Discrete Optimization

Single PON network design with unconstrained splitting stages

Luís Gouveia [a,1], Maria João Lopes [b,c,*], Amaro de Sousa [d,2]

[a] CIO-DEIO, Faculdade de Ciências, Universidade de Lisboa, 1749-016 Lisboa, Portugal
[b] Instituto Universitário de Lisboa (ISCTE-IUL), Av. das Forças Armadas, 1649-026 Lisboa, Portugal
[c] CIO, Faculdade de Ciências, Universidade de Lisboa, 1749-016 Lisboa, Portugal
[d] Instituto de Telecomunicações, Universidade de Aveiro, 3810-193 Aveiro, Portugal

ABSTRACT

A Passive Optical Network (PON) is a network technology for deploying access networks based on passive optical components. In a single PON access network, the client terminals are connected to a Central Office through optical splitters and interconnecting fibers where each splitter splits in equal parts the input optical signal coming from the Central Office over its different output fibers. In this paper, we consider PON topology solutions where the splitting ratio and the number of splitting stages are not constrained to a given target design but, instead, are decided based on the cost of the solutions. We present different Integer Linear Programming formulations to model this problem and provide computational results showing that the optimal solutions can be computed for realistic problem instances. In addition, we describe how the formulations can be adapted for the traditional PON topology approaches and present computational results showing that significant cost gains are obtained with the unconstrained splitting stage approach.

© 2014 Elsevier B.V. All rights reserved.

1. Introduction

In access networks based on passive optical components, a Passive Optical Network (PON) connects an output port of an OLT (Optical Line Terminal), located in a Central Office, to a set of ONUs (Optical Network Units), one for each client (herein, designated as client terminals). Such connections (from the Central Office to the client terminals) are based on passive optical components, i.e., optical splitters and fibers. When the number of client terminals is larger than the capacity of a PON, multiple PONs are deployed, each one corresponding to one output port of the OLT.

In this paper, we address the single PON network design problem in the context of densely populated urban scenarios. In these scenarios, client terminals are concentrated on a small number of locations (i.e., buildings) which enables us to model the problem with fewer variables and constraints. In the general case, the length of the path from the Central Office to each client terminal, and the number of splitters on it, is constrained by the maximum optical power that can be sent by the Central Office and the power

loss suffered by the optical signal on fibers and splitters. In densely populated urban scenarios, though, distances between the Central Office and the client locations are short and such constraints do not need to be considered. Moreover, network infrastructures, like ducts or fiber cables, are usually available in these scenarios (or can be used from other utility entities) which enable us to model connection costs by a cost per fiber model.

In the single PON network design problem, we have to decide where to install splitters and how to interconnect all network elements (Central Office, splitters and client terminals) through optical fiber connections. Splitters are optical devices that split in equal parts the optical signal coming from the Central Office over their output ports in a power of 2; this is the splitting ratio of the splitter, which can be 1:2, 1:4, 1:8 and so on. There are costs associated with the different splitter types (defined by their splitting ratios) and splitter locations and there are costs associated with fiber connections which are higher for longer fibers. The objective is to determine the minimum cost PON configuration.

Current research dedicated to PON network design usually assumes a PON capacity of 64, i.e., each output port of the OLT hosted in the Central Office can connect a maximum of 64 client terminals to the network. Nevertheless, it is foreseen that in the near future the capacity of the PONs will evolve to 128 or 256 client terminals as pointed out in ITU Standard (2010).

Most of recent works (an overview of these works is presented in next section) address the multiple PON network design problem

* Corresponding author at: Instituto Universitário de Lisboa (ISCTE-IUL), Av. das Forças Armadas, 1649-026 Lisboa, Portugal. Tel.: +351 217 903 228.
    E-mail addresses: legouveia@fc.ul.pt (L. Gouveia), mjfl@iscte.pt (M.J. Lopes), asou@ua.pt (A. de Sousa).
  [1] Tel.: +351 217 500 409.
  [2] Tel.: +351 234 370 052.

where a set of clients are to be connected through more than one PON because the total number of clients is higher than the capacity of a single PON. This is a very complex optimization problem since besides the design of each PON, this problem also includes the clustering of client terminals to PONs. Due to its complexity, the multiple PON network design problem is tackled with approximate methods. Such solution techniques are of interest for quickly finding solutions in the evaluation process of many different network scenarios and for deciding which PONs should be implemented and which client terminals should be connected to each PON. Nevertheless, for the final network design, exact methods that can find optimal solutions for each PON are preferable.

In this work, we address the single PON network design case, i.e., we assume that the clustering of client terminals to PONs has been previously decided and each PON design must be individually determined. Unlike most of previous works, we consider a PON topology where the splitting ratio and the number of splitting stages are not constrained to any target design but, instead, are decided by the optimization task based on the minimum cost objective. Unlike the traditional approaches, where the single PON network design is easy for real size problem instances, this is a hard problem as it will be seen in the computational results.

We start by presenting a generic model for the single PON network design problem, which is non-linear. Then, we propose several ILP (Integer Linear Programming) models and test them by solving test instances with up to 256 client terminals (a PON capacity value which is foreseen in the near future). In the context of the computational results, we compare the efficiency of the different models and show that the most efficient modeling alternatives are able to solve to optimality all problem instances.

In addition, we describe how the proposed models can be adapted for the traditional PON topology approaches. We present computational results that compare the obtained optimal costs between the different approaches. These cost comparisons show that significant cost gains are obtained for PONs of larger capacity with the unconstrained splitting stage approach when compared with the traditional approaches.

The paper is organized as follows. Section 2 describes the optimization problem and overviews the recent works on PON network design. Section 3 presents the several modeling alternatives to the problem. Section 4 presents and discusses the problem instances and the computational results. Finally, Section 5 presents the conclusions and the issues for future research.

2. Problem description

Consider the example shown in Fig. 1 defining the location of the Central Office (where the OLT is hosted) and of the client terminals that must be connected in a total of 64 client terminals (numbers close to client locations indicate the number of client terminals). Fig. 1 also shows the potential locations for the splitters.

The simplest possible PON configuration considers a single splitting stage. In this case, the Central Office is connected to a splitter and this splitter is connected to the individual client terminals. Since the number of terminals is 64 in our example, the splitter must have a splitting ratio of 1:64 (as illustrated in Fig. 2). In the downstream, the optical signal sent by the Central Office is split in equal parts by the splitter among all its output ports. This solution minimizes the cost associated with splitters while penalizing the cost associated with fibers. Note that if the number of terminals is lower than 64, but higher than 32, a splitter with splitting ratio 1:64 is still required although some of its output ports are not in use.

Some recent works have considered single splitting stage solutions in the design of multiple PONs in the context of greenfield
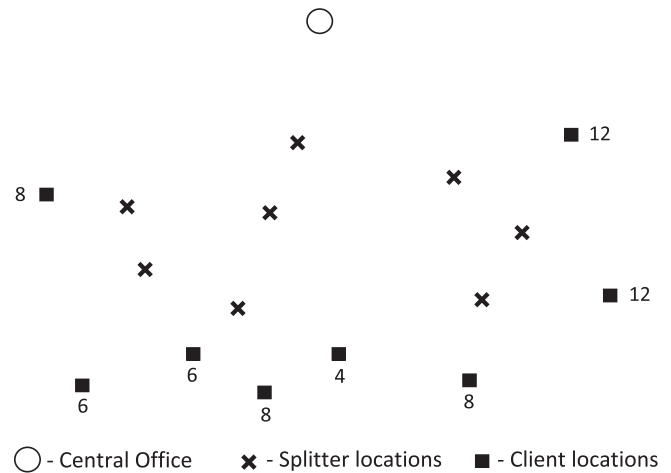

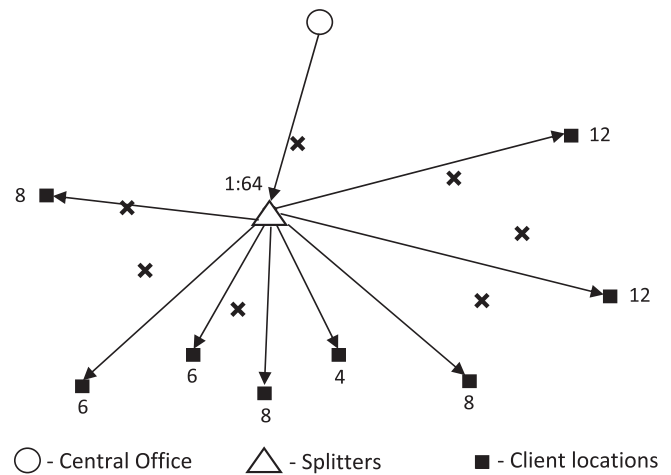**Fig. 1.** Central office, splitter and client locations.

Fig. 2. A single splitting stage solution.

scenarios, i.e., scenarios where no infrastructure exists and its deployment costs must be considered in the optimization task. Approximate solution methods are proposed both in Bley, Ljubic, and Maurer (2013) and in Li and Chen (2009) where this last work also considers maximum length constraints to the fiber lengths of each PON.

