A Combinatorial Optimisation Approach to the Design of Dual Parented Long-Reach Passive Optical Networks

Hadrien Cambazard, Deepak Mehta, Barry O'Sullivan, and Luis Quesada
4C, University College Cork, Cork, Ireland
Marco Ruffini, David Payne, and Linda Doyle
CTVR, University of Dublin, Trinity College Dublin, Ireland

***Abstract*—We present an application focused on the design of resilient long-reach passive optical networks. We specifically consider dual parented networks whereby each customer must be connected to two metro sites via a local exchange sites. An important property of such a placement is resilience to single metro node failure. The objective of the application is to determine the optimal position of a set of metro-nodes such that the total optical fibre length is minimised. We prove that the decision variant of this problem is NP-Complete. We present three alternative combinatorial optimisation approaches to finding an optimal metro node placement using: a mixed integer linear programming formulation of the problem; a hybrid approach that uses clustering as a preprocessing step; and, finally, a local search approach. We consider a detailed case-study based on a network for Ireland. The hybrid approach scales well and finds solutions that are close to optimal, with a runtime that is two orders-of-magnitude better than the MIP model. The local search approach is consistently good on all benchmarks.**

I. Introduction

Long Reach Passive Optical Networks (LR-PONs) are gaining increasing interest as they provide a low cost and economically viable solution for fibre-to-the-home network architectures [1]. The LR-PON by-passes local exchanges and replaces the metro transmission systems that would previously exist between the old local exchange sites and the metro-node. A major fault occurrence would be a complete failure of the metro-node that terminates the LR-PON, which could affect tens of thousands of customers. The *dual homing* protection mechanism for LR-PON enables customers to be connected to two metro-nodes via local exchange site, so that whenever a single node fails all customers are still connected to a back-up [2]. The part of the LR-PON that we want to protect is between the metro node and the old local exchange site. An important property of such a network is resilience to a single metro node failure. Nevertheless, connecting an exchange site to an additional metro node incurs additional cost.

The objective of the application is to determine the optimal positions of a set of metro-nodes such that the total fibre length required for connecting each exchange site to two metro-nodes is minimized. In this paper we prove that this problem is NP-Complete. Our solution relies on the application of standard combinatorial optimisation techniques to the design of resilient LR-PON. We present three alternative approaches to solving the optimal metro node placement problem. We first present a mixed integer programming (MIP) formulation of the problem. We then introduce a hybrid approach that uses clustering as a preprocessing step to filter out a set of possible locations for metro node placements for each exchange site and then solve the resulting problem optimally. Furthermore, we also present a local search approach that extends the well known Partitioning Around Medoids (PAM) algorithm [3] to address the double coverage requirement of our problem. We consider a detailed case-study based on a network for Ireland. We show that the hybrid approach scales well and finds solutions that are close to optimal, with a runtime that is two orders of magnitude better than the MIP model. Also, the local search approach is consistently good on all benchmarks.

The rest of the paper is organised as follows: Section II describes the motivation and an informal description of the problem addressed in this paper. Section III presents a formal description of the problem and gives its complexity. Section IV presents a MIP formulation of the problem. Section V introduces a hybrid approach that uses clustering as a preprocessing step, and finally a local search approach is described in VI. Empirical comparison of different approaches is presented in section VII followed by conclusions and future work in Section VIII.

II. Problem Description

Over the past decade traffic has grown exponentially at an average annual rate above 75% prompted by a multitude of new on-line content sharing applications such as Facebook and YouTube. Although the forecast for traffic growth over the next 5 years is reduced, it still suggests an average annual compound rate of about 37% with Internet video application traffic growing at about 47%. As High Definition (HD) and 3D video will increasingly be delivered over the Internet, such forecasts do not seem to over-estimate the traffic scenario. Additionally, delivering high peak data rates becomes increasingly important for delivering satisfactory quality of experience, especially for real-time services. Fiber-To-The-Premises (FTTP) in particular Fiber-To-The-Home (FTTH) seems to be the only solution capable of providing scalable access bandwidth for the foreseeable future.

Passive Optical Networks (PONs) are widely recognized as an economically viable solution to deploy FTTP and FTTH, by virtue of the ability to share costly equipment and fibre among a number of customers. In particular, the Long-Reach PON (LR-PON) is gaining interest. LR-PON provides an economically viable solution as the number of active network nodes can be reduced by two orders-of-magnitude and all electronic data processing can be removed from the local exchange sites, thereby reducing both cost and energy consumption [1]. However, a major fault occurrence like a complete failure of a single metro-node that terminates the LR-PON could affect tens of thousands of customers. Therefore, protection against a metro node failure is of primary importance for the LR-PON-based architecture.

