3$) be the value of the optimal of the relaxed formulation (PRLTF) (respectively (RLTF)), then $\bar{z}_3 \geqslant \bar{z}_4$*

Proof. Since the constraints $\sum_{h\in H_i} x_i^h = 1$ exist for the both formulations, then it suffices to check that any solution verifying the constraints (3.18)−(3.20) of (RLTF) is also feasible for (3.23) of (PRLTF). In fact, since $x_j^h \geqslant w_{ij}^h,\ \forall h \in H, i \in S_h$, then $x_j^h \geqslant x_i^h x_j^h,\ \forall h \in H, i \in S_h$, therefore

$$O_i x_i^h - \sum_{j\in N} x_i^h x_j^h d_{ij} \geqslant O_i x_i^h - \sum_{j\in S_h} x_j^h d_{ij},\ \forall h \in H, i \in S_h.$$

That is

$$f_i^h \geqslant O_i x_i^h - \sum_{j\in S_h} x_j^h d_{ij},\ \forall h \in H, i \in S_h.$$

Finally the result follows. □

TABLE 1. Formulations size comparison.

Formul.	#BVar	#CVar	#Constraints
(PF)	$O(\|N\|^2 + \|N\|\|H\|)$	–	$O(\|N\|^2\|H\| + \|N\|)$
(RLTF)	$O(\|N\|^2\|H\| + \|N\|\|H\|)$	–	$O(\|N\|^2\|H\| + \|N\|)$
(PRLTF)	$O(\|N\|\|H\|)$	$O(\|N\|\|H\|)$	$O(\|N\|\|H\| + \|N\|)$

Table 1 summarizes the number of the binary variables, denoted by #BVar, the continuous variables, designated by #CVar and the constraints denoted by #Constraints.

Actually, the number of variables and constraints are less than those specified in Table 1, since $S_h \subset N$ and $H_i \subset H$. Nevertheless, we mention these values as orders of magnitude for comparison purpose only and to justify the relative performance of each formulation.

From the propositions above and since the (PRLTF) has the least number of variables and constraints, it is expected to be the most effective. For the formulations (RLTF) and (PF), it is interesting to consider a computational experiments comparison between them in order to point out the tradeoff between the number of variables and linear relaxation quality.

It is noteworthy that an alternative way to obtain formulation (PRLTF) is to aggregate the flow variables of the formulation given in [13]. Indeed, by denoting ϕ^i_{hk} as the total flow issued by the node i and circulating between the hubs h and k, a valid mixed integer program of ANHP can be derived from [13] as follows:

$$\min \sum_{i\in N}\sum_{h\in H}\sum_{k\in H}\phi^i_{hk} \tag{4.2}$$

subject to

$$O_i x^h_i - \sum_{j\in S_h} d_{ij}x^h_j = \sum_{k\in H}\phi^i_{hk} - \sum_{k\in H}\phi^i_{kh}, \quad \forall h\in H, i\in S_h, \tag{4.3}$$

$$\sum_{h\in H_i} x^h_i = 1, \quad \forall i\in N, \tag{3.2}$$

$$x^h_i \in \{0,1\}, \quad \forall h\in H, i\in S_h, \tag{3.6}$$

$$\phi^i_{hk} \geqslant 0, \quad \forall h,k\in H, i\in N, \tag{4.4}$$

By aggregating the flow variables $\sum_{k\in H}\phi^i_{hk} = f^h_i$ we obtain easily (PRLTF). Nevertheless, the two formulations that we derived from the nonlinear program by applying RLT are more rigorous and systematic. In other terms, we recover automatically formulation of [13] by applying a partial RLT to the nonlinear formulation of ANHP. Besides, we obtained stronger formulation for ANHP as we have shown in this Section that (RLTF) is stronger than (PF) and (PRLTF).

5. Double allocation mode

The modeling achieved above is relative to the single hub allocation mode. As mentioned in Section 1, although a given node is linked physically to two hubs, the total traffic issued by the node is processed by only one hub while the second one is for standby purpose. When a failure occurs on the active link or on the current hub, the whole traffic is switched over the standby link with the other hub.

In this section, we consider an alternative mode, the so-called load-sharing, and we compare it to the single allocation mode. In this mode, each node is connected to two hubs *via* two active links simultaneously. Each link holds the half of the traffic issued by a given node and going to another different node. This configuration presents the advantage of switching over 50% of the traffic instead of the total traffic. This variant is different from the multiple allocation mode as defined in [14]. Indeed, the traffic originated by a node must be distributed to the links with the corresponding hub with equal share. For instance, if there are four links, then each link must hold exactly 25% of the traffic. In this case, the formulation (PRLTF) can be modified easily to fit the double allocation mode. The hubs equipment existing in the operator network permit the implementation of such an allocation mode.

