Chapter 5

Refined mathematical model

This chapter details the refinements made to the basic model of the previous chapter, focussing on improving model accuracy and real-world correlation. Refinements made include using Geographic Information System (GIS) datasets, coverage, different splitter types, economies of scale and fiber duct sharing. The refined model is tested with four datasets and analyzed before concluding that along with its improved accuracy, solution times are now infeasible for larger dataset sizes.

5.1 Model refinements

Using the basic model of chapter 4 as a basis, additional refinements can now be made to improve the accuracy of the model and the correlation it has to real-world PON deployments. As the results of the basic model have shown, there is a need to address cost overestimations in terms of fiber duct sharing and to use more accurate or closer to real-world datasets. To address the latter, the model will now be altered to accept GIS-mapped datasets.

5.1.1 Input data

GIS map data is essentially a geo-referenced set of points or *nodes*, defined by their latitude and longitude, connected with a subset of its neighbours through a set of trails as seen in figure 5.1. These trails are also known as *edges* and each one has a distinct cost or *weight*. A node that represents a location of a CO, splitter or ONU is of particular interest and is termed a Point of Interest (POI). Since these POIs correlate exactly to the placement of equipment in a real-world scenario, a model using GIS data is much more useful in practical applications.

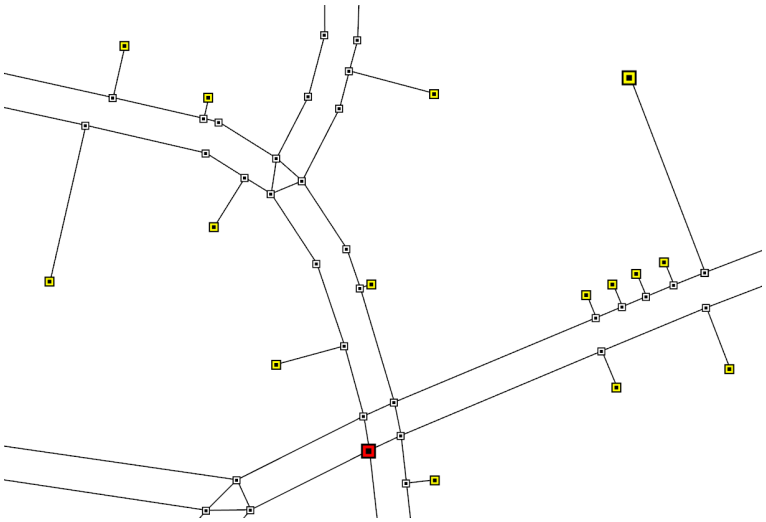


Figure 5.1: GIS data containing different node types and trails

Unfortunately, with the property that nodes are only connected to a subset of its neighbours comes the difficulty of finding the distance between two non-neighbouring or unconnected nodes. One way of finding the distance between two POIs is through the use of shortest-path algorithms. The two most widely used algorithms are Dijkstra's algorithm [53] and the Bellman-Ford algorithm [54, 55]. These find the shortest path between two nodes in a graph by systematically increasing a frontier line from the source node until it encompasses the destination node. The algorithm also continually keeps track of the shortest path calculated to reach the frontier.

The difference between the two algorithms is the Bellman-Ford algorithm's ability to handle graphs where the relative cost of moving from one node to another (its

weight) can be negative. Dijkstra's algorithm can only handle positive weights but it is faster, with a worst-case asymptotic performance of $\mathcal{O}(|E| + |V|log|V|)$ as opposed to Bellman-Ford's $\mathcal{O}(|V||E|)$, where E is the number of edges and V is the number of nodes or *vertices* on the map. Since distances between nodes on a map can never be negative, Dijkstra's algorithm is used, as specified in Appendix C.

Incorporating support for GIS datasets into the basic model is simple, requiring no changes to the model. However, a pre-processing step is required to determine shortest path distances between all POIs. This step involves running Dijkstra's algorithm with the CO as source node, saving all distances to splitters. Next, it is run with every splitter as source nodes, saving distances to all ONUs. Therefore, the algorithm is run $|\mathbf{S}| + 1$ times, recording distances in ℓ_i^{CO} and ℓ_{ij}. Luckily, since each iteration of the algorithm is independent, the process can be multi-threaded, reducing the time to calculate the required paths by a factor of n when using an n-core processor.

5.1.2 Fiber duct sharing

As node density increases, the probability of two or more fibers sharing some part of their respective routes increases dramatically. This is especially true for datasets with civil constraints, where nodes will share potential routes alongside roads or use the same underground ducts. With trenching cost being one of the largest expenses in a PON deployment, fiber duct sharing or path sharing can bring about a significant saving, making it an essential addition to the model.