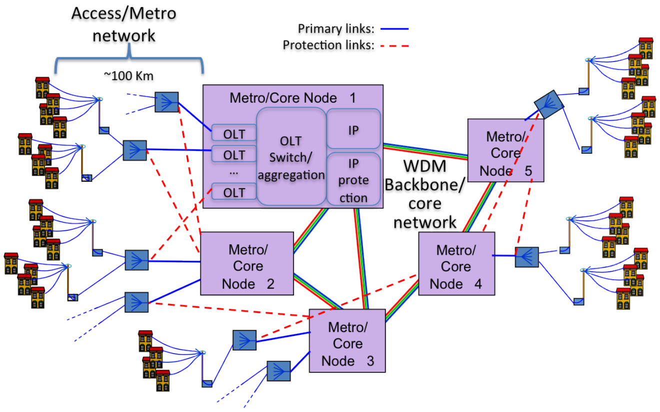


Figure 1. Example of a Long-Reach PON showing both access/metro and backbone/core networks

A basic and effective protection mechanism for LR-PON is to dual parent each system onto two metro/outer core nodes [4], [2]. This is similar to a simple protection solution for IP routers known as double or redundant protection [5]. Figure 1 shows an example of a PON network, together with its Wavelength-division Multiplexing (WDM) backbone interconnections. Each PON is dual parented, with the dashed lines representing the protection links. In this work we have considered protection links up to the first PON split (or local exchange site), leaving the "last-mile" unprotected. This is a common choice for residential customers, while protection can be extended to the user premises for business customers [4], [2]. For example, considering Figure 1, if metro node 1 fails, its PONs will be protected by metro node 2. We remark that node 2 needs to be over-provisioned with much larger IP capacity in order to protect the additional load [6]. Providing such a protection mechanism can significantly increase network overall cost because fibre deployment is a significant contribution to the total cost of the PON installation. Therefore, we focus on the problem of finding an optimal set of positions of k metro-nodes such that the cost of connecting optical fibres between metro-nodes and exchange sites is minimised. The domain of positions available for a metro node is the set of positions associated with the existing old local exchange sites.

III. Problem Formalization

In this section we formally describe the problem of LR-PON deployment for a real geography based on the data provided by the Irish incumbent operator. More precisely, first we present the definition and the complexity of the so called *Single Coverage Problem* where each exchange site is connected to a single metro node and then present the definition and complexity of the *Double Coverage Problem* where each exchange site is connected to two metro-nodes.

Definition 1 (Single Coverage Problem): An instance of the Single Coverage Problem (SCP) is defined by $\langle A, B, c, k, \phi \rangle$, where (A, B) is a complete bipartite graph with cost function c such that c_{ij} is the cost of allocating node $i \in A$ to node $j \in B$, k is an integer value such that $k \leq |B|$ and ϕ is some real value. An allocation from A to $M \subseteq B$ maps each node i of A to the cheapest node j of M such that $j = \arg\min_{j \in M} c_{ij}$ and $k = |M|$. The total cost of the allocation is the sum of the allocation cost of each node of A. The problem is to verify whether there exists a subset M of k nodes of B such that the total cost is less than or equal to ϕ.

Proposition 1: The Single Coverage Problem is NP-Complete.

Proof: A reduction from Hitting Set Problem [7] is obtained as follows: given a collection C of subsets of a finite set S and a positive integer $m \leq |S|$, the Hitting Set problem is to decide whether there is a subset $S' \subseteq S$ with $|S'| \leq m$ such that S' contains at least one element from each subset in C. The reduction to SCP, $\langle A, B, c, k, \phi \rangle$, goes as follows. We have a node in A for each set S_i in C and a node in B for each $j \in S$. The cost of all edges (i, j) is either 0 if j is in S_i or 1 otherwise. We set $\phi = 0$ and $k = m$. The constructed instance of SCP has a solution of cost 0 iff there exists a hitting set of size m for C. ■

Definition 2 (Double Coverage Problem): An instance of the Double Coverage Problem (DCP) is also defined by $\langle A, B, c, k, \phi \rangle$, where (A, B) is a complete bipartite graph with cost function c, where c_{ij} is the cost of allocating node $i \in A$ to node $j \in B$, k is an integer value such that $k \leq |B|$ and ϕ is some real value. An allocation from A to $M \subseteq B$ maps each node i of A to the cheapest node j_1 of M such that $j_1 = \arg\min_{j_1 \in M} c_{ij_1}$, and to the second cheapest node j_2 of M such that $j_2 = \arg\min_{j_2 \in M | j_2 \neq j_1} c_{ij_2}$. The total cost of the allocation is the sum of the allocation costs of each node of A to **two** nodes of B. The problem is to verify whether there exists a subset M of k nodes of B such that the total cost is less than or equal to ϕ.

Proposition 2: The Double Coverage Problem is NP-Complete.

Proof: We can reduce SCP to DCP by adding one extra node to B and setting the cost function accordingly. More precisely, let $B' = B \cup \{s\}$. Let c' be the cost function such that $c'_{ij} = c_{ij}$ if $i \in A$ and $j \in B$, otherwise $c'_{ij} = \beta$ such that $\beta < \min_{i \in A, j \in B} c_{ij}$. Solving the SCP instance $\langle A, B, c, k, \phi \rangle$ is equivalent to solving the DCP instance $\langle A, B', c', k+1, \phi + |A| \times \beta \rangle$. Notice that any solution of the SCP instance can be transformed into a DCP solution by setting s as the cheapest node for every node in A and making the SCP allocation equivalent to the allocation of the second cheapest node in the DCP instance. Similarly, any solution of the DCP instance can be transformed into a SCP solution by ignoring the cheapest node since the cost associated with the allocation of the cheapest nodes is bound to be equal to or greater than $|A| \times \beta$, thus making the allocation of the second cheapest node a valid solution of the SCP instance. ■

The reduction of SCP assumes that each allocation cost c_{ij} is independent of each other. That is, c_{ij} poses no restriction on any other $c_{i'j'}$. Apart from showing the complexity of the problems introduced, these reductions suggest alternative ways of solving the problems that are being reduced. For instance, it is an interesting insight to show how to solve the HITTING SET PROBLEM using a solver for DCP.