We denote by $q_i^h \geqslant 0$ the total traffic issued by the node i and processed by the hub h with the double allocation mode. Then, the following inequality holds:

$$\frac{O_i}{2}x_i^h - \sum_{j \in S_h} \frac{d_{ij}}{2} x_j^h \leqslant q_i^h, \quad \forall h \in H, i \in S_h,$$

Then, the formulation of this variant, denoted by (PRLTFDA) can be written as follows:

$$\min \sum_{i \in N} \sum_{h \in H} q_i^h \tag{5.1}$$

subject to

$$O_i x_i^h - \sum_{j \in S_h} d_{ij} x_j^h \leqslant 2 q_i^h, \quad \forall h \in H, i \in S_h, \tag{5.2}$$

$$\sum_{h \in H_i} x_i^h = 2, \quad \forall i \in N, \tag{5.3}$$

$$x_i^h \in \{0,1\}, \quad \forall h \in H, i \in S_h, \tag{3.6}$$

$$q_i^h \geqslant 0, \quad \forall h \in H, i \in S_h. \tag{5.4}$$

The constraint (5.2) computes the total traffic processed by the hub coming from the node to which it is allocated. This traffic is half the total traffic issued by the node. The constraint (5.3) states that each node must be connected to two different hubs *via* two active links. Note that the allocation variables x_i^h remain binary.

6. Computational experiments

All the computational experiments were made with OPL/Cplex 12.2 package on a machine that has Intel Xeon 3,3 GHz CPU and 12 Gb of RAM. We set the running time limit to 1 hour.

We conduct the experiments on the three formulations with the same data sets in order to compare the computational effort required. The data are randomly generated but based on the real life network of *Tunisie Telecom*.

The network is composed of 110 nodes and 5 hubs that transform the voice into IP packets and are situated at the frontier of the IP transport backbone. The possible links between the nodes and the hubs are randomly generated taking into account that each node can be connected to, at least, two hubs for security reasons. To this end, a number; either 0 or 1, is generated randomly for each pair i, h. If for any given node i, the number of different possible hubs is less than 2, then the generation is repeated for that node, until obtaining more than 2 different hubs.

The elements d_{ij} of the traffic matrix are randomly generated between 1 and 10 embracing the actual traffic pattern of the operator.

Although the actual network has 110 terminal nodes, we generate other instances corresponding to a number of nodes ranging from 90 to 120 with a fixed number of hubs to 5 in order to test the performance of the formulations on wider basis. In fact, for each possible choice of $|N|$, we generate 10 different instances.

We run three series of experiments: the first concerns the quality of the lower bounds of the three formulations. The second focuses on the computational effort required and the gain in IP traffic when using the optimization

TABLE 2. Relative gaps comparison for $|N| = 90$.

Inst.	S	RG_{RLTF}	RG_{PRLTF}	RG_{PF}
1	11 124	4.962	26.438	27.490
2	11 238	4.850	26.945	29.576
3	9122	0.636	28.910	16.086
4	10 434	4.246	29.714	28.314
5	11 174	6.600	25.008	32.377
6	9508	0.000	22.841	19.303
7	11 512	7.056	31.499	34.089
8	11 141	2.965	27.537	27.893
9	11 299	2.770	27.813	28.993
10	11 606	6.721	24.529	33.203

approach instead of the current allocation approach of the operator. In the third, we considered different numbers of hubs ranging from 6 to 8, in order to test its influence on the computational time. For that purpose, we take the traffic data of the 100 and 110 terminal nodes instances and generate randomly three sets S_h corresponding to $|H| = 6, 7$ and 8. In addition, we present results about the failure mode, in which a hub fails, and the altered traffic must be switched over the backup hubs. Finally, we compare experimentally between the single and the double allocation modes by considering the formulations (PRLTF) and (PRLTFDA).

6.1. Quality of the lower bounds

In order to assess the strength of the formulations (PF), (RLTF) and (PRLTF), we run their corresponding MIP and linear relaxations and note the value of the relative gaps between the values of the optimal integer solutions and the relaxed optimal solutions. The closer to zero is the gap, the stronger is the formulation.

We consider instances relative to 5 hubs and number of terminal nodes ranging from 90 to 120 nodes. For each number of nodes 90, 100, 110 and 120, the results are reported respectively in Table 2, 3, 4 and 5. In order to compute the relative gap, we run the formulation (PRLTF) which was able to provide the optimal solution for all instances within the time limit. By denoting S as the value of the optimal integer solution provided by (PRLTF), Rlx_{PF}, Rlx_{RLTF} and Rlx_{PRLTF} as the value of the linear relaxation of the formulations (PF), (RLTF) and (PRLTF) respectively, the relative gap RG_{PF}, RG_{RLTF} and RG_{PRLTF}, are computed as follows:

$$RG_n = 100 \times \frac{S - Rlx_n}{S}, \tag{6.1}$$

where $n = PF$, $RLTF$ and $PRLTF$.

The columns of Table 2 to Table 5 are the following:

- Inst: the identification number of the instance,
- S: the value of the optimal integer solution provided by (PRLTF),
- RG_n: the relative gap of formulation n computed by equation (6.1).

The results confirm the strength of the formulation (RLTF) obtained by applying RLT on the nonlinear formulation (NLF). As stated in [23, 24], RLT leads to stronger compact formulations. Indeed, with (RLTF), 24 instances out of 40 have a relative gap inferior to 5%. Moreover, the maximum relative gap is within 8%.