A more flexible solution is to consider two splitting stages, which often arises in practice. In this case, the Central Office is connected to a first stage splitter, then this splitter is connected to different second stage splitters and, finally, these splitters are connected to the individual client terminals. In Fig. 3, we illustrate a two splitting stage solution where the splitting ratio is 1:4 in the first stage and 1:16 in the second stage (note that other splitting ratios can also be adopted on each stage). As in the previous case, the optical signal sent by the Central Office must be split in equal parts by 64 before reaching each client terminal. Nevertheless, in this case, the split of the optical signal is distributed by the two splitting stages instead of being done on a single splitting stage (as in the previous case): it is first split by 4 in the first stage (the number of terminals served by each of these output connections is 64/4 = 16) and, then, each of these output signals is split by 16 in the second stage (the number of terminals served by each of these output connections is 16/16 = 1).

Note that, in this case, the client terminals in the same location can be connected to the network through different splitters (in
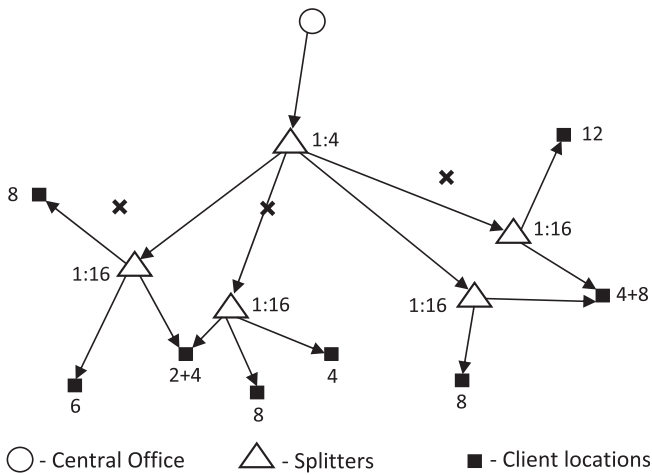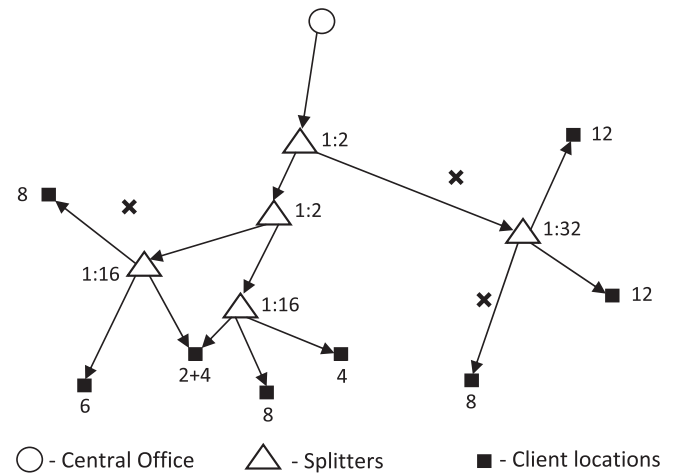
Fig. 3. A two splitting stage solution.

Fig. 4. An unconstrained splitting stage solution.

Fig. 3, this is the case in two different client locations). When compared with the single splitting case, this solution increases the cost associated with splitters (there are more splitters in the network) while decreasing the cost associated with fibers (most of the fibers are shorter since this solution enables the second stage splitters to be located closer to the client locations).

Chardy, Costa, Faye, and Trampont (2012) have recently addressed the design of multiple PONs using two splitting stage solutions for each PON where the splitting ratio of each stage is pre-defined. As in our case, they consider densely populated urban scenarios with existing network infrastructures. They propose an ILP model with additional strengthening constraints and graph reduction techniques that can obtain feasible solutions with small gaps for realistic problem instances. Kim, Lee, and Han (2011) consider the design of PON access networks where the objective is to minimize the network cost by proper location of the optical splitters. They formulate the problem as a multi-level capacitated facility location problem on an underlying tree topology with non-linear link costs. The single splitting and two splitting stage variants are addressed with separate formulations for each case.

In our work, we consider a more general approach, which we name unconstrained splitting stage approach. In this case, we let the number of splitting stages from the Central Office to the client terminals to be different for different terminals. As a consequence, in the same splitting stage the ratio may differ from one splitter to another. In Fig. 4, we illustrate a possible unconstrained splitting stage solution. In this case, the clients on the right-hand side of the figure are connected to the network through two splitting stages (1:2 in first stage and 1:32 in second stage) while the clients on the left-hand side of the figure are connected through three splitting stages (1:2 in first stage, 1:2 in second stage and 1:16 in third stage). Note, though, that the number of splitting stages and the splitting ratio of each stage is still such that the optical signal sent by the Central Office is split by 64 before reaching each client terminal. Finally, as in the previous case, we can also have client locations whose terminals are connected to the network through different splitters.

The main advantage of the unconstrained splitting stage approach is to enable more flexibility in the PON topology, aiming to reach cost savings in the network design solutions. In the computational results, we will show that for PONs of larger capacity, the cost savings can be significant when compared to the best of the previous approaches.

The work with a PON design variant similar to ours is the one proposed in Eira, Pedro, and Pires (2012), which deals with the design of multiple PONs with unconstrained splitting stages. That

work considers not only equipment and installation costs (CAPEX) but also network operational costs (OPEX). It proposes an ILP model based on a complex generalization of the hierarchical concentrator location problem. Nevertheless, the reported computational results show that instances with 100 client terminals and 12 possible splitter locations take almost 40 hours to be solved by CPLEX and, because of this, the authors also propose an approximate algorithm for the two splitting stage case. As it will be shown in the computational results, our most efficient ILP models are able to find optimal integer solutions for PONs with up to 256 client terminals and up to 42 possible splitter locations with much shorter running times. One important conclusion of the work presented in Eira et al. (2012) is that, due to the OPEX cost component, optimal solutions always consider a minimum number of PONs (i.e., OLT ports) and the number of client terminals assigned to each PON is closer to its maximum capacity for most of the PONs.

As a final remark, in this section we have considered the different PON topology approaches using an example (Fig. 1) where the Central Office is not a possible splitter location. In the general case, PON operators might consider that the first stage splitter can be hosted in the Central Office (directly connected to the output port of the OLT). This case can be easily considered in our models by including the Central Office in the set of the potential locations for the splitters.

3. Problem modeling

In this section, we present several formulations for the single PON network design problem. This problem is formulated on a network, where the Central Office, the intermediate locations and the client locations are represented by nodes of the network. The intermediate nodes represent the eligible locations for optical splitters and, thus, some of them may not be included in the PON. Two modeling alternatives can be chosen for the client terminals: either each terminal is represented by a node or the terminals sharing the same location are represented by a single node. Since in densely populated urban areas several client terminals share the same location (i.e., multiple clients in the same building), we have adopted the second alternative since it involves fewer nodes and the resulting models involve fewer variables and constraints.

Note that when the number of client terminals is exactly the capacity of the PON (which is a power of 2) and since the splitting ratios of splitters are a power of 2, the number of terminals served by each fiber connection is always a power of 2. Consider the previous example of Fig. 4. The connection from the Central Office to

the first stage splitter serves $64 = 2^6$ client terminals. The connections from the first stage splitter to the second stage splitters serve $32 = 2^5$ client terminals each (=64/2). Then, the connections from the 1:2 second stage splitter to the third stage splitters serve $16 = 2^4$ client terminals each (=32/2). Finally, the connections from all other splitters to clients serve $1 = 2^0$ client terminal each.

In our problem, the number of clients served by each connection is interpreted as a flow. The single PON network design problem can be modeled as a single-source/multi-destination network flow problem with flow constraints and additional constraints on the intermediate nodes. We consider that the number of client terminals is always equal to the capacity of the PON. If not, a dummy client location node with the required number of client terminals is included in the graph with arcs from all intermediate locations and associated costs equal to zero. With this modeling approach, the possible flow values of each connection are reduced to a minimum set composed by all powers of 2 (between 1 and the capacity of the PON). As it will be seen later on, this approach enables the definition of efficient ILP models.

In our problem, a PON has to satisfy the following five conditions: (i) it involves exactly one connection outgoing from the Central Office and serving *NT* client terminals (*NT* is the capacity of the PON); (ii) there is at most one connection incident on each intermediate node; (iii) each intermediate node hosts at most one splitter; (iv) the number of client terminals served by the output connections of a splitter is equal for all its output connections (a direct consequence of the split in equal parts of the incoming optical signal over the output connections) and (v) the number of output connections of each splitter is a power of 2. Before formulating the problem, we present the notation used in this paper.