Let E be a set of exchange sites of Ireland whose locations are fixed. In Figure 2 all the points are locations of exchange sites in Ireland.[1] In this paper we focus on the double coverage problem where $A = B = E$. Let l_i be the load of the exchange site $i \in E$ which is equivalent to the number of customers that are connected to the exchange site i.

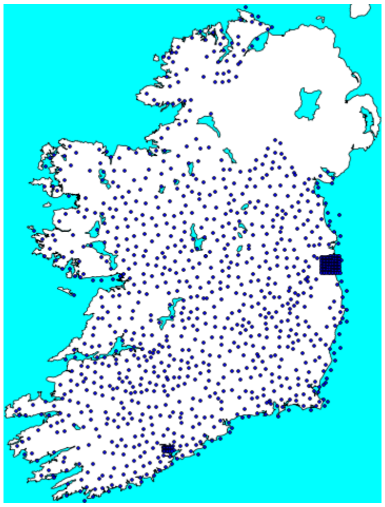


Figure 2. Exchange Sites in Ireland

Let k be the number of metro-nodes that are required to be placed in Ireland. A metro node can be placed at any position where an exchange site is located. Thus, the set of positions available for each metro node is the set of positions of all the exchange sites. Let d be a matrix where d_{ij} denotes Euclidean distance between the positions of exchange sites i and j. In order to account for the fact that the amount of fibre needed to connect two network points is usually larger than their Euclidean distance, because fibre paths generally follow the layout of the road network, a routing factor of 1.6 is applied. Let c_{ij} be the cost of connecting exchange site i to a metro node placed at the location of an exchange site $j \in E$, which is computed as follows:

$$c_{ij} = 1.6 \times d_{ij} \times \alpha_i \times l_i.$$

This cost model is based on the work of one of the authors while working at British Telecom [1]. Here α_i is constant and its value is dependent on the load of the exchange site i. The value of α_i decreases as the load increases since sharing of the fibre increases.

IV. MIP MODEL

The objective is to determine the positions of k metro-nodes such that each exchange site is connected to two metro-nodes and the sum of the costs of the connections between exchange sites and their respective metro-nodes is minimized. The closest metro node of an exchange site is called the primary metro-node while the second closest is called the secondary metro node.

Constants: Let E be a set of exchange sites whose locations are fixed. Let k be the number of metro-nodes whose positions are to be determined. Let c_{ij} be the cost of connecting an exchange site i to a metro node placed at the location of an exchange site $j \in E$.

Variables: $\forall (i,j) \in E \times E, x_{ij} \in \{0,1\}$ denotes whether exchange site i is connected to a metro node j. $\forall j \in E, y_j \in \{0,1\}$ denotes whether j is used as a metro node.

Constraints: Each exchange site $i \in E$ should be connected to two metro-nodes:

$$\forall i \in E : \sum_{j \in E} x_{ij} = 2 \tag{1}$$

Constraint (1) implicitly enforces that the primary and secondary metro-nodes of exchange site i should be different. For each exchange site $i \in E$ its primary and secondary metro-nodes can be inferred based on the costs of connecting i to the metro-nodes respectively.

If the metro node connected to an exchange site $i \in E$ is placed at the location of exchange site j then y_j is one:

$$\forall i, j \in E : y_j \geq x_{ij} \tag{2}$$

The number of constraints of type (2) is (n^2) which can grow quickly for large values of $n = |E|$, in which case

[1] Notice that some points are outside the boundary of Ireland. This is because the GPS co-ordinates that we have are not precise enough.

they can be replaced by the following weaker constraint:

$$\forall j \in E : |E| \times y_j \geq \sum_{i \leq n} x_{ij}$$

The number of used locations for metro-nodes should be equal to k:

$$\sum_{j \leq n} y_j = k$$

Objective.: The objective is to minimize the cost of the connection between local exchanges and their corresponding metro-nodes, i.e,

$$\min \sum c_{ij} \times x_{ij}$$

V. Cluster-Based Sampling

For the MIP model, as presented in the previous section, the set of positions of all exchange sites is considered as the domain of the metro node position for each exchange site. This may prohibit us from solving the problem optimally as the size of the set of the positions of the exchange sites increases. In order to overcome this scalability issue, we propose a heuristic approach as a preprocessing step for selecting a small subset of metro node positions for each exchange site and then use the MIP model to solve the problem optimally.