Furthermore, the strengths of the formulations (PF) and (PRLTF) are incomparable, as for some instances the gap of (PF) is superior or inferior to that of (PRLTF). But, it is noteworthy that in most of the instances (PRLTF) has better lower bound than (PF).

6.2. Computational effort for the formulations

In this set of experiments, we test the performance of the formulations in terms of the computational time necessary to reach the optimal integer solution. The computational times for formulations (PF), (RLTF) and

TABLE 3. Relative gaps comparison for $|N| = 100$.

Inst.	S	RG_{RLTF}	RG_{PRLTF}	RG_{PF}
1	13 844	2.774	25.917	27.133
2	14 879	6.240	28.353	32.215
3	13 657	4.503	30.115	31.352
4	12 289	0.000	22.062	22.109
5	14 131	8.074	33.708	32.089
6	14 592	6.387	27.298	32.872
7	12 043	0.316	22.921	23.003
8	13 444	4.227	25.979	27.777
9	13 298	2.876	22.454	26.570
10	12 843	1.830	28.086	25.770

TABLE 4. Relative gaps comparison for $|N| = 110$.

Inst.	S	RG_{RLTF}	RG_{PRLTF}	RG_{PF}
1	16 725	0.043	24.550	25.935
2	15 753	2.210	31.722	27.809
3	17 363	6.497	30.395	33.546
4	16 666	7.331	28.746	32.144
5	17 543	5.626	32.659	32.463
6	13 440	0.030	23.782	14.493
7	16 944	4.543	29.926	31.120
8	16 497	4.402	27.145	30.085
9	17 917	7.056	31.201	34.554
10	15 349	3.429	28.132	29.000

TABLE 5. Relative gaps comparison for $|N| = 120$.

Inst.	S	RG_{RLTF}	RG_{PRLTF}	RG_{PF}
1	19 432	2.385	24.360	26.950
2	19 036	0.701	29.508	26.583
3	20 870	6.557	29.085	32.966
4	20 408	5.762	25.850	32.724
5	19 805	5.254	25.414	30.667
6	19 979	6.402	28.071	30.555
7	20 761	3.687	27.044	30.278
8	19 428	3.710	29.328	30.308
9	20 953	5.720	31.193	32.120
10	21 118	8.171	32.116	34.082

(PRLTF) are reported from Table 6 to Table 9 for different values of $|N|$. Besides Inst and S, defined in Section 5.1, the columns of the Tables 6 to 9 are the following:

- T_{PF}, T_{RLTF} and T_{PRLTF}: the running time in seconds of, respectively, the formulations (PF), (RLTF) and (PRLTF),
- G_{PF}, G_{RLTF} and G_{PRLTF}: the gap between the lower bound and the best integer solution found in % for the formulations (PF), (RLTF) and (PRLTF). When the gap is zero, then the corresponding formulation reaches the optimal solution.

TABLE 6. Results comparison for $|N| = 90$.

Inst	S	T_{PF}	G_{PF}	T_{RLTF}	G_{RLTF}	T_{PRLTF}	G_{PRLTF}
1	11 124	31,1	0	17,7	0	4,8	0
2	11 238	50,9	0	26,8	0	8,2	0
3	9122	2,8	0	1,7	0	0,6	0
4	10 434	37,4	0	13,7	0	2,0	0
5	11 174	213,4	0	192,7	0	18,8	0
6	9508	1,3	0	0,9	0	0,6	0
7	11 512	409,6	0	437,4	0	60,6	0
8	11 141	30,6	0	10,1	0	4,7	0
9	11 299	23,9	0	11,6	0	2,4	0
10	11 606	270,9	0	278,6	0	18,8	0

TABLE 7. Results comparison for $|N| = 100$.

Inst	S	T_{PF}	G_{PF}	T_{RLTF}	G_{RLTF}	T_{PRLTF}	G_{PRLTF}
1	13 844	30,4	0	20,2	0	3,1	0
2	14 879	372,0	0	581,1	0	29,2	0
3	13 657	63,8	0	47,7	0	14,0	0
4	12 289	2,5	0	2,0	0	1,0	0
5	14 131	758,3	0	507,1	0	78,9	0
6	14 592	409,9	0	478,5	0	39,2	0
7	12 043	5,0	0	4,6	0	2,1	0
8	13 444	40,0	0	20,1	0	7,2	0
9	13 298	29,8	0	14,1	0	5,8	0
10	12 843	26,4	0	17,4	0	2,3	0

TABLE 8. Results comparison for $|N| = 110$.