Even though the concept of sharing trenching costs between a number of interleaving paths is simple, as illustrated in figure 5.2, implementation proves to be a lot more complex. The solution lies with the use of a modelling technique used in network flow optimization, where flows of commodities can share paths between core routers before being routed to a specific path at the edge of the network. To constrain the flows contained within these shared paths due to limited bandwidth, it is necessary to be able to access both the total path and the intermediary points in the path, called

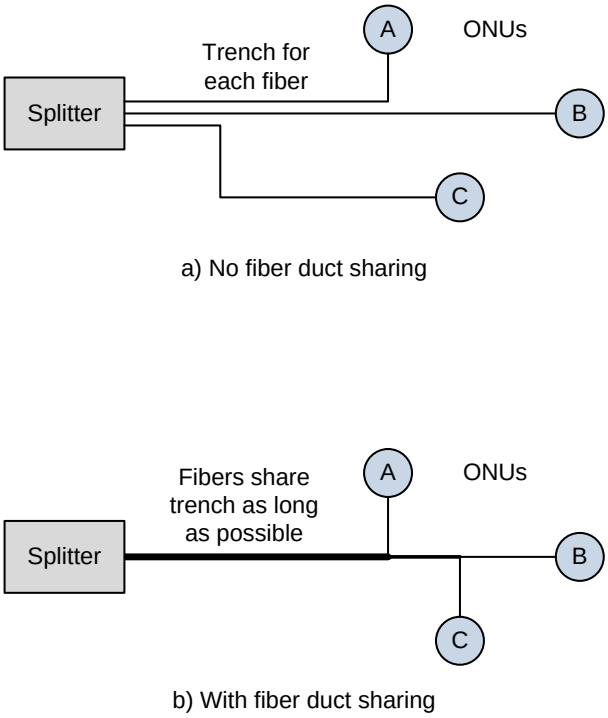


Figure 5.2: The concept of fiber duct sharing

edges.

This approach can be manipulated for use in PON planning, with the path representing the fiber and the edges representing trench segments. Therefore each path is unique but edges can be shared among paths, effectively allowing for fiber duct sharing. Keeping with the network flow terminology, we can now define:

- **Commodity pair** - Two POIs that need to be connected, i.e. a CO and a splitter or a splitter and an ONU. The set of all relevant commodity pairs is defined as $\mathbf{K} \neq \varnothing$.
- **Path** - A link between a *commodity pair*, consisting of a series of interconnected edges and representing a fiber link. The set of all paths between commodities is given by $\mathbf{P} \neq \varnothing$. Note that more than one path can exist between the same commodity pair.
- **Edge** - A link between two neighbouring *nodes* and also the smallest building block of a path, representing a trench segment. The set $\mathbf{E} \neq \varnothing$ contains all edges

in the map.

Figure 5.3 further illustrates the differences between these terms.

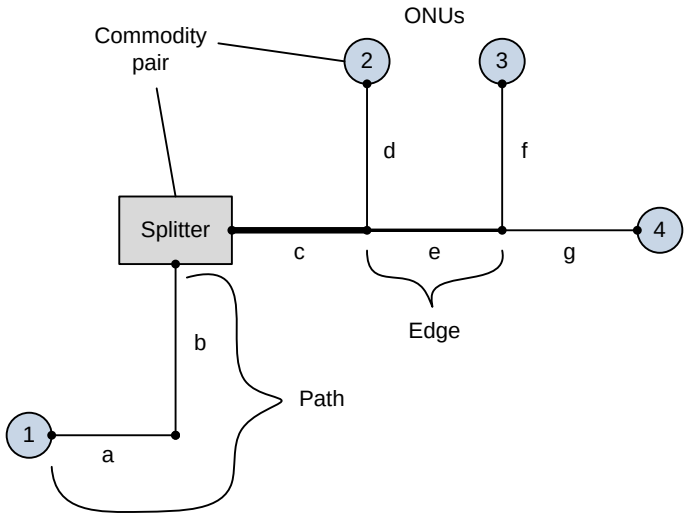


Figure 5.3: Fiber duct sharing terminology

Modelling fiber duct sharing

To be able to model fiber duct sharing, the link usage variable of the basic model, ϕ_{ij}, is replaced by two new binary variables, y_p and x_e, indicating the usage of the p-th path and e-th edge respectively. This separates the usage of a link into its trenching and fiber components as required.