One simple approach to overcome this could be to limit the number of metro node positions of each exchange site based on their distances from the exchange site. More precisely, select k closest/cheapest metro node positions for each exchange site e. This heuristic approach is called *k-cheapest neighbours*. One of the drawbacks of this approach is that the resulting problem can be inconsistent especially when k is small. Therefore, one is required to find a value of k such that the problem is satisfiable. Another issue could be that an optimal solution of the resulting problem may not be of good quality despite the problem being satisfiable depending on the value of k. Obviously when $k = |E|$ we will always find the best solution but at the expense of more time. There is a trade-off between the value of k and the time required to find a solution of good quality.

We propose a new approach for computing a sample of positions where a metro node can be placed at a given exchange site. This heuristic approach is called *cluster-based sampling*. The pseudo-code is depicted in Algorithms 1 and 2. The general idea is to apply a variant of the k-means algorithm [8] for computing k clusters of exchange sites. Whenever a local minimum is reached within the algorithm, a best exchange-site position, based on some criterion, is selected from each cluster as a possible location of the metro node for all the exchange sites within that cluster. A sample of positions for each exchange site is computed by repeating this process a given number of times. The cardinality of this set is considerably smaller than the full set of positions of exchange sites. Since each exchange site should be connected to two metro-nodes the algorithm for weighted k-means clustering is adapted to ensure that each exchange site is in exactly two clusters.

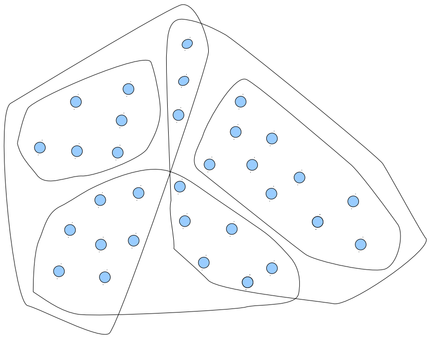


Figure 3. 5 overlapping clusters.

First we describe Algorithm 1 that computes k overlapping clusters. An example of overlapping clusters is presented in Figure 3 where the value of k is 5. Notice that each point is present in two clusters. The algorithm COMPUTEOVERLAPPINGCLUSTERS starts by selecting k points, $\langle m_1, \ldots, m_k \rangle$, randomly from a given set of sites E. These points represent initial k means of the overlapping clusters. Each m_i is associated with two attributes: $m_i.X$ denotes the X dimension and $m_i.Y$ denotes the Y dimension. Initially the cost is set to infinity. Each exchange site is assigned to two clusters: the one associated with the closest mean and another with the second closest mean. In the algorithm a cluster i is represented by $P_i \cup S_i$ such that m_i is the closest mean for each $p \in P_i$ and m_i is the second closest mean for each $p \in S_i$. We use $\texttt{dist}(p_i, p_j)$ to denote the Euclidean distance between the points p_i and p_j and $\texttt{W}[p_i]$ to denote the weight associated with a site p_i, which is equivalent to $\alpha_i \times l_i$ for our problem. The cost is evaluated by summing the weighted distances of all the points of the clusters with respect to their corresponding means. If the new cost is less than the current cost then the new means are calculated for all clusters. While the new cost is better than the previous cost the assignment of the exchange sites to two clusters and the update of the means is repeated. The algorithm returns the tuple $\langle P*, S* \rangle$. The complexity of each iteration within the while loop of Algorithm 1 is $\mathcal{O}(n\,k)$, where n is the number of sites and k is the required number of metro-nodes (or the number of clusters).

The input of SAMPLINGPOINTS (Algorithm 2) are $\texttt{nbruns}$, k and E. Here $\texttt{nbruns}$ denotes the number of times the overlapping clusters should be computed, k denotes the number of clusters and E denotes the set of exchange sites. $\texttt{Pos}(x_j)$ denotes a set of metro node positions of an exchange site x_j. Initially, $\texttt{Pos}(x_j)$ is an empty-set for each exchange site x_j. First, COMPUTEOVERLAPPINGCLUSTERS is invoked which returns a set of overlapping

Algorithm 1 COMPUTEOVERLAPPINGCLUSTERS(E, k)

```
cost $\leftarrow \infty$
select $k$ points, $m_1, m_2, \ldots, m_k$, randomly from $E$
loop $\leftarrow$ True
While loop do
  $\forall i \leq k :\ P_i \leftarrow \{x_j | \forall i* \leq k : \texttt{dist}(x_j, m_i) \leq \texttt{dist}(x_j, m_{i*})\}$
  $\forall i \leq k :\ S_i \leftarrow \{x_j | \exists i* \leq k : \texttt{dist}(x_j, m_{i*}) \leq \texttt{dist}(x_j, m_i) \wedge$
                    $\forall l \neq i* : \texttt{dist}(x_j, m_i) \leq \texttt{dist}(x_j, m_l)\}$
  newcost $\leftarrow \sum_{1 \leq i \leq k} \sum_{x_j \in P_i \cup S_i} W[x_j] \times \texttt{dist}(x_j, m_i)$
  If cost > newcost
    cost $\leftarrow$ newcost
    $P* \leftarrow P,\ S* \leftarrow S$
    For $i = 1, \ldots, k$ do
      $m_i.X \leftarrow \sum_{x_j \in P_i \cup S_i} W[x_j] \times x_j.X / \sum_{x_j \in P_i \cup S_i} W[x_j]$
      $m_i.Y \leftarrow \sum_{x_j \in P_i \cup S_i} W[x_j] \times x_j.Y / \sum_{x_j \in P_i \cup S_i} W[x_j]$
  Else
    loop $\leftarrow$ False
Return $\langle P*, S* \rangle$
```