Inst	S	T_{PF}	G_{PF}	T_{RLTF}	G_{RLTF}	T_{PRLTF}	G_{PRLTF}
1	16 725	53,4	0	24,1	0	4,3	0
2	15 753	41,0	0	21,1	0	9,1	0
3	17 363	1062,2	0	1008,1	0	240,1	0
4	16 666	1549,3	0	579,1	0	96,2	0
5	17 543	819,5	0	450,7	0	59,6	0
6	13 440	2,4	0	2,3	0	0,7	0
7	16 944	124,3	0	363,9	0	25,5	0
8	16 497	148,2	0	58,6	0	25,2	0
9	17 917	1562,0	0	1991,9	0	376,9	0
10	15 349	77,0	0	50,0	0	17,8	0

Although (RLTF) has much more binary variables, it has better running time than the formulation (PF) in many instances. In fact, (RLTF) gives a tighter linear relaxation making the branch-and-cut algorithm of the Cplex solver to perform better.

The formulation (PRLTF) turned out to be the most computationally effective when compared to (PF) and (RLTF). Indeed, it offers the best trade-off between size and quality of the lower bound. It has the least number of variables and constraints and gives the optimal solution with the least running times.

Particularly, for *Tunisie Telecom* network ($|N| = 110$) with the formulation (PRLTF), the running times do not exceed 7 min, whereas they are of about 30 min for (PF) and (RLTF).

In addition we compare between a random nodes' allocation, which is currently used by the operator, and the optimal allocation provided by (PRLTF). The results are reported in Figure 2. The results point out clearly the

TABLE 9. Results comparison for $|N| = 120$.

Inst	S	T_{PF}	G_{PF}	T_{RLTF}	G_{RLTF}	T_{PRLTF}	G_{PRLTF}
1	19 432	60,2	0	32,0	0	5,3	0
2	19 036	20,0	0	9,9	0	3,8	0
3	20 870	2684,9	0	2839,9	0	310,0	0
4	20 408	3539,8	0	1040,1	0	212,9	0
5	19 805	1078,6	0	1580,7	0	66,6	0
6	19 979	1158,3	0	1563,8	0	134,6	0
7	20 761	152,4	0	86,6	0	157,6	0
8	19 428	126,0	0	70,0	0	27,9	0
9	20 953	1966,5	0	2981,3	0	218,1	0
10	21 118	>1 h	6	>1 h	3	1507,4	0

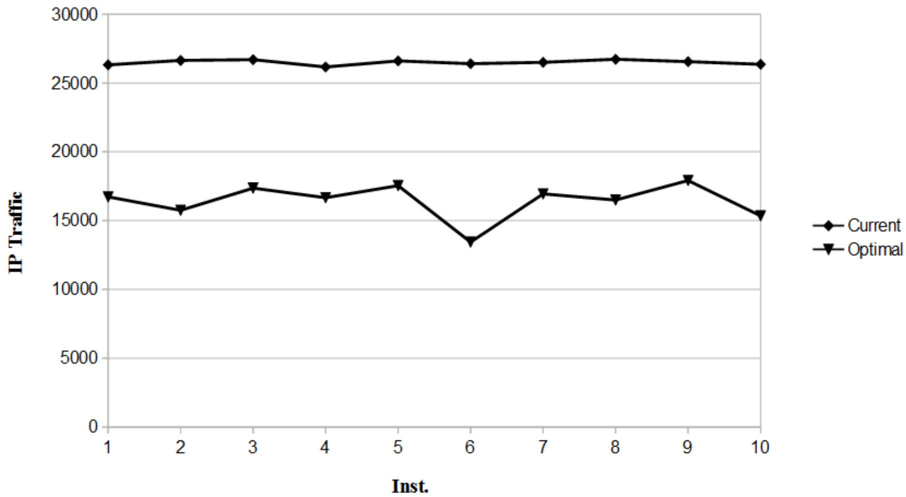

FIGURE 2. Optimal *vs.* current allocation.

saving of the IP transformation operations made by the optimal allocation mode compared to the non-optimal one.

6.3. Varying the number of hubs

In this set of experiments, we run the MIP formulations (PF), (RLTF) and (PRLTF) by taking the instances corresponding to $|N| = 100$ and 110 and by varying $|H| = 6, 7$ and 8. The results are reported in Table 10 to Table 12 for $|N| = 100$ and Table 13 to Table 15 for $|N| = 110$. The columns of the tables are the following:

- S_{PF}, S_{RLTF} and S_{PRLTF}: the value of the solution of, respectively, the formulations (PF), (RLTF) and (PRLTF),
- T_{PF}, T_{RLTF} and T_{PRLTF}: the running time of, respectively, the formulations (PF), (RLTF) and (PRLTF) in seconds,
- G_{PF}, G_{RLTF} and G_{PRLTF}: the gap between the lower bound and the best integer solution found in % for the formulations (PF), (RLTF) and (PRLTF). When the gap is zero, then the corresponding formulation reaches the optimal solution. Conversely, when this gap is nonzero and the running time does not exceed 3600 s, then this means that Cplex went to out-of-memory status and the case is marked with asterisk ($*$).

Table 10. Results comparison for $|H| = 6$ and $|N| = 100$.