Linking the behaviour of the two components can now be done in the form of a constraint as in equation (5.1). This constraint ensures that when a path is used, all the edges contained within it should be marked as used. Then the rest of the constraints can refer to path usage, with the relevant edges being marked as used automatically. Let $\mathbf{P}(e) \subseteq \mathbf{P}$ be a subset consisting of all paths containing edge e.

$$\sum_{p \in \mathbf{P}(e)} y_p \leq \Delta x_e, \quad \forall e \in \mathbf{E} \tag{5.1}$$

Note that in this case, $\Delta \geq |\mathbf{P}(e)|$, where Δ should be as small as possible to avoid numerical instability. The separate trenching and fiber costs can now be determined by noting that the total path cost can be determined by summing y_p over all paths. Similarly, the total edge or trenching cost can be determined by summing x_e over all edges. These costs can be seen in equations (5.2) and (5.3), where σ_p and σ_e denote the costs of the p-th path and e-th edge respectively.

$$C_{fiber} = \sum_{p \in \mathbf{P}} \sigma_p y_p \tag{5.2}$$

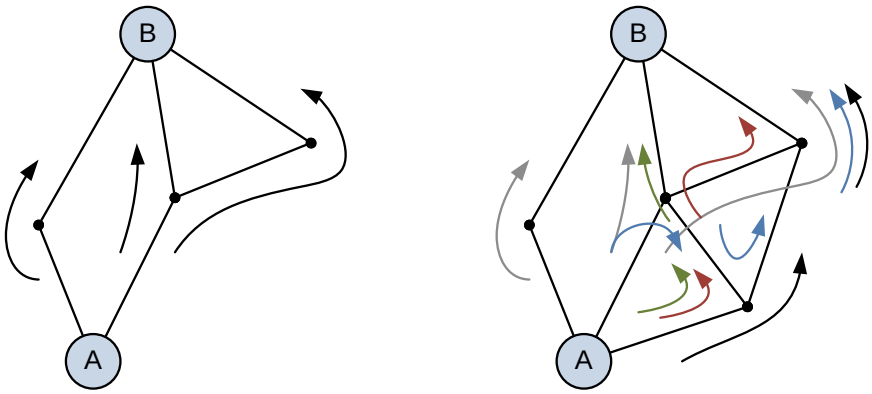
$$C_{trench} = \sum_{e \in \mathbf{E}} \sigma_e x_e \tag{5.3}$$

The per edge cost σ_e can be calculated by multiplying the cost per meter of trenching with the length of the e-th edge. The per path cost σ_p is determined by multiplying the cost per meter of fiber with the sum of all the edge lengths contained within the path.

Dealing with multiple paths

Since a path is defined as a link between a commodity pair, a number of paths can exist between any two POIs. With larger, highly connected maps, the number of possible paths between commodity pairs increase at an exponential rate, making the computation of all paths infeasible. Figure 5.4 illustrates how the addition of a single node can increase the number of paths. Therefore, to be able to solve the newly proposed model, a heuristic needs to be implemented.

The simplest heuristic to limit the number of paths between commodity pairs is to only use the shortest path between them. This method disregards a large number of paths that may have yielded a lower total PON deployment cost due to them being able to share a larger portion of their length with another path. It can be argued however, that since fiber cost is of the same order of magnitude as trenching cost, the effective cost of the shortest path will not be orders of magnitude larger than a longer path that shares a longer trench.



a) 3 non-cyclic paths b) 7 non-cyclic paths

Figure 5.4: Possible paths between commodity pairs with different numbers of intermediary nodes

A compromise between calculating all the paths and calculating just the shortest path can be made by using a k shortest path algorithm such as Yen's algorithm [56,57]. This type of algorithm, as the name suggests, produces the k shortest paths between commodity pairs. Therefore, paths can be generated which are slightly longer than the shortest path to allow for more potential sharing. Paths that are substantially longer, likely having a higher overall cost regardless of the amount of sharing, will be discarded.

With a worst-case running time of $\mathcal{O}(k|V|(|E|+|V|log|V|))$ [57], where k is the number of paths to generate, V is the set of nodes and E is the set of edges, it can be noted that the time required to generate paths increase linearly with k. Comparing this with the time required to generate only the shortest path ($k = 1$) using Dijkstra's algorithm, $\mathcal{O}(|E|+|V|log|V|)$, it is clear that the k shortest path case is $k|V|$ times slower, which can be excessive for large datasets. It is therefore necessary to determine the feasibility of using more than one shortest path in the model.

5.1.3 Non-symmetrical fiber cost

One of the assumptions made for the basic model is that fiber between the