Algorithm 2 SAMPLINGPOINTS(nbruns, E, k)

```
crun $\leftarrow$ 0
$\forall x_j \in E$, Pos$(x_j) \leftarrow \emptyset$
While crun < nbruns do
  crun $\leftarrow$ crun $+1$
  $\langle P, S \rangle \leftarrow$ computeOverlappingClusters$(E, k)$
  For $i = 1, \ldots, k$ do
    If $P_i \neq \emptyset$ then
      select $s \in P_i$ such that
      $\forall s' \in P_i\ \sum_{x_j \in P_i \cup S_i} W[x_j] \times \texttt{dist}(s, x_j) \leq$
                $\sum_{x_j \in P_i \cup S_i} W[x_j] \times \texttt{dist}(s', x_j)$
    Else if $S_i \neq \emptyset$
      select $s \in S_i$ such that
      $\forall s' \in S_i\ \sum_{x_j \in S_i} W[x_j] \times \texttt{dist}(s, x_j) \leq$
                $\sum_{x_j \in S_i} W[x_j] \times \texttt{dist}(s', x_j)$
    $\forall x_j \in P_i \cup S_i :$ Pos$(x_j) \leftarrow$ Pos$(x_j) \cup \{s\}$
Return Pos
```

clusters such that each exchange site is present in exactly two clusters. Recall that a cluster (of exchange sites) i is denoted by $P_i \cup S_i$. After that an element s is selected from each cluster as a possible metro node position for all the exchange sites within $P_i \cup S_i$. Also recall that $P_i \cup S_i$ means that the selected metro node s is the cheapest/closest for each $e \in P_i$ and it is second cheapest for each $e \in S_i$. Therefore, if $P_i \neq \emptyset$ then s is selected from P_i such that the sum of the weighted distances between s and all the exchange sites of the cluster i is minimum. Otherwise it is selected from S_i. This entire procedure is repeated nbruns times. Algorithm 1 can be seen as a variant of weighted k-means clustering algorithm. The main difference is that in the original algorithm clusters are pair-wise mutually exclusive but Algorithm 1 computes overlapping clusters as required by the problem.

VI. LOCAL SEARCH APPROACH

The double coverage problem is very close to facility location problems and the p-median (also called k-medoid) problem [3]. The k-medoid problem differs from the double coverage in two respects: each point has to be connected to two centers (metro-nodes), rather than one, and the cost is proportional to the Euclidian distance; this is often the case for the k-medoid problem but not necessarily. The standard algorithm for solving the k-medoid problem is the PAM algorithm which can be seen as a form of local search. In the following, we investigate a similar technique for our specific context.

Solution Representation: A solution is represented by a list of different points $M = \{m_1, \ldots, m_k\} \subseteq E$ with $|M| = k$ encoding the metro-nodes. Firstly, all the points covered by a metro node are stored in a table denoted $C(m_i)$. Secondly, the two metro-nodes allocated to each point x_j are stored as $s_1(x_j)$, $s_2(x_j)$. Thirdly, for each point $x_j \in E$, the list $M(x_j)$ of metro-nodes sorted by distance to x_j is maintained; $M(x_j)$ can be implemented as a priority queue (based on a heap) to get a logarithmic complexity for updating it *i.e* inserting and removing elements. Each point x_j appears in the two lists $C(s_1(x_j))$, $C(s_2(x_j))$ and the two metro-nodes $s_1(x_j)$, $s_2(x_j)$ are always the first two elements in $M(x_j)$ so that each point is always covered by its two closest metro-nodes. This invariant is maintained throughout the search. Finally, the total cost of all points connected to a given metro node m_i is stored in $cost(m_i)$ and objective function is equal to $\sum_{m_i \in M} cost(m_i)$.

Neighborhood: The neighborhood is defined by moving a metro node from one position to another. A node $m_i \in M$ is removed from M and a new node $x_j \in E - M$ is added to M. The only move considered is, therefore, to change the location of one of the existing metro-nodes and is denoted by a pair (m_i, x_j).

The variation of the objective function (ΔC) due to moving m_i to x_j can be computed in $O(n)$ and is summarized as follows :

1) DE-CONNECTION: The cost of de-connecting all nodes from m_i is directly given by $cost(m_i)$ so this step takes $O(1)$.
2) RE-CONNECTION: For all $x'_j \in C(m_i)$ (all points connected to m_i) the cost of re-connecting x'_j can be computed in constant time using the list $M(x'_j)$. The metro node that should cover x'_j is one of the first three elements in $M(x'_j)$ (because one element is the metro node m_i that is removed). The three elements of highest priority are necessarily in the first three levels of the tree, i.e, top seven nodes of the tree (the root of a subtree in a priority queue has a higher priority than its children). This step takes $O(|C(m_i)|)$.
3) UPDATE CONNECTIONS: For all $x'_j \in E - C(m_i)$ we must check whether x_j is now closer to x'_j than $s_1(x'_j)$ or $s_2(x'_j)$ and add the gain in cost if it is the case. This final step takes $O(n)$.