NT	capacity of the PON, NT is a power of 2
0	node 0 represents the Central Office
S	set of intermediate nodes
T	set of client locations
N	set of all network nodes: $N = \{0\} \cup S \cup T$
A	set of arcs (i,j) representing pairs of nodes that can have an optical connection between them: node 0 has only outgoing arcs to nodes $s \in S$ and nodes $t \in T$ have only incoming arcs from nodes $s \in S$
M	set of all possible number of output connections of splitters: $M = \{2, 4, 8, 16, 32, 64, \dots, NT\}$
Q	set of all possible flow values on connections from $\{0\} \cup S$ to S: $Q = \{2, 4, 8, 16, 32, 64, \dots, NT\}$; from node 0 to an element of S the only flow value is NT and between elements of S the possible values are in $\{2, 4, 8, 16, 32, \dots, (NT/2)\}$
n_i	number of client terminals in location $i \in T$
c_{ij}	cost of a fiber connection associated with arc (i,j)
β_m	cost of a splitter of type 1:m, with $m \in M$
α_i	cost associated with the installation of a splitter in intermediate node $i \in S$

3.1. Generic formulation

We can obtain different formulations by specifying different sets of linear constraints for the problem described before. We start with a non-linear flow based model, similar to others used in the literature, with additional adequate constraints.

For this generic formulation, we consider the following sets of variables: binary variables $u_{ij} = 1, (i,j) \in A$, which indicate whether connection (i,j) is used or not; integer variables $f_{ij}, (i,j) \in A$, representing the flow value of connection (i,j), which is equal to the number of terminals served by this connection, binary variables $p_i^m = 1, i \in S, m \in M$, which indicate whether a splitter 1:m is installed on node i or not and the integer variables $q_i, i \in S$, representing the flow value on each arc outgoing from intermediate node i. The model is as follows:

$$\text{Min} \quad \sum_{i \in S}\sum_{m \in M}(\alpha_i + \beta_m)p_i^m + \sum_{(i,j) \in A: j \in S} c_{ij}u_{ij} + \sum_{(i,j) \in A: j \in T} c_{ij}f_{ij} \tag{1}$$

Subject to :

$$\sum_{(0,j) \in A: j \in S} u_{0j} = 1 \tag{2}$$

$$\sum_{(i,j) \in A: i \in S \cup \{0\}} u_{ij} \leqslant 1 \qquad j \in S \tag{3}$$

$$\sum_{m \in M} p_i^m \leqslant 1 \qquad i \in S \tag{4}$$

$$\sum_{(i,j) \in A: j \in S} u_{ij} + \sum_{(i,j) \in A: j \in T} f_{ij} = \sum_{m \in M} m p_i^m \qquad i \in S \tag{5}$$

$$\sum_{(j,i) \in A: j \in S \cup \{0\}} f_{ji} = \sum_{(i,j) \in A: j \in S \cup T} f_{ij} \qquad i \in S \tag{6}$$

$$\sum_{(i,j) \in A: i \in S} f_{ij} = n_j \qquad j \in T \tag{7}$$

$$f_{0j} = NT u_{0j} \qquad (0,j) \in A : j \in S \tag{8}$$

$$f_{ij} = q_i u_{ij} \qquad (i,j) \in A : i, j \in S \tag{9}$$

$$q_i \in \{2, 4, \dots, (NT/2)\} \qquad i \in S \tag{10}$$

$$u_{ij} \in \{0,1\} \qquad (i,j) \in A \tag{11}$$

$$f_{ij} \geqslant 0 \text{ and integer} \qquad (i,j) \in A \tag{12}$$

$$p_i^m \in \{0,1\} \qquad i \in S; m \in M \tag{13}$$

The objective is to minimize the total cost, which is the sum of the costs of intermediate nodes, optical splitters and fibre connections. Constraint (2) guarantees that the solution is connected to the Central Office by exactly one connection, whereas constraints (3) ensure that there is at most one connection ingoing to each intermediate node. Constraints (4) guarantee that each intermediate node hosts at most one splitter, whereas constraints (5) ensure that there are m connections outgoing from the intermediate node i if and only if this intermediate node hosts a splitter of type 1:m. These m outputs are either connections from i to other intermediate nodes $\left(\sum_{(i,j) \in A: j \in S} u_{ij}\right)$ or connections to m terminals $\left(\sum_{(i,j) \in A: j \in T} f_{ij}\right)$. The flow conservation is ensured by constraints (6). Constraints (7) guarantee that each terminal is included in the solution. Constraints (8) guarantee that the connection outgoing from the Central Office serves NT terminals, whereas constraints (9) together with (10) ensure that the flow outgoing from an intermediate node is equally divided in powers of 2. Furthermore, constraints (5) together with (6), (9) and (10) guarantee that the connections outgoing from an intermediate node are either connections between intermediate nodes or connections to terminal nodes. Finally, (11)–(13) are the domain constraints. Constraints (9) are the non-linear constraints and guarantee that the flow in connection (i,j) is equal to a power of 2.

3.2. Constructing a linear model

As the previous model shows, with the chosen set of variables it is far from obvious how to write linear constraints guaranteeing that the flow value on each arc leaving a splitter is equal to a power of 2 as well as to guarantee that the flow is equally divided by all arcs leaving a given splitter. More precisely, this happens for arcs (i,j) where $j \notin T$. We can overcome this difficulty by using discretized binary variables $z_{ij}^q = 1, (i,j) \in A : i \in S \cup \{0\}, j \in S, \quad q \in Q$, indicating whether the flow value on arc (i,j) is equal to q or not as in Gouveia (1995) (see also Gouveia & Saldanha da Gama (2006)). For connections between intermediate nodes and terminal

nodes, we consider the integer variables $z_{ij}, i \in S, j \in T$, representing the number of terminals located in j and connected to i, previously denoted by $f_{ij}, i \in S, j \in T$. The discretized variables are related with the original variables u_{ij} and f_{ij}, in the following way:

$$u_{0j} = z_{0j}^{NT} \qquad (0,j) \in A : j \in S \tag{14.1}$$

$$u_{ij} = \sum_{q \in Q} z_{ij}^q \qquad (i,j) \in A : i,j \in S \tag{14.2}$$

$$f_{0j} = NT z_{0j}^{NT} \qquad (0,j) \in A : j \in S \tag{15.1}$$

$$f_{ij} = \sum_{q \in Q} q z_{ij}^q \qquad (i,j) \in A : i,j \in S \tag{15.2}$$

$$f_{ij} = z_{ij} \qquad (i,j) \in A : j \in T \tag{15.3}$$

By either adding these equalities or doing the adequate substitutions on the previous model, we obtain a model for the problem which guarantees that the flow on each connection is a power of 2. However, we still need to guarantee that the output flow of a splitter is the same on each outgoing connection. To guarantee this condition, we consider new binary variables $w_i^q, i \in S, q \in Q \cup \{1\}$, indicating whether each flow value outgoing from a splitter installed on node i is equal to q or not, and the following linking constraints:

$$\sum_{(i,j) \in A : j \in S} z_{ij}^q \leqslant (NT/q) w_i^q \quad i \in S; q \in Q \tag{16.1}$$

$$\sum_{(i,j) \in A : j \in T} z_{ij} \leqslant NT w_i^1 \quad i \in S \tag{16.2}$$

$$\sum_{(j,i) \in A : j \in S \cup \{0\}} \sum_{q \in Q} z_{ji}^q = \sum_{q \in Q \cup \{1\}} w_i^q \quad i \in S \tag{17}$$

Constraints (16.1) and (16.2) match the flow value in variables w with the flow in variables z. Also, due to constraints (3), (14.1) and (14.2), the right-hand side of (17) is not greater than 1 and thus, the flow entering each intermediate node is split in only one way. Thus (16.1) and (16.2) and (17) ensure the flow values outgoing from an intermediate node i are equal. Furthermore, (17) guarantee that if there is a flow entering node $i, i \in S$, then exactly one of the variables $w_i^q, q \in Q \cup \{1\}$, will take the value 1. Moreover, if the flow is null then all variables $w_i^q, q \in Q \cup \{1\}$, will take the value 0. For completeness, we rewrite the model after these modifications, which we denote by P1A:

$$\text{Min} \quad \sum_{i \in S} \sum_{m \in M} (\alpha_i + \beta_m) p_i^m + \sum_{(i,j) \in A : j \in S} \sum_{q \in Q} c_{ij} z_{ij}^q + \sum_{(i,j) \in A : j \in T} c_{il} z_{il} \tag{18}$$