Inst	S_{PF}	T_{PF}	G_{PF}	S_{RLTF}	T_{RLTF}	G_{RLTF}	S_{PRLTF}	T_{PRLTF}	G_{PRLTF}
1	15 088	128.6	0	15 088	41.8	0	15 088	37.6	0
2	15 675	216.6	0	15 675	200.4	0	15 675	44.8	0
3	14 758	1164.0	0	14 758	1409.2	0	14 758	169.6	0
4	14 655	249.7	0	14 655	270.5	0	14 655	36.4	0
5	15 307	483.2	0	15 307	282.5	0	15 307	156.0	0
6	15 562	67.4	0	15 562	54.9	0	15 562	6.6	0
7	15 659	>1 h	0.989	15833	>1 h	3.086	15659	1061.6	0
8	14 667	193.9	0	14 667	83.6	0	14 667	16.1	0
9	14 945	134.2	0	14 945	124.0	0	14 945	25.8	0
10	14 916	268.0	0	14 916	151.3	0	14 916	68.3	0

Table 11. Results comparison for $|H| = 7$ and $|N| = 100$.

Inst	S_{PF}	T_{PF}	G_{PF}	S_{RLTF}	T_{RLTF}	G_{RLTF}	S_{PRLTF}	T_{PRLTF}	G_{PRLTF}
1	16 070	>1 h	2.929	16 062	>1 h	3.013	15 930	1420.6	0
2	16 697	>1 h	3.162	16 697	>1 h	3.535	16 666	2819.2	0
3	14 734	462.7	0	14 734	432.5	0	14 734	93.2	0
4	14 287	421.1	0	14 287	177.6	0	14 287	38.6	0
5	14 365	3.4	0	14 365	7.1	0	14 365	2.6	0
6	16 114	2073.0	0	16 114	3137.5	0	16 114	514.3	0
7	15 798	1728.7	0	15 798	2398.1	0	15 798	236.4	0
8	15 261	132.9	0	15 261	107.9	0	15 261	13.2	0
9	14 117	142.2	0	14 117	38.8	0	14 117	21.0	0
10	14 230	121.4	0	14 230	27.2	0	14 230	13.2	0

Table 12. Results comparison for $|H| = 8$ and $|N| = 100$

Inst	S_{PF}	T_{PF}	G_{PF}	S_{RLTF}	T_{RLTF}	G_{RLTF}	S_{PRLTF}	T_{PRLTF}	G_{PRLTF}
1	15 307	533.7	0	15 307	442.9	0	15 307	121.5	0
2	14 954	255.2	0	14 954	144.9	0	14 954	28.1	0
3	14 615	32.4	0	14 615	17.4	0	14 615	4.8	0
4	16 226	>1 h	2.934	16 343	>1 h	4.010	16 226	>1 h	0.609
5	15 481	405.1	0	15 481	235.7	0	15 481	98.4	0
6	15 142	>1 h	2.458	15 207	>1 h	3.074	15 015	581.8	0
7	15 097	421.2	0	15 097	146.3	0	15 097	36.9	0
8	14 500	135.1	0	14 500	31.0	0	14 500	10.0	0
9	17 196	>1 h	3.946	17 207	>1 h	4.554	17 196	>1 h	2.37
10	15 765	1645.9	0	15 765	436.1	0	15 765	131.2	0

We notice that the computational effort augments when $|H|$ increases, thus increasing the size of the problem. Nonetheless, the formulation (PRLTF) still performs better than (PF) and (RLTF). In addition, when in some instances (PRLTF) does not give the optimal solution within the time limit, it provides the solution with the best gap.

6.4. Failure modes

The failure mode is associated to a failure that can occur on a given hub. All the active links with that hub are disrupted and the affected traffic is switched over the standby links. In fact, upon a hub failure, the terminal

TABLE 13. Results comparison for $|H| = 6$ and $|N| = 110$.

Inst	S_{PF}	T_{PF}	G_{PF}	S_{RLTF}	T_{RLTF}	G_{RLTF}	S_{PRLTF}	T_{PRLTF}	G_{PRLTF}
1	18 129	>1 h	8.516	18 016	125.4	0	18 016	55.9	0
2	16 612	1414.99	0	16 612	3.6	0	16 612	1.7	0
3	18 052	>1 h	12.266	18 052	269.6	0	18 052	51.9	0
4	17 298	>1 h	11.670	17 289	30.2	0	17 289	10.6	0
5	18 329	>1 h	11.717	18 329	1043.7	0	18 329	97.0	0
6	20 072	>1 h	15.808	19 460	>1 h	2.51	19 436	2450.5	0
7	18 617	>1 h	12.992	18 678	>1 h	1.71	18 617	446.6	0
8	19 550	>1 h	18.547	18 434	1615.1	0	18 434	220.5	0
9	18 727	>1 h	10.095	18 727	84.8	0	18 727	17.9	0
10	17 597	>1 h	13.963	17 455	669.7	0	17 455	108.6	0

TABLE 14. Results comparison for $|H| = 7$ and $|N| = 110$.