Computing the variation of the objective function is dominated by the last step whose cost is $O(n)$. The update of the data structures used in the solution representation is done when a move is performed, i.e. once per iteration, and is dominated by the update of $M(x_j)$, which takes $O(n\,log(k))$ when implementing $M(x_j)$ with a priority queue.

Search: The first improving move is always accepted and a simple tabu mechanism prevents a node which was a metro node in the last t iterations (where t is the length of the tabu list) to become a metro node again. The initialization of the search is done by picking k random points which always gives a feasible solution of the coverage problem. The exploration of the neighborhood made of $k(n-k)$ possible re-allocation of a metro node is performed from the last point where it was left so that all metro-nodes are regularly examined. If no improving move is found, a metro-node is chosen at random and relocated to a random position.

Overall, one iteration of the local search approach takes $O(n^2\,k)$ which is very costly and is a limit for the PAM algorithm. This approach is still applicable for our ranges of values of n and is usually known to give near optimal results for k-medoids problem.

VII. Empirical Results

In this section we investigate different approaches for determining locations of metro-nodes in Ireland. To be precise, we compared an exact approach based on a mixed integer linear programming formulation with a local search approach and two hybrid approaches where for each exchange site a set of metro-node positions are discarded based on a criterion. The exact approach is denoted by MIP, the local search approach is denoted by LS, the hybrid approaches based on k-cheapest neighbours and cluster-based sampling are denoted by KCN and CBS respectively.

Table I
Results for 19 metro-nodes.

$\|E\|$	optimal	CBS (GAP)	Time	
			MIP	CBS
100	470,439,821	0 %	0.25	0.71
200	475,779,040	0 %	2.29	1.59
300	476,876,335	0.000%	36.90	3.15
400	477,736,761	0.009%	52.76	4.40
500	476,930,454	0.014%	96.89	6.59
600	476,860,839	0.013%	168.47	8.49
700	477,825,864	0.012%	1,277.27	14.68
800	477,432,981	0.033%	498.29	17.43
900	477,608,042	0.019%	817.24	20.09
1000	477,730,261	0.029%	1,081.61	32.78
1100	477,789,473	0.038%	1,716.27	39.32

Table II
Results for 20 metro-nodes.

$\|E\|$	optimal	CBS (GAP)	Time	
			MIP	CBS
100	456,703,030	0 %	0.27	0.69
200	462,745,384	0 %	2.26	1.68
300	463,322,390	0.001%	14.84	2.88
400	464,669,197	0 %	70.46	4.57
500	464,018,395	0.001%	115.40	6.78
600	464,181,132	0.034%	226.34	11.00
700	464,696,666	0.006%	405.57	11.49
800	464,576,759	0.034%	661.01	19.25
900	464,918,687	0.039%	1,108.49	27.26
1000	464,968,787	0.028%	1,587.39	31.74
1100	465,066,168	0.034%	8,777.75	54.53

We used CPLEX (version 12.1 with 8 parallel threads) for solving all the integer linear programming formulation of the instances of double coverage problem. All the algorithms were implemented using Java as a programming language. In our experiments, we varied the number of metro-nodes between 18 and 24 for Ireland. The results are reported for 19, 20, 23 and 24 metro-nodes. The original problem had 1100 exchange sites. In order to do systematic experimentation, we generated 10 instances of smaller sizes. These instances are representative of the original instance since they were generated by applying k-means algorithm on the original instance by varying k [2] (or the number of required exchange sites) from 100 to 1000 in steps of 100 [3]. All the experiments were run on Linux 2.6.25 x64 on Dual Quad Core Xeon CPU with overall 11.76 GB of RAM and processor speed of 2.66GHz.

The results for MIP are presented in Tables I-IV. All the experiments for this approach were run to completion. The optimal values computed using this approach are shown under the column named "optimal". The results in terms of time (in seconds) are also reported. In terms of time this was the most expensive approach especially when the number of exchange sites are more than 500.

Table III
Results for 23 metro-nodes.

$\|E\|$	optimal	CBS (GAP)	Time	
			MIP	CBS
100	421,504,120	0 %	0.24	0.72
200	429,159,208	0.048%	7.63	2.19
300	429,880,291	0 %	19.82	2.85
400	430,115,650	0.005%	161.45	5.41
500	430,043,176	0.001%	350.34	9.18
600	429,866,927	0.033%	713.30	10.52
700	430,802,977	0.019%	1,761.50	18.17
800	430,755,591	0.011%	2,631.64	21.79
900	430,737,706	0.024%	3,858.39	30.84
1000	430,918,149	0.008%	7,537.79	39.18
1100	430,839,593	0.026%	9,706.40	36.61

In the tables the CBS (GAP) is the percentage of the optimal cost by which the best cost obtained by the CBS approach exceeds. This gap is always within a 0.04% gap, which is significantly low. The application of cluster-based sampling for discarding a set of metro node positions for each exchange site before the search starts can be an overhead. However, it pays off since the time required for search reduces significantly without sacrificing the quality of the solution. From Tables I-IV it is clear that the CBS

[2] The load of the mean is the agregation of the loads of the nodes inside the cluster, which explains the little variation on the optimal value.