Subject to:

$$\sum_{(0,j) \in A : j \in S} z_{0j}^{NT} = 1 \tag{19}$$

$$\sum_{(i,j) \in A : i \in S \cup \{0\}} \sum_{q \in Q} z_{ij}^q \leqslant 1 \quad j \in S \tag{20}$$

$$\sum_{m \in M} p_i^m \leqslant 1 \quad i \in S \tag{4}$$

$$\sum_{(i,j) \in A : j \in S} \sum_{q \in Q} z_{ij}^q + \sum_{(i,j) \in A : j \in T} z_{ij} = \sum_{m \in M} m p_i^m \quad i \in S \tag{21}$$

$$\sum_{(j,i) \in A : j \in S \cup \{0\}} \sum_{q \in Q} q z_{ji}^q = \sum_{(i,j) \in A : j \in S} \sum_{q \in Q} q z_{ij}^q + \sum_{(i,j) \in A : j \in T} z_{ij} \quad i \in S \tag{22}$$

$$\sum_{(i,j) \in A : i \in S} z_{ij} = n_j \quad j \in T \tag{23}$$

$$\sum_{(i,j) \in A : j \in S} z_{ij}^q \leqslant (NT/q) w_i^q \quad i \in S; q \in Q \tag{16.1}$$

$$\sum_{(i,j) \in A : j \in T} z_{ij} \leqslant NT w_i^1 \quad i \in S \tag{16.2}$$

$$\sum_{(j,i) \in A : j \in S \cup \{0\}} \sum_{q \in Q} z_{ji}^q = \sum_{q \in Q \cup \{1\}} w_i^q \quad i \in S \tag{17}$$

$$z_{ij}^q \in \{0, 1\} \quad (i,j) \in A : j \in S; q \in Q \tag{24}$$

$$z_{ij} \geqslant 0 \text{ and integer} \quad (i,j) \in A : j \in T \tag{25}$$

$$p_i^m \in \{0, 1\} \quad i \in S; m \in M \tag{13}$$

$$w_i^q \in \{0, 1\} \quad i \in S; q \in Q \cup \{1\} \tag{26}$$

The objective function (18) and constraints (19)–(23) are taken from (1)–(3), (5)–(12) by using the relations (14.1), (14.2), (15.1)–(15.3) to remove the variables u_{ij} and f_{ij}. Constraints (8) have become redundant and are removed from the model, whereas (9) and (10) are guaranteed by the definition of the new flow variables together with (16.1), (16.2), (17) and (20), and are also removed. Finally, (13), (24)–(26) are the domain constraints.

In the next sections, we propose several model enhancements. In Section 3.3 we derive valid inequalities from a subproblem defined by (16.1), (24) and (26). Then, in Sections 3.4 and 3.5, the enhancements are based on disaggregation of variables and constraints. In Section 3.4, the variables defined for the optical splitters are disaggregated by adding information on the value of the outgoing flow; the constraints concerning the splitter output connections are disaggregated accordingly. In Section 3.5, valid equalities relating the variables w to the new variables introduced in the previous subsection are included in the model; these equalities motivate the disaggregation of another set of constraints.

3.3. Model enhancements (Part 1)

One typical way of improving the linear programming (LP) relaxation of a model is to find substructures in the model for which valid inequalities are known. In the case of model P1A, we consider the subproblem defined by constraints (16.1), (24) and (26). The LP relaxation of such a model can be easily improved (see, e.g., Gouveia, Moura, & Sousa (2011)) by adding the constraints:

$$z_{ij}^q \leqslant w_i^q \quad (i,j) \in A : i,j \in S; q \in Q \tag{27.1}$$

Similarly, the LP relaxation of the system define by constraints (16.2), (25) and (26) for $q = 1$, can be improved by adding constraints

$$z_{ij} \leqslant n_j w_i^1 \quad (i,j) \in A : j \in T \tag{27.2}$$

Note that constraints (27.2) can be seen as a disaggregation of constraints (16.2) because we obtain (16.2) adding (27.2) for all $j \in T$. Since the inclusion of constraints (27.1), instead of constraints (16.1), also permits the definition of a valid model with the remaining constraints, we denote by P1B the model P1A with (27.1) replacing (16.1), and with (27.2) replacing (16.2).

We denote by P1AB the model P1A with constraints (27.1) and (27.2). As mentioned before, constraints (27.2) are a disaggregation of constraints (16.2). Thus, constraints (27.2) dominate (16.2), which can be removed from P1AB. However, some preliminary computational experiments showed that the inclusion of (16.2), besides (27.2), reduces the CPU time for determining the optimal integer solutions. Naturally, P1AB can be considered as either P1A augmented with (27.1) and (27.2) or P1B augmented with (16.1) and (16.2).

The following result summarizes the relation between the LP relaxation of the models presented so far (we denote by P_L the LP relaxation of model P and by v(P) the optimal value of model P).

Result 1 $- v(P1A_L) \leqslant v(P1AB_L); v(P1B_L) \leqslant v(P1AB_L)$

Note that the above inequalities follow directly from the construction of the models. The computational results show that for all the instances tested the inequalities are strict. Also, it is easy to show that there is no relation between the values $v(P1A_L)$ and $v(P1B_L)$. However, for all the tested instances the relation $v(P1A_L) < v(P1B_L)$ holds.

The computational results presented in Section 4 will show that the models proposed until now are extremely time-consuming for the largest problem instances and, for several of the instances none of the proposed models was able to determine the optimal integer solution in six hours of runtime. The computational results will also show that the Linear Programming bounds are weak. In general, P1B generates better LP bounds than P1A and P1AB generates only slightly better LP bounds than P1B.

3.4. Model enhancements (Part 2)

In order to derive formulations with a stronger linear programming relaxation, we follow a strategy used for strengthening node degree variables as in Duhamel, Gouveia, Moura, and Souza (2012) and Gouveia et al. (2011) (see also Gouveia & Moura (2012)) and disaggregate the variables p_i^m. Consider the new variables $p_i^{mq} = 1, i \in S, m \in M, q \in Q \cup \{1\}$, which indicate whether a splitter $1:m$ with output flow value equal to q is installed on node i or not. Clearly, the pairs (m, q) that are feasible are such that the value mq does not exceed NT. The former variables defined for optical splitters and the new ones are related in the following way:

$$p_i^m = \sum_{q \in Q \cup \{1\}} p_i^{mq} \quad i \in S; m \in M \tag{28}$$

Using relations (28), we can replace each variable p_i^m, in model P1A, by the right-hand side of (28). The new model has the same LP relaxation value as the original model P1A since we have only performed variable substitution and for the moment nothing has been gained from adding more variables. However, the main idea of this disaggregation is to observe that the output flow values of an optical splitter must be equal to the flow value on connections outgoing from this optical splitter (see constraints (30.1) and (30.2) below). Thus, the constraints concerning the optical splitter output, rewritten using relations (28),

$$\sum_{(i,j) \in A: j \in S} \sum_{q \in Q} z_{ij}^q + \sum_{(i,j) \in A: j \in T} z_{ij} = \sum_{m \in M} \sum_{q \in Q \cup \{1\}} mp_i^{mq} \quad i \in S \tag{29}$$

can be disaggregated into

$$\sum_{(i,j) \in A: j \in S} z_{ij}^q = \sum_{m \in M} mp_i^{mq} \quad i \in S; q \in Q \tag{30.1}$$

and

$$\sum_{(i,j) \in A: j \in T} z_{ij} = \sum_{m \in M} mp_i^{m1} \quad i \in S \tag{30.2}$$

We can simplify the resulting model by using the following observations. First, note that, as pointed out before when motivating (30.1) and (30.2), we obtain (29) after adding (30.1) for all q and then adding the result to (30.2). Thus, constraints (29) can be removed from the model. We denote by P2A the model obtained by using relations (28) to replace each variable p_i^m, and adding constraints (30.1) and (30.2) and removing constraints (29). For completeness, we write next the resulting P2A after these modifications:

$$\text{Min} \sum_{i \in S} \sum_{m \in M} \sum_{q \in Q \cup \{1\}} (\alpha_i + \beta_m) p_i^{mq} + \sum_{(i,j) \in A: j \in S} \sum_{q \in Q} c_{ij} z_{ij}^q$$
$$+ \sum_{(i,j) \in A: j \in T} c_{il} z_{il} \tag{31}$$

Subject to:

$$\sum_{(0,j) \in A: j \in S} z_{0j}^{NT} = 1 \tag{19}$$

$$\sum_{(i,j) \in A: i \in S \cup \{0\}} \sum_{q \in Q} z_{ij}^q \leqslant 1 \quad j \in S \tag{20}$$

$$\sum_{m \in M} \sum_{q \in Q \cup \{1\}} p_i^{mq} \leqslant 1 \quad i \in S \tag{32}$$

$$\sum_{(i,j) \in A: j \in S} z_{ij}^q = \sum_{m \in M} mp_i^{mq} \quad i \in S; q \in Q \tag{30.1}$$

$$\sum_{(i,j) \in A: j \in T} z_{ij} = \sum_{m \in M} mp_i^{m1} \quad i \in S \tag{30.2}$$

$$\sum_{(j,i) \in A: j \in S \cup \{0\}} \sum_{q \in Q} qz_{ji}^q = \sum_{(i,j) \in A: j \in S} \sum_{q \in Q} qz_{ij}^q + \sum_{(i,j) \in A: j \in T} z_{ij} \quad i \in S \tag{22}$$

$$\sum_{(i,j) \in A: i \in S} z_{ij} = n_j \quad j \in T \tag{23}$$

$$\sum_{(i,j) \in A: j \in S} z_{ij}^q \leqslant (NT/q) w_i^q \quad i \in S; q \in Q \tag{16.1}$$

$$\sum_{(i,j) \in A: j \in T} z_{ij} \leqslant NT w_i^1 \quad i \in S \tag{16.2}$$

$$\sum_{(j,i) \in A: j \in S \cup \{0\}} \sum_{q \in Q} z_{ji}^q = \sum_{q \in Q \cup \{1\}} w_i^q \quad i \in S \tag{17}$$

$$z_{ij}^q \in \{0, 1\} \quad (i, j) \in A : j \in S; q \in Q \tag{24}$$

$$z_{ij} \geqslant 0 \text{ and integer} \quad (i, j) \in A : j \in T \tag{25}$$

$$p_i^{mq} \in \{0, 1\} \quad i \in S; m \in M; q \in Q \cup \{1\} \tag{33}$$

$$w_i^q \in \{0, 1\} \quad i \in S; q \in Q \cup \{1\} \tag{26}$$

The objective function (31) and constraints (32) are obtained by applying relations (28) to (18) and (4), respectively. In addition to (24)–(26), (33) are domain constraints.