Inst	S_{PF}	T_{PF}	G_{PF}	S_{RLTF}	T_{RLTF}	G_{RLTF}	S_{PRLTF}	T_{PRLTF}	G_{PRLTF}
1	20 728	>1 h	27.2	19 309	>1 h	4.605	19 242	>1 h	0.829
2	19 201	>1 h	11.2	19 173	134.7	0	19 173	19.7	0
3	18 690	>1 h	24.9	18 310	2976.2	0	18 310	247.3	0
4	19 118	>1 h	23.4	18 748	1795.8	0	18 748	215.4	0
5	21 089	>1 h	21.0	19 840	>1 h	2.842	19 840	1846.1	0
6	20 169	>1 h	22.6	20 063	>1 h	3.477	20 126	1706.8	2.501*
7	19 243	>1 h	21.9	18 792	268.7	0	18 792	75.7	0
8	17 908	>1 h	8.7	17 893	23.1	0	17 893	16.0	0
9	18 575	>1 h	23.1	18 427	2771.7	0	18 427	339.4	0
10	17 858	>1 h	9.3	17 794	31.0	0	17 794	8.8	0

TABLE 15. Results comparison for $|H| = 8$ and $|N| = 110$.

Inst	S_{PF}	T_{PF}	G_{PF}	S_{RLTF}	T_{RLTF}	G_{RLTF}	S_{PRLTF}	T_{PRLTF}	G_{PRLTF}
1	17 830	>1 h	24.304	17 447	24.9	0	17 447	31.5	0
2	21 901	>1 h	30.594	20 593	>1 h	3.738	20 593	>1 h	1.308
3	18 813	>1 h	28.508	18 588	>1 h	2.732	18 413	603.7	0
4	20 388	1956.8	31.750*	19865	>1 h	5.414	19 564	>1 h	1.346
5	19 560	>1 h	26.920	19 064	1012.7	0	19 064	164.6	0
6	21 561	>1 h	28.404	21 459	>1 h	4.898	21 264	1825.1	3.450*
7	19 686	>1 h	29.104	18 519	488.8	0	18 519	215.2	0
8	20 146	820.0	31.580*	19 406	>1 h	4.263	19108	731.9	0
9	20 560	>1 h	19.511	19 439	89.6	0	19 439	12.4	0
10	18 733	>1 h	27.300	18 081	213.4	0	18 081	59.2	0

nodes associated with that hub detect the failure through Sigtran protocol. They switch over the traffic from the active links to the standby links and the traffic is recovered.

In this experiments, we considered the network of *Tunisie Telecom* with 110 nodes and 5 hubs. We assume one failure occurring on one hub at a time specified in the first column of Table 16. We compare the values of the optimal solution and the running times between the failure modes (each one associated to a given hub) and the normal mode, in which all the hubs work normally.

All the experiments have reached the optimal solution within relatively small running times. The model with the failure mode is rapidly obtained because of the variable fixing due to the failure of the hub h. In fact, when a given h fails then the associated variables f_i^h and x_i^h are both fixed to 0. The solution associated to the failure mode is close to that of the normal situation, except for few instances.

TABLE 16. Failure mode.

h	Inst	Sol_h	G_h	S	Gap_h	T_h
	1	16 816	0	16 725	0.54	1.6
	2	17 084	0	15 753	8.45	2.3
	3	17 612	0	17 363	1.43	4.3
	4	16 785	0	16 666	0.71	3.4
	5	18 337	0	17 543	4.53	14.3
	6	13 440	0	13 440	0.00	0.9
	7	17 188	0	16 944	1.44	2.2
	8	18 094	0	16 497	9.68	15.0
	9	17 917	0	17 917	0.00	5.3
	10	15 515	0	15 349	1.08	2.2
2	1	16 725	0	16 725	0.00	1.5
	2	15 763	0	15 753	0.06	2.7
	3	17 363	0	17 363	0.00	4.3
	4	16 666	0	16 666	0.00	2.3
	5	17643	0	17 543	0.57	6.3
	6	13 440	0	13440	0.00	0.6
	7	16 944	0	16 944	0.00	2.5
	8	16 497	0	16 497	0.00	3.0
	9	18 540	0	17 917	3.48	90.5
	10	15 349	0	15 349	0.00	2.5
3	1	16 725	0	16 725	0.00	2.2
	2	15 753	0	15 753	0.00	1.8
	3	17 553	0	17 363	1.09	2.8
	4	16 672	0	16 666	0.04	2.7
	5	17 543	0	17 543	0.00	1.7
	6	13 441	0	13 440	0.01	0.6
	7	16 944	0	16 944	0.00	6.6
	8	16 549	0	16 497	0.32	3.3
	9	18 023	0	17 917	0.59	10.2
	10	16 707	0	15 349	8.85	5.8
4	1	18 349	0	16 725	9.71	2.1
	2	16 078	0	1 5753	2.06	1.5
	3	17 363	0	17 363	0.00	4.7
	4	16 666	0	16 666	0.00	3.7
	5	17 543	0	17 543	0.00	3.6
	6	13 911	0	13 440	3.5	1.4
	7	16 954	0	16 944	0.06	2.1
	8	16 497	0	16 497	0.00	2.3
	9	17 917	0	17 917	0.00	9.6
	10	15 390	0	15 349	0.27	1.4
5	1	17 058	0	16 725	1.99	0.8
	2	15 753	0	15 753	0.00	1.9
	3	17 392	0	17 363	0.17	4.2
	4	16 672	0	16 666	0.04	14.1
	5	17 933	0	17 543	2.22	3.5
	6	18 213	0	13 440	35.51	1.8
	7	18 532	0	16 944	9.37	13.1
	8	16 667	0	16 497	1.03	2.3
	9	18 114	0	17 917	1.1	5.7
	10	15 349	0	15 349	0.00	4.1