[3] All the generated instances are available on http://4c.ucc.ie/~lquesada/DoubleCoverage/page/instances.htm.

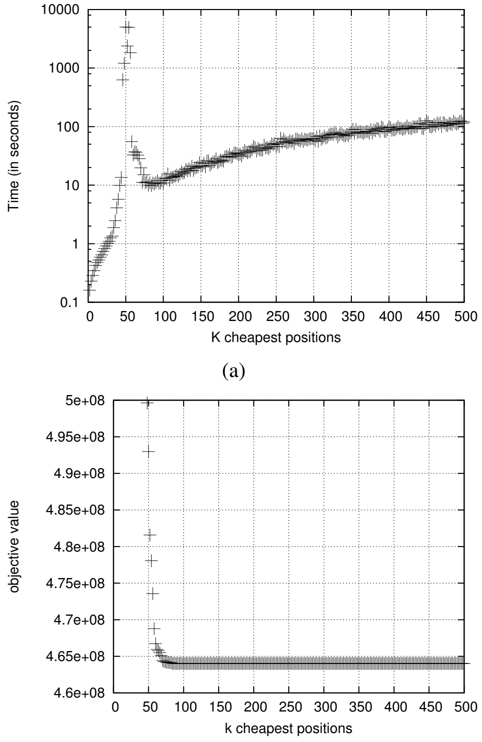
Table IV
RESULTS FOR 24 METRO-NODES.

$\|E\|$	optimal	CBS (GAP)	Time MIP	 CBS
100	411,560,864	0 %	0.34	0.71
200	419,088,008	0 %	9.04	2.15
300	420,069,722	0.005%	23.64	2.93
400	419,722,195	0 %	68.85	4.88
500	419,700,725	0 %	182.44	7.39
600	419,773,717	0.039%	293.45	9.48
700	420,102,946	0.008%	903.16	11.71
800	420,352,288	0.007%	1,532.55	16.61
900	420,235,833	0.011%	1,752.59	21.46
1000	420,317,577	0.009%	3,657.55	25.93
1100	420,347,707	0.025%	4,316.71	33.29

approach reduces the time significantly. In fact for harder instances it requires almost two orders-of-magnitude less time than that of the MIP approach.

Although the KCN approach may solve a problem instance quicker than the MIP approach, one issue is to determine the right value of k. A small k may result in making the problem inconsistent and a large k may result in spending more time. Also, despite having a satisfiable problem when k is set to a relatively lower value, it can still result in spending more time than that required for solving the original problem, when all the positions are considered for all exchange sites. This is illustrated in Figure 4 by plotting the results for solving an instance of double coverage problem where the number of exchange sites is 500 and the number of metro-nodes is 20. For both Figures 4(a) and 4(b) x-axis denotes the value of k, which is varied from 2 to 500 in steps of 2. The y-axis of Figure 4(a) is the time required to solve the instance and the y-axis of Figure 4(b) is the optimal value corresponding to k. Notice that when k is less than or equal to 46 the problem is always unsatisfiable. An interesting point to observe is that when k is between 48 and 56 the time required to solve can be up to 2 orders-of-magnitudes more than that required when k is 500. Also notice that when k is set to 84 an optimal solution is discovered and the time required to find an optimal solution is also the least. The results of KCN approach are not reported in Tables I-IV for two reasons. First, determining the right value of k is not always possible and additionally there is an overhead. Second, the other hybrid approach CBS almost always outperform KCN in terms of time without degrading the quality of the solution.

The advantage of the hybrid approach CBS over KCN is that if an original instance is satisfiable then a modified instance obtained by CBS is also satisfiable. This holds true even when the size of the domain of metro-node positions for each exchange site is 2, albeit the cost of the solution may increase. Another advantage is that it does not enforce any lower bound restriction on the domain size of the metro-node positions for any exchange site. An upper bound restriction is implicitly imposed by the parameter nbruns which is equal to the number of times Algorithm 1 is invoked for computing over-lapping clusters. The domain of the metro-node positions for different exchange sites could be different which is demonstrated in Figure 5 where the number of metro-nodes is 20 and the number of exchange sites is 500.



(a)

(b)

Figure 4. KCN approach for 20 metro-nodes and 500 exchange sites.

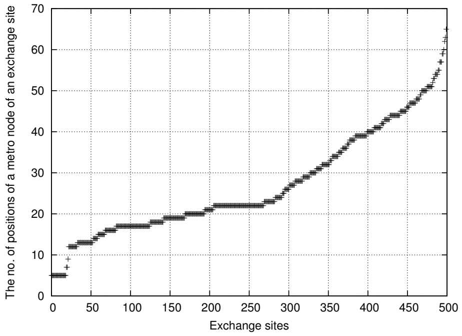


Figure 5. Domain size of positions of 20 metro-nodes for 500 exchange sites.

Finally, we discuss the results obtained by local search. The time out was 100 seconds for problem instances containing less than or equal to 500 exchange sites and $1,800$ seconds for problem instances containing more than 500 exchange sites. The stopping criterion was time but it may be possible to specify other terminating conditions. For each instance the algorithm was run 10 times with different seeds. Interestingly local search always managed to find an optimal solution within the specified time limit for the

Table V
AVERAGE AND MAXIMUM TIME FOR LS TO FIND THE OPTIMAL SOLUTION.