Similarly to what has been described in Section 3.3, in the context of model P1A, we obtain a second model, denoted by P2B, by replacing, in P2A, constraints (16.1) and (16.2) by (27.1) and (27.2) and a third model, denoted by P2AB, by augmenting P2A with constraints (27.1) and (27.2). The following results (Result 2 and Result 3) update the relation between the LP relaxation of the models presented so far (as before, the inequalities follow from the construction of the models).

Result 2 $- v(P2A_L) \leqslant v(P2AB_L); v(P2B_L) \leqslant v(P2AB_L)$

The computational results show that for some instances the inequalities are strict. It is easy to show that there is no relation between the values $v(P2A_L)$ and $v(P2B_L)$. However, in our computations, the relation $v(P2A_L) < v(P2B_L)$ holds.

Result 3

$- v(P2A_L) \geqslant v(P1A_L); v(P2B_L) \geqslant v(P1B_L); v(P2AB_L) \geqslant v(P1AB_L)$

The computational results show that for some instances the second and third inequalities are strict. For the first inequality, all instances tested provided the same bound and, thus, it is still open whether equality holds or there are instances for which strict inequality holds.

The computational results presented in Section 4 show the advantages of the disaggregation described in this subsection, namely the reductions in CPU times required to obtain the optimal integer solutions. However, these CPU times are still large for the largest instances. Furthermore, for several of the instances tested none of the proposed models so far has been able to determine the optimal integer solution within six hours of runtime. This has motivated the disaggregation described next.

3.5. Model enhancements (Part 3)

In the previous sections, the decision variables originally associated with the optical splitters have been disaggregated. The new variables consider information not only about the optical splitter type but also about the optical splitter output flow. However, the models also involve the binary variables $w_i^q, i \in S, q \in Q \cup \{1\}$, indicating whether each flow value outgoing from a splitter installed on node i is equal to q. For the moment, no constraints linking the two sets of variables have been included. In this section we introduce constraints linking these two sets of variables and exploit them to derive new models.

Consider a node $i, i \in S$. If the flow value outgoing from a splitter installed on this node is equal to $q, q \in Q$, then we must have $w_i^q = 1$ and $p_i^{mq} = 1$, for exactly one m in M. On the other hand, if there is no optical splitter installed on this node whose output flows are equal to $q, q \in Q$, then $\sum_{m \in M} p_i^{mq} = 0$ and $w_i^q = 0$. Thus, the following equalities are valid:

$$w_i^q = \sum_{m \in M} p_i^{mq} \quad i \in S; q \in Q \cup \{1\} \tag{34}$$

These equalities are not satisfied by the solutions of P2A$_L$, P2B$_L$ and P2AB$_L$(this is easy to establish). The computational results reported in Section 4 will show that the inclusion of (34) in P2B improves significantly the lower bounds provided by P2B$_L$. Using relations (34), we can replace each variable w_i^q in the model P2A by the right-hand side of (34), and obtain the following model:

$$\text{Min} \quad \sum_{i \in S} \sum_{m \in M} \sum_{q \in Q \cup \{1\}} (\alpha_i + \beta_m) p_i^{mq} + \sum_{(i,j) \in A: j \in S} \sum_{q \in Q} c_{ij} z_{ij}^q$$
$$+ \sum_{(i,j) \in A: j \in T} c_{il} z_{il} \tag{31}$$

Subject to:

$$\sum_{(0,j) \in A: j \in S} z_{0j}^{NT} = 1 \tag{19}$$

$$\sum_{(i,j) \in A: i \in S \cup \{0\}} \sum_{q \in Q} z_{ij}^q \leqslant 1 \quad j \in S \tag{20}$$

$$\sum_{m \in M} \sum_{q \in Q \cup \{1\}} p_i^{mq} \leqslant 1 \quad i \in S \tag{32}$$

$$\sum_{(i,j) \in A: j \in S} z_{ij}^q = \sum_{m \in M} m p_i^{mq} \quad i \in S; q \in Q \tag{30.1}$$

$$\sum_{(i,j) \in A: j \in T} z_{ij} = \sum_{m \in M} m p_i^{m1} \quad i \in S \tag{30.2}$$

$$\sum_{(j,i) \in A: j \in S \cup \{0\}} \sum_{q \in Q} q z_{ji}^q = \sum_{(i,j) \in A: j \in S} \sum_{q \in Q} q z_{ij}^q + \sum_{(i,j) \in A: j \in T} z_{ij} \quad i \in S \tag{22}$$

$$\sum_{(i,j) \in A: i \in S} z_{ij} = n_j \quad j \in T \tag{23}$$

$$\sum_{(i,j) \in A: j \in S} z_{ij}^q \leqslant \sum_{m \in M} (NT/q) p_i^{mq} \quad i \in S; q \in Q \tag{35.1}$$

$$\sum_{(i,j) \in A: j \in T} z_{ij} \leqslant \sum_{m \in M} NT p_i^{m1} \quad i \in S \tag{35.2}$$

$$\sum_{(j,i) \in A: j \in S \cup \{0\}} \sum_{q \in Q} z_{ji}^q = \sum_{m \in M} \sum_{q \in Q \cup \{1\}} p_i^{mq} \quad i \in S \tag{36}$$

$$z_{ij}^q \in \{0,1\} \quad (i,j) \in A : j \in S; q \in Q \tag{24}$$

$$z_{ij} \geqslant 0 \text{ and integer} \quad (i,j) \in A : j \in T \tag{25}$$

$$p_i^{mq} \in \{0,1\} \quad i \in S; m \in M; q \in Q \cup \{1\} \tag{33}$$

Constraints (35.1), (35.2) and (36) are obtained by applying (34) to constraints (16.1), (16.2) and (17). Note that either constraints (20) or (32) can be removed from the model, since one of these sets becomes redundant, due to constraints (36). We will remove constraints (20) since they involve more variables than (32). Note also that (35.1) and (35.2) can be removed since they are dominated by constraints (30.1) and (30.2), because the coefficients of variables p_i^{mq} in the right hand side of constraints (35.1) and (35.2) are greater or equal than the coefficients of the same variables in the right-hand side of (30.1) and (30.2).

Constraints (34) also permit us to disaggregate constraints (36) into:

$$\sum_{(j,i) \in A: j \in S \cup \{0\}} z_{ji}^q = \sum_{(m,q') \in M \times (Q \cup \{1\}): mq' = q} p_i^{mq'} \quad i \in S; q \in Q \tag{37}$$

Furthermore, it is interesting to see that these new constraints (37) also guarantee the flow conservation, i.e., they imply the flow conservation constraints. To see this, note that multiplying each constraint (30.1) by q and then adding the results to (30.2) we obtain:

$$\sum_{j \in S: (i,j) \in A} \sum_{q \in Q} q z_{ij}^q + \sum_{j \in T: (i,j) \in A} z_{ij} = \sum_{q \in Q \cup \{1\}} \sum_{m \in M} mq p_i^{mq} \quad i \in S \tag{38}$$

Then, multiplying each constraint (37) by q, adding these equations and performing adequate substitutions we obtain:

$$\sum_{q \in Q} \sum_{(j,i) \in A: j \in S \cup \{0\}} q z_{ji}^q = \sum_{q \in Q \cup \{1\}} \sum_{m \in M} mq p_i^{mq} \quad i \in S \tag{39}$$

Thus, the left-hand sides of (38) and (39) are equal and we obtain:

$$\sum_{q \in Q} \sum_{(j,i) \in A: j \in S \cup \{0\}} q \, z_{ji}^q = \sum_{(i,j) \in A: j \in S} \sum_{q \in Q} q z_{ij}^q + \sum_{(i,j) \in A: j \in T:} z_{ij} \quad i \in S$$

which are the flow conservation constraints (22).