In Table 16, the following columns are presented:

- h: the hub subject to a failure,
- Inst.: the identification number of the instance,
- Sol_h: the value of the optimal solution associated to the failure of the hub h,
- G_h: the gap between the best integer solution found and the lower bound returned by Cplex for the failure situation associated to the hub h. When it is zero, it indicates that the optimal solution is reached,
- S: the value of the optimal solution for the normal situation,
- Gap_h: the gap between the solutions of the normal situation and the failure mode associated to the hub h, computed as follows

$$Gap_h = 100 \times \frac{Sol_h - S}{S}$$

- T_h: the computational time associated to the failure of the hub h in seconds.

It turned out that the model gives rapidly the optimal new allocation of the nodes to the hubs in order to minimize the total traffic processed in the failure mode. In addition, the augmentation of the IP traffic due to the failure mode remains relatively small.

6.5. Double allocation

The computational results relative to the double allocation are presented in Table 17 with $|N| = 110$ and $|H| = 5$ (*Tunisie Telecom's* network configuration). Therein, we compare between the single allocation and the double allocation configurations in terms of the values of the solutions and the maximum load of traffic processed by each hub. We denote by:

- Inst.: the identification number of the instance,
- S: the value of the optimal solution of the single allocation,
- Sol_d: the value of the optimal solution of the double allocation,
- Gap_d: the gap between S and Sol_d computed as follows:

$$Gap_d = 100 \times \frac{S - Sol_d}{S}$$

- $C_{\max}$: the maximum value of the total processed traffic over the hubs in the single allocation mode, calculated as follows:

$$C_{\max} \geqslant \sum_{i \in S_h} f_i^h, \quad \forall h \in H.$$

- C_{maxd}: the maximum value of the total processed traffic over the hubs in the double allocation mode, calculated as follows:

$$C_{\text{maxd}} \geqslant \sum_{i \in S_h} q_i^h, \quad \forall h \in H.$$

- T_d: the computational time for the double allocation mode in seconds.

The double allocation configuration turned out to be more advantageous than the single allocation. In fact, in almost all the instances, the total traffic injected into the backbone with the double allocation is inferior to that of the single allocation mode. In addition, the maximum value of the load over all the hubs is almost the half of that relative to the single allocation. Finally, the computational time is small for the double allocation when compared to that of the single one, permitting rapid optimal solutions.

TABLE 17. Results comparison between single *vs.* double allocation.

Inst.	S	Sol_d	Gap_d	$C_{\max}$	C_{maxd}	T_d
1	16 725	16268.0	2.73	8708	4354	0.9
2	1 5753	15424.0	2.09	6948	4785	1.0
3	17 363	16298.0	6.13	9133	4567	2.5
4	16 666	15488.5	7.07	8124	4062	1.8
5	17 543	16622.5	5.25	8864	4474	1.9
6	13 440	14805.5	−10.16	6854	4127	0.8
7	16 944	16287.5	3,87	7452	4617	2.0
8	16 497	16023.0	2.87	7734	4347	5.7
9	17 917	16877.0	5.80	9496	4748	6.1
10	15 349	14853.0	3.23	6986	3990	2.0

7. Conclusion

In the present paper, we have addressed the hub allocation problem consisting of allocating the terminal nodes to the hubs in order to minimize the total flow processed by the hubs. This objective is motivated by the fact that processing flow on hubs induces some undesirable delay, especially for real time services of the telecommunications applications.

We have given three compact formulations in order to overcome the difficulty related to the real life networks. One formulation was straightforwardly devised and the other two are variants of the application of the RLT. The first variant proved to be very strong as the lower bound is close to the optimal solution.

We conducted many computational experiments on real life network of the Tunisian operator *Tunisie Telecom* that confirmed the efficiency of our proposed formulations. Particularly, the last one was effective and was obtained by partial RLT in order to have the resulting integer linear program of the least size.

We investigated and compared between the single and double allocation modes of the nodes to the hubs and pointed out experimentally the advantages of the latter mode.

We have provided the network managers of the operator with a compact formulation, efficient and easy to implement in order to test different configurations and to adapt to changing traffic patterns or to a failure occurring on a given hub. This work has facilitated the management of their network on optimal basis.

Acknowledgements. Dr. Mehdi Mrad gratefully acknowledges the technical and financial support of Research Center of College of Engineering, Deanship of Scientific Research, King Saud University.

References

[1] B. Addis, G. Carello and A. Ceselli, Exactly solving a two-level location problem with modular node capacities. *Networks* **59** (2012) 161–180.

[2] A. Alumur Sibel, Y. Kara Bahar and E. Karasan Oya, Multimodal hub location and hub network design. *Omega* **40** (2012) 927–939.