	20		23		24	
$\lvert E\rvert$	avg	max	avg	max	avg	max
100	0.04	0.16	0.07	0.26	0.21	1.11
200	0.22	1.28	0.28	0.98	1.22	2.88
300	0.52	0.96	0.88	3.65	1.03	2.47
400	2.25	9.08	2.91	11.27	2.06	5.24
500	4.91	14.74	3.54	9.35	4.04	9.42
600	4.57	16.50	5.46	14.66	8.32	19.71
700	2.11	7.06	12.36	26.75	9.67	23.70
800	14.55	52.56	13.26	62.88	27.61	90.33
900	41.51	271.70	22.95	64.34	22.51	48.00
1000	43.90	113.20	42.30	84.76	21.92	76.84
1100	141.89	401.54	57.21	190.15	60.23	127.10

instances whose results are reported in Table I-IV. Although the search was terminating after reaching the threshold, the optimal solution was found much earlier in most of the cases. Table V reports the average time and maximum time over 10 runs at which the optimal solution was discovered for 20, 23 and 24 metro-nodes. In terms of time, LS performs significantly better than the MIP approach. In order to compare the CBS approach with the LS approach the time required for the CBS approach was used as the time-limit for the LS approach and their gaps with respect to the optimal cost of the MIP approach are reported in Table VI. The results reported in Table VI are for solving the instances that contain 1100 exchange sites and the number of metro-nodes was varied from 19 to 30. The results suggest that CBS reduces the gap to the optimal value faster than LS as the number of metro-nodes increases.

Table VI
COST GAP FOR LS AND CBS

#nodes	LS	CBS
19	0.009	0.038
20	0.022	0.034
21	0.014	0.037
22	0.005	0.018
23	0.027	0.026
24	0.031	0.025
25	0.004	0.013
26	0.023	0.016
27	0.025	0.008
28	0.070	0.014
29	0.136	0.007
30	0.063	0.006

VIII. CONCLUSIONS AND FUTURE WORK

We have studied and solved the double coverage problem arising in long reach passive optical networks that are robust to single node failures. We showed that the double coverage problem is NP-Complete. In order to minimize the total length of optical fibres that connect metro-nodes and exchange sites we modeled the problem using mixed integer linear programming.

We proposed and studied a hybrid approach that performs cluster-based sampling as a preprocessing step in order to reduce the possiblities of metro node positions for exchange sites. We showed that the hybrid approach can significantly reduce the time required to solve the double coverage problem up to two orders-of-magnitude, especially when the size of the problem instance is large. Our study also shows that the best solutions obtained by using the hybrid approach CBS are almost optimal.

We also studied a local search approach that extends the well known Partitioning Around Medoids (PAM) algorithm. The extension basically allows us to express that each exchange site should be covered by two metro-nodes instead of just one. The study has shown that this approach almost always find the optimal solution of the instances related with the Irish data.

In our future work we would like to extend our approaches so that they allow us to specify the reach of the metro-nodes. Consequently, this may make some problem instances inconsistent. Therefore it would also be interesting to extend the problem definition where only a given percentage of total customers are required to be dually covered.

ACKNOWLEDGEMENTS

This paper was supported by Science Foundation Ireland under grant 08/CE/I1423.

REFERENCES

[1] D. B. Payne, "FTTP deployment options and economic challenges," in *Proceedings of the 36th European Conference and Exhibition on Optical Communication (ECOC 2009)*, 2009.

[2] D. K. Hunter, Z. Lu, and T. H. Gilfedder, "Protection of long-reach PON traffic through router database synchronization," *Journal of Optical Communications and Networking*, vol. 6, no. 5, pp. 535–549, 2007.

[3] L. Kaufman and P. Rousseeuw, *Finding Groups in Data An Introduction to Cluster Analysis*. New York: Wiley Interscience, 1990.

[4] A. J. Phillips, J. M. Senior, R. Mercinelli, M. Valvo, P. J. Vetter, C. M. Martin, M. O. van Deventer, P. Vaes, and X. Z. Qiu, "Redundancy strategies for a high splitting optically amplified passive optical network," *Journal of Lightwave Technology*, vol. 19, no. 2, pp. 137–149, 2001.

[5] S. D. Maesschalck, D. Colle, A. Groebbens, C. Develder, A. Lievens, P. Lagasse, M. Pickavet, P. Demeester, F. Saluta, and M. Quagliatti, "Intelligent optical networking for multilayer survivability," *IEEE Communications Magazine*, vol. 40, no. 1, pp. 42–49, 2002.

[6] M. Ruffini, D. B. Payne, and L. Doyle, "Protection strategies for long-reach PON," in *Proceedings of the 36th European Conference and Exhibition on Optical Communication (ECOC 2010)*, 2010.

[7] M. Garey and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, 1979.

[8] S. P. Lloyd, "Least squares quantization in pcm," *IEEE Transactions on Information Theory*, vol. 28, pp. 129–137, 1982.