For completeness, we rewrite the model after these modifications:

$$\text{Min} \quad \sum_{i \in S} \sum_{m \in M} \sum_{q \in Q \cup \{1\}} (\alpha_i + \beta_m) p_i^{mq} + \sum_{(i,j) \in A: j \in S} \sum_{q \in Q} c_{ij} z_{ij}^q + \sum_{(i,j) \in A: j \in T} c_{il} z_{il}$$
$$\tag{31}$$

Subject to:

$$\sum_{(0,j) \in A: j \in S} z_{0j}^{NT} = 1 \tag{19}$$

$$\sum_{m \in M} \sum_{q \in Q \cup \{1\}} p_i^{mq} \leqslant 1 \quad i \in S \tag{32}$$

$$\sum_{(i,j)\in A:j\in S} z_{ij}^q = \sum_{m\in M} m p_i^{mq} \quad i\in S; q\in Q \tag{30.1}$$

$$\sum_{(i,j)\in A:j\in T} z_{ij} = \sum_{m\in M} m p_i^{m1} \quad i\in S \tag{30.2}$$

$$\sum_{(i,j)\in A:i\in S} z_{ij} = n_j \quad j\in T \tag{23}$$

$$\sum_{(j,i)\in A:j\in S\cup\{0\}} z_{ji}^q = \sum_{(m,q')\in M\times(Q\cup\{1\}):mq'=q} p_i^{mq'} \quad i\in S; q\in Q \tag{37}$$

$$z_{ij}^q \in \{0,1\} \quad (i,j)\in A : j\in S; q\in Q \tag{24}$$

$$z_{ij} \geqslant 0 \text{ and integer} \quad (i,j)\in A : j\in T \tag{25}$$

$$p_i^{mq} \in \{0,1\} \quad i\in S; m\in M; q\in Q\cup\{1\} \tag{33}$$

The previous model is denoted by P3A. In the previous sections, different sets of linking constraints have been considered to derive other models. Models P1B and P2B have been obtained by replacing in P1A and P2A constraints (16.1) and (16.2) by (27.1) and (27.2), respectively. With respect to P3A, constraints (16.1) and (16.2) have been rewritten as (35.1) and (35.2) and then shown to be redundant. Also, constraints (27.1) and (27.2) need to be rewritten since they involve the variables w_i^q. Using relations (34), we obtain:

$$z_{ij}^q \leqslant \sum_{m\in M} p_i^{mq} \quad (i,j)\in A : j\in S; q\in Q \tag{40.1}$$

$$z_{ij} \leqslant n_j \sum_{m\in M} p_i^{m1} \quad (i,j)\in A : j\in T \tag{40.2}$$

From an integer point of view, constraints (40.1) and (40.2) do not need to be involved in the previous model since constraints (30.1) and (30.2) already link the flow variables with the variables defined for optical splitters. Thus, we may view constraints (40.1) and (40.2) as valid inequalities to strengthen the linear programming relaxation of model P3A. We denote by P3AB the model P3A augmented with these inequalities to be consistent with the notation of the model P1 and P2.

The following results (Result 4, Result 5 and Result 6) update the relation between the LP relaxation of the models presented so far (as before, the inequalities follow from the construction of the models).

Result 4 – $v(\text{P3A}_L) \geqslant v(\text{P2A}_L); v(\text{P3AB}_L) \geqslant v(\text{P2AB}_L)$

The computational results show that for all instances the bounds are equal with respect to the bounds of the first inequality. However, we have built unrealistic instances (where we consider the cost of a splitter type much greater than the cost of the other splitter types) for which strict inequality holds.

Result 5 – $v(\text{P3AB}_L) \geqslant v(\text{P2B}_L)$

This result follows directly from the second inequalities of results 2 and 4.

Result 6 – $v(\text{P3AB}_L) \geqslant v(\text{P3A}_L)$

Here, the computational results show that for many instances the inequalities in results 5 and 6 are strict (and substantial improvements are obtained).

The computational results will show that the models proposed in this subsection reduce significantly the CPU times required to determine the optimal integer solution. The improvement on the LP bounds generated by the model P3AB is also significant, when compared with the ones obtained with the models P2B and P2AB. It is interesting to point out that the models P3 differ from the models P2 augmented with the equalities (34) mainly because of the disaggregation of constraints (36). This disaggregation seems to be the key to reduce the CPU times.

4. Computational results

In order to evaluate the ILP models proposed in Section 3, we have generated different problem scenarios with characteristics from urban areas. Each scenario is defined in an area with N_h horizontal streets and N_v vertical streets with a distance D_h between horizontal streets and a distance D_v between vertical streets. The set S of intermediate locations (i.e., candidate locations for splitters) is composed by all cross-sections, in a total of $N_h \times N_v$ locations. Both node 0 (the location of Central Office) and all client locations of set T are randomly placed on a point belonging to a street. Fig. 5 shows an example of an instance with $N_h = 4, N_v = 5$ and $|T| = 20$.

For each scenario, we set the number of client terminals n_j for each $j \in T$, by randomly generating an integer value between T_{min} and T_{max} with a uniform distribution. Table 1 presents the parameters used in the generation of 12 different scenarios, together with the total number of client terminals and the required PON capacity NT of the instance. The scenarios are grouped in three sets: S_1 with scenarios requiring a PON with a capacity of 64 terminals, S_2 with scenarios requiring a PON with a capacity of 128 terminals and S_3 with scenarios requiring a PON with a capacity of 256 terminals. The scenarios of the same set consider different client terminal concentration values, i.e., different average number of client terminals per client location.

Note that the cost of an optimal solution is a trade-off between fiber costs (given by c_{ij}) and splitter associated costs (given by α_i and β_m). In general, the relation between these two cost components depends on each particular case (for example, ownership or third party rental of fiber cables and ducts, fiber termination labor costs that can vary between different countries, etc.). If fiber costs are dominant, optimal PON solutions include more splitters in order to minimize the total fiber length. On the other hand, if splitter associated costs are dominant, optimal PON solutions include fewer splitters in order to minimize the splitter costs.

In order to study the efficiency of the proposed models in both cases, we have defined two test instances for each previously described scenario and indicate them as A and B. In both cases, we have considered the fiber costs given by $c_{ij} = c_f + c_u \times d_{ij}$, where $c_f = 10$ is a fixed cost due to fiber termination, $c_u = 1$ is a cost per unit of fiber length and d_{ij} is the Manhattan distance from node i to node j. We distinguish the instances A and B in the splitter associated costs. In test instances A, we consider a splitter cost as a
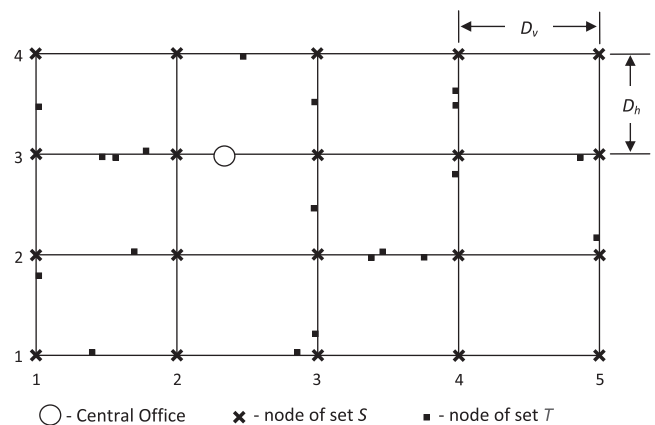

Fig. 5. Illustration of a problem scenario.

Table 1
Parameters of all problem scenarios.

Set	Scenarios	N_h	N_v	D_h	D_v	$|T|$	T_{min}	T_{max}	$\sum n_j$	NT
S_1	1	4	4	120	120	10	4	8	60	64
	2	4	4	120	120	15	2	6	63	64
	3	4	5	120	100	20	2	4	52	64
	4	4	5	120	100	30	1	3	59	64
S_2	5	5	5	100	100	10	8	16	119	128
	6	5	5	100	100	20	4	8	115	128
	7	5	6	100	90	30	2	6	121	128
	8	5	6	100	90	60	1	3	112	128
S_3	9	5	7	100	80	20	8	16	244	256
	10	5	7	100	80	40	4	8	233	256
	11	6	7	90	80	60	2	6	243	256
	12	6	7	90	80	120	1	3	248	256

Table 2
Gaps (in percentage) between the LP bounds and the optimal cost values.