[3] S. Alumur and B.Y. Kara, Network hub location problems: The state of the art. *Eur. J. Oper. Res.* **190** (2008) 1–21.

[4] A. Balma, A.B. Hadj−Alouane and N. Ben Hadj-Alouane, Optimizing voice processing resources of the Trunk Gateways in NGN networks. In: *Proc. of International Multi-Conference on Complexity, Informatics and Cybernetics: IMCIC 2014* (2014) 152–155.

[5] I. Contreras, J.F. Cordeau and G. Laporte, Benders Decomposition for Large-Scale Uncapacitated Hub Location. *Oper. Res.* **59** (2011) 1477–1490.

[6] A.M. Campbell, T.J. Lowe and L. Zhang, The p-hub center allocation problem. *Eur. J. Operat. Res.* **176** (2007) 819–835, ISSN 0377–2217.

[7] J.F. Campbell and M.E. OKelly, Twenty-Five Years of Hub Location Research. *Trans. Sci.* **46** (2012) 153–169.

[8] Ch. Jeng−Fung, A hybrid heuristic for the uncapacitated single allocation hub location problem. *Omega* **35** (2007) 211–220.

[9] I. Contreras and E. Fernandez, General network design: A unified view of combined location and network design problem. *Eur. J. Operat. Res.* **219** (2012) 680–697.

[10] I. Correia, S. Nickel and F. Saldanha-da-Gama, The capacitated single-allocation hub location problem revisited: A note on a classical formulation. *Eur. J. Oper. Res.* **207** (2010) 92–96.

[11] I. Correia, S. Nickel and F. Saldanha-da-Gama, Single-assignment hub location problems with multiple capacity levels. *Transp. Res. Part B* **44** (2010) 1047–1066.

[12] M.G. Costa, M.E. Captivo and J. Climaco, Capacitated single allocation hub location problem: a bi-criteria approach. *Comput. Oper. Res.* **35** (2008) 3671–3695.

[13] A.T. Ernst and M. Krishnamoorthy, Solution algorithms for the capacitated single allocation hub location problem. *Ann. Oper. Res.* **86** (1999) 141–159.

[14] A.T. Ernst, H. Hamacher, H. Jiang, M. Krishnamoorthy and G. Woeginger, Uncapacitated single and multiple allocation p-hub center problems. *Comput. Oper. Res.* **36** (2009) 2230–2241.

[15] M. Haouari, N. Maculan and M. Mrad, Enhanced compact models for the connected subgraph problem and for the shortest path problem in digraphs with negative cycles. *Comput. Oper. Res.* **40** (2013) 2485–2492.

[16] B. Haouari, S. Bhar Layeb and H.D. Sherali, Tight compact models and comparative analysis for the prize collecting Steiner tree problem. *Discrete Appl. Math.* **161** (2013) 618–632.

[17] J.Q. Hu and B. Leida, Traffic Grooming, Routing, and Wavelength Assignment in Optical WDM Mesh Networks. *Proceedings of the IEEE INFOCOM* **2004** (2004) 495–501; Available at: DOI: https://doi.org/10.1109/INFCOM.2004.1354521.

[18] B.Y. Kara and B. Tansel, On the single-assignment p-hub center problem. *Eur. J. Oper. Res.* **125** (2000) 648–655.

[19] F. Larumbe and B. Sanso, Optimal Location of Data Centers and Software Components in Cloud Computing Network Design. *Proceedings of the 12th IEEE/ACM International Symposium on Cluster*, Cloud and Grid Computing (CCGrid) (2012) 841–844.

[20] I. Rodríguez−Martín and J.J. Salazar−González, Solving a capacitated hub location problem. *Eur. J. Oper. Res.* **184** (2008) 468–479.

[21] S. Gelareh and S. Nickel, Hub location problems in transportation networks. *Trans. Res. Part E: Logistics and Trans. Rev.* **47** (2011) 1092–1111.

[22] H.D. Sherali, Y. Lee and T. Park, New modeling approaches for the design of local access transport area networks. *Eur. J. Oper. Res.* **127** (2000) 94–108.

[23] H.D. Sherali and W.P. Adams, A hierarchy of relaxations between the continuous and convex hull representations for zero-one programming problems. *SIAM J. Discrete Math.* **3** (1990) 411–30.

[24] H.D. Sherali and W.P. Adams, A hierarchy of relaxations and convex hull characteristics for mixed-integer zero-one programming problems. *Discrete Appl. Math.* **52** (1994) 83–106.

[25] H.D. Sherali, S.C. Sarin and P. Tsai, A class of lifted path and flow-based formulations for the asymmetric traveling salesman problem with and without precedence constraints. *Discrete Optimiz.* **3** (2006) 20–32.

[26] H. Yaman and G. Carello, Solving the hub location problem with modular link capacities. *Comput. Oper. Res.* **32** (2005) 3227–3245.

[27] He. Yan and Wu. Tao, Canrong Zhang, Zhe Liang, An improved MIP heuristic for the intermodal hub location problem. *Omega* **57** (2015) 203–211.