Instance	P1A	P1B	P1AB	P2A	P2B	P2AB	P3A	P3AB
1A	52.43	30.38	29.52	52.43	28.74	28.72	52.43	13.12
2A	58.30	32.51	32.03	58.30	30.75	30.48	58.30	15.57
3A	57.82	27.51	26.69	57.82	25.44	25.44	57.82	7.84
4A	59.29	25.36	24.95	59.29	24.20	24.20	59.29	7.14
5A	33.16	20.40	20.36	33.16	20.07	20.07	33.16	9.49
6A	57.10	32.49	32.40	57.10	31.72	31.72	57.10	13.60
7A	59.15	30.83	29.88	59.15	29.27	29.26	59.15	11.01
8A	56.69	29.03	28.66	56.69	28.11	28.11	56.69	8.52
9A	45.74	28.21	27.78	45.74	27.48	27.48	45.74	13.02
10A	48.22	28.12	27.75	48.22	27.34	27.33	48.22	11.05
11A	53.26	30.46	29.84	53.26	29.43	29.42	53.26	11.80
12A	57.23	29.86	29.50	57.23	29.15	29.15	57.23	11.16
1B	60.65	44.56	43.91	60.65	42.73	42.73	60.65	8.07
2B	65.39	47.51	47.20	65.39	44.95	44.95	65.39	7.21
3B	62.98	40.42	39.84	62.98	37.89	37.89	62.98	6.34
4B	66.88	43.63	43.35	66.88	41.94	41.94	66.88	8.83
5B	45.55	36.30	36.25	45.55	35.80	35.80	45.55	7.42
6B	66.25	49.56	49.51	66.25	48.58	48.58	66.25	5.06
7B	67.14	47.16	46.48	67.14	45.52	45.52	67.14	5.06
8B	66.59	47.81	47.57	66.59	46.64	46.64	66.59	6.16
9B	63.19	52.40	52.14	63.19	51.76	51.76	63.19	16.85
10B	63.79	50.97	50.73	63.79	50.23	50.23	63.79	10.68
11B	65.78	50.44	50.03	65.78	49.46	49.46	65.78	10.38
12B	68.53	50.15	49.91	68.53	49.39	49.39	68.53	9.30
Average	58.4	37.8	37.3	58.4	36.5	36.5	58.4	9.8

sum of a fixed component $\alpha_i = 100$ for all nodes $i \in S$ and a type dependent cost as follows: $\beta_2 = 10, \beta_4 = 15, \beta_8 = 22, \beta_{16} = 34, \beta_{32} = 50, \beta_{64} = 76, \beta_{128} = 114$ and $\beta_{256} = 170$. In test instances B, we multiply the previous values by 5, i.e., we consider a fixed component $\alpha_i = 500$ for all nodes $i \in S$ and a type dependent cost as follows: $\beta_2 = 50, \beta_4 = 75, \beta_8 = 110, \beta_{16} = 170, \beta_{32} = 250, \beta_{64} = 380, \beta_{128} = 570$ and $\beta_{256} = 850$.

For all 24 test instances, we have solved all ILP models presented in Section 3 using CPLEX 12.3 from IBM ILOG. First, we have computed the LP lower bounds. Table 2 presents the obtained gaps (in percentage) between the LP lower bounds and the value of the optimal integer solutions. Table 2 also shows, in its last line, the average gaps exhibited by each model. The conclusions from these results are that, for these instances, models P1A, P2A and P3A have always the same LP bound and that the additional constraints considered in the other models always improve the gaps. Note that, in particular, the additional constraints (40.1) and (40.2) used in model P3AB have reduced significantly the LP bound of model P3A.

While solving the ILP models with CPLEX, we have defined the following branching priorities: for models P1, we have set a higher branching priority to variables p_i^{NT} and $z_{0i}^{NT}, i \in S$, while for the other models, we have set a higher branching priority to variables p_i^{NT1} and $z_{0i}^{NT}, i \in S$. In both cases, the priorities assigned to p_i^{NT} (in models P1) and p_i^{NT1} (in models P2 and P3) are higher than the ones assigned to $z_{0i}^{NT}, i \in S$. No priority was set to the remaining variables. The assigned branching priorities were motivated by the analysis of the LP relaxation solutions. These priorities aim to quickly remove (in the branch-and-cut tree) the characteristics of the LP solutions not included in feasible integer solutions. In LP solutions, many variables associated with the optical splitters of higher ratio 1:NT take non zero values (p_i^{NT} in models P1 and p_i^{NT1} in models P2 and P3), whereas in an integer solution only one of such variables might be non-zero. In addition, when one of these variables is fixed to one, the ILP is trivially solved. Therefore, the highest priority value was assigned to these variables. On the other hand, in LP solutions many variables associated with connections outgoing from the Central Office take non zero values, while in an integer solution only one of these connections is in the solution. Furthermore, after fixing the inclusion of one of these connections, the resulting problem is reduced to a subproblem with one less level. Thus, the second highest priority value was assigned to these variables.

Remember that, as described in Section 3, model P3AB is defined based on model P3A augmented by constraints (40.1) and (40.2). The number of these additional constraints is very large when compared with the number of constraints of model P3A. Since these constraints are not needed to define the problem, we have defined them in CPLEX as "user cuts". In this way, CPLEX adds them to its own sets of cuts and uses them only if they help in its

own optimization process. The runtimes of P3AB reported in the next tables were obtained in this way.

The computer used to obtain the optimal solutions is an Intel (R) Core (TM) i3 550 3.2 gigahertz with 4 gigabyte. We have defined a runtime limit of six hours for CPLEX to obtain the optimal integer solutions. Tables 3–5 present the results for the instances of set S_1, S_2 and S_3, respectively. Besides the cost of the optimal integer solution of each test instance (column 'OPT'), these tables present for each model the runtime (in seconds) for obtaining the optimal integer solution. When a percentage value is presented, it means that CPLEX did not solve to optimality the model within the six hours of runtime limit and the percentage value is the gap between the obtained upper and lower bounds at the end of the execution.

The results of Table 3 indicate that the problem instances of set S_1 are easy to solve with all proposed models. All instances have been solved to optimality within at most a few minutes. Note that these instances are easy also because the instance sizes (in number of elements of sets S and T) are small. Nevertheless, models P3A and P3AB already exhibit a better performance than the others since they can solve all problem instances within at most 2 seconds. Table 4, shown next, presents the results obtained for the instances of set S_2.

These results indicate that some of these instances are harder to solve. Moreover, the differences between the results obtained by the different models become more significant: the P1 models do not solve to optimality some of these instances; the P2 models

Table 3
Optimal cost values and runtime (in seconds) for S_1.

Instance	P1A	P1B	P1AB	P2A	P2B	P2AB	P3A	P3AB	OPT
1A	14	35	21	7	5	9	1	1	6698
2A	8	13	9	2	2	2	1	1	6204
3A	32	96	56	15	24	19	1	1	6313
4A	25	67	31	17	23	10	1	1	6612
1B	14	23	18	1	1	1	1	1	9886
2B	6	11	3	1	1	1	1	1	9509
3B	11	31	25	2	5	2	1	1	9095
4B	44	189	81	5	8	9	2	2	10,254

Table 4
Optimal cost values and runtime (in seconds) for S_2.

Instance	P1A	P1B	P1AB	P2A	P2B	P2AB	P3A	P3AB	OPT
5A	10	23	10	2	6	7	1	1	7205
6A	222	938	1599	56	89	64	2	2	9277
7A	13,550	3.07%	3.79%	1404	4117	3029	28	26	10,994
8A	12,048	3.54%	3.10%	543	1660	1797	23	23	10,192
5B	4	13	8	1	2	1	1	1	10,417
6B	91	280	121	3	7	5	1	1	14,327
7B	988	1827	1528	10	27	23	2	2	16,273
8B	1103	6509	3070	17	168	186	4	2	15,772

Table 5
Optimal cost values and runtime (in seconds) for S_3.

Instance	P1A	P1B (%)	P1AB	P2A	P2B	P2AB	P3A	P3AB	OPT
9A	11,942	0.74	3735	200	1663	698	6	6	14,142
10A	3.32%	5.05	4.67%	6239	12,626	7151	24	30	16,011
11A	7.78%	9.29	8.35%	3.49%	3.99%	3.91%	899	404	18,478
12A	8.01%	10.00	8.97%	4.27%	4.79%	4.99%	5892	3978	19,820
9B	3.99%	8.44	5.57%	504	2480	2367	25	21	23,780
10B	5.70%	7.87	6.53%	1093	7541	5087	38	38	25,880
11B	7.48%	8.80	7.76%	6225	0.81%	1.26%	173	199	28,398
12B	7.16%	8.06	7.57%	21,277	1.80%	1.42%	316	306	30,367

Table 6
Runtime (in seconds) for S_3.

Instance	P3A*	P3AB*	P3A	P3AB
9A	6	32	6	6
10A	36	295	24	30
11A	3332	9622	899	404
12A	16,006	3.15%	5892	3978
9B	26	184	25	21
10B	67	301	38	38
11B	262	1827	173	199
12B	760	7265	316	306

Table 7
Single, two and unconstrained splitting stage optimal costs of S_1.

Instance	1:64	1:2 + 1:32	1:4 + 1:16	1:8 + 1:8	U	Gain (%)
1A	13,655	9769	7282	6867	6698	2.5
2A	12,326	9284	6905	6204	6204	0.0
3A	12,666	8868	6491	6366	6313	0.8
4A	11,897	9341	7650	6835	6612	3.3
1B	14,395	11,409	9886	11,259	9886	0.0
2B	13,030	10,924	9509	10,596	9509	0.0
3B	13,370	10,508	9095	10,758	9095	0.0
4B	12,601	10,981	10,254	11,227	10,254	0.0

Table 8
Single, two and unconstrained splitting stage optimal costs of S_2.

Instance	1:128	1:2 + 1:64	1:4 + 1:32	1:8 + 1:16	U	Gain (%)
5A	20,410	14,000	7557	9186	7205	4.7
6A	23,569	19,483	13,410	9551	9277	2.9
7A	28,523	19,611	14,880	11,497	10,994	4.4
8A	26,726	19,406	13,715	10,996	10,192	7.3
5B	21,266	15,848	10,471	13,962	10,471	0.0
6B	24,425	21,331	16,270	14,327	14,327	0.0
7B	29,379	21,459	17,740	16,273	16,273	0.0
8B	27,582	21,254	16,575	15,772	15,772	0.0

can still find all optimal solutions within the maximum running time and the P3 models solve all instances with small running times (at most 28 seconds with P3A and 26 seconds with P3AB). Table 5, shown next, presents the results obtained for the instances of set S_3.

The results of Table 5 show that only models P3A and P3AB have solved all problem instances to optimality, some of them with short computing times. All other models have performed quite worse than these two models since they could not solve to optimality many instances. In fact, we consider that P3AB is better than P3A because it was able to solve the hardest problems in shorter runtimes (namely, instances 12A and 11A). Nevertheless, for the other easier instances, both models (P3A and P3AB) seem to be equivalent.

A global analysis of the results shown in Tables 3–5 shows two additional relevant conclusions. First, the required runtimes to solve the test instances is higher for instances A than for instances B. Note that instances A consider lower splitter associated cost, which means that the PON design problem is harder when the fiber associated costs are dominant. Second, the test instances with lower client terminal concentration values (i.e., the same number of client terminals distributed over a larger number of locations) are harder for all models. This observation is not surprising since the models of these instances exhibit more variables and constraints.

In order to access the improvements obtained by the adopted branching priorities and the use of constraints (40.1) and (40.2) as CPLEX "user cuts", in the next Table 6 we present the running

times obtained by models P3A and P3AB without such settings (shown in columns 'P3A*' and 'P3AB*'). For comparison reasons, we repeat the running times of models P3A and P3AB already shown in the previous Table 5. We show these results only for set S_3, which is the set of the hardest problem instances where the differences are more significant. These results clearly show that without using the branching priorities (and 'user cuts', in case of P3AB), the running times become larger (in particular for the hardest problems) and one of the instances (instance 12A) could not be solved to optimality by model P3AB.

Note that the proposed models can be easily adapted to address the single PON design problem for the traditional PON topology approaches (see discussion of Section 2). Using model P3A, this is done in the following way. In the single splitting stage case, we only consider the variables associated with splitters of maximum

Table 9
Single, two and unconstrained splitting stage optimal costs of S_3.

Instance	1:256	1:2 + 1:128	1:4 + 1:64	1:8 + 1:32	1:16 + 1:16	U	Gain (%)
9A	63,145	42,535	29,824	20,047	16,677	14,142	15.2
10A	57,833	38,465	27,932	21,201	16,947	16,011	5.5
11A	64,478	43,766	29,745	23,484	20,478	18,478	9.8
12A	68,881	50,485	34,496	25,795	21,255	19,820	6.8
9B	64,225	44,687	33,100	25,335	25,789	23,780	6.1
10B	58,913	40,617	31,208	26,489	26,059	25,880	0.7
11B	65,558	45,918	33,021	28,772	29,590	28,398	1.3
12B	69,961	52,637	37,772	31,083	30,367	30,367	0.0

ratio $p_i^{NT,1}$, $i \in S$, or equivalently, we set the variables associated with splitters of all other ratios equal to zero. In addition, we do not consider the variables associated with fiber connections between intermediate nodes, i.e., we set them equal to zero. In the two splitting stage case, let $1:m_1$ be the type of optical splitters required in the first splitting stage. We only consider the variables associated with splitters $p_i^{m_1,(NT/m_1)}$ and $p_i^{(NT/m_1),1}$, $i \in S$, or equivalently, we set all other variables associated with splitters equal to zero. In addition, we only consider the variables associated with fiber connections between intermediate nodes whose flow value is equal to (NT/m_1), i.e., we set all other variables associated with fiber connections equal to zero.

Tables 7–9 show the optimal cost values obtained for the three sets of instances and for the different single splitting and two splitting stage configurations. These tables also include the (previously presented) optimal cost values obtained for the unconstrained splitting stage solutions (column 'U') and the last column ('Gain') shows the cost gains, in percentages, obtained by the unconstrained splitting stage solution when compared with the best among all other solutions.

Note that the optimal unconstrained splitting stage costs cannot be worse than any of the other costs because all other solutions are feasible solutions for our problem. In general, Tables 7 and 8 show that for PONs with a capacity of 64 and 128 terminals, relevant cost gains are already obtained for many instances A (i.e., with lower splitter associated costs) and Table 9 shows that for PONs of higher capacity (256 terminals), there are cost gains in almost all instances and the cost gains are quite significant for instances A. These results demonstrate the relevance of the problem addressed in this paper, i.e., the optimal design of unconstrained splitting stage PONs.

As a final remark, we note that the single splitting stage and the two splitting stage adapted formulations are much more compact when compared with the unconstrained splitting stage formulations and this has a strong impact in the runtime required to solve the problem instances. All instances from 1 to 10 were solved to optimality within 2 seconds of computational time and the hardest cases were instance 12B, with a splitting stage distribution of 1:8 + 1:32, that took 21 seconds and instance 11B, with a splitting stage distribution of 1:16 + 1:16, that took 11 seconds.

5. Conclusions

In this paper, we have addressed the single PON network design problem with unconstrained splitting stages, i.e., the splitting ratio and the number of splitting stages are not constrained to a given target design but, instead, are decided based on the cost of the network solution.

We have proposed several models for this problem, which are reformulations, based on discretization of variables and disaggregation of variables and constraints, of a generic single-source/multi-destination flow model with additional constraints on the intermediate nodes. The results showed the advantages of the reformulations in the sense that the reformulations introduced on the models always helped in reducing the solution runtimes. Moreover, with the last reformulations, we were able to solve to optimality all considered problem instances.

We have also described how model P3A can be adapted for the traditional PON topology approaches (both single and two splitting stage problems). Note that the same adaptation can be used in all other proposed models. We have presented computational results showing that for PONs of larger capacity significant cost gains can be obtained with the unconstrained splitting stage approach when compared with the traditional approaches.

For future work, a natural follow up is to address the network design of multiple PONs with unconstrained splitting stages and allowing different topologies for each PON. This problem is of particular interest in the network planning stages when PON equipment solutions for 128 and 256 client terminals are envisaged. The clustering of client terminals to PONs taking into account that each PON can have any topological configuration (i.e., not constrained to a traditional topology) is potentially more efficient than current solution techniques available from the literature, provided that techniques able to find solutions with small optimality gaps can be achieved.

References

Bley, A., Ljubic, I., & Maurer, O. (2013). Lagrangian decompositions for the two-level FTTx network design problem. *EURO Journal on Computational Optimization, 1*(3–4), 221–252.

Chardy, M., Costa, M.-C., Faye, A., & Trampont, M. (2012). Optimizing splitter and fiber location in a multilevel optical FTTH network. *European Journal of Operational Research, 222*(3), 430–440.

Duhamel, C., Gouveia, L., Moura, P., & Souza, M. (2012). Models and heuristics for the k-degree constrained minimum spanning tree problem with node-degree costs. *Networks, 60*(1), 1–18.

Eira, A., Pedro, J., & Pires, J. (2012). Optimized design of multistage passive optical networks. *Journal of Optical Communication Networks, 4*(5), 402–411.

Gouveia, L. (1995). A 2n constraint formulation for the capacitated minimal spanning tree problem. *Operations Research, 43*, 130–141.

Gouveia, L., & Moura, P. (2012). Enhancing discretized formulations: The knapsack reformulation and the star reformulation. *TOP, 20*, 52–74.

Gouveia, L., Moura, P., & Sousa, A. (2011). Prize collection Steiner trees with node degree dependent costs. *Computers & Operations Research, 38*, 234–245.

Gouveia, L., & Saldanha da Gama, F. (2006). On the capacitated concentrator location problem: A reformulation by discretization. *Computers & Operations Research, 33*, 1242–1258.

Kim, Y., Lee, Y., & Han, J. (2011). A splitter location–allocation problem in designing fiber optic access networks. *European Journal of Operational Research, 210*(2), 425–435.

Li, J., & Chen, G. (2009). Cost minimization planning for greenfield passive optical networks. *Journal of Optical Communications and Networking, 1*(1), 17–29.

ITU-T G.987.1 Standard. (2010). 10-Gigabit-capable passive optical networks (XG-PON): General requirements. International Telecommunication Union. January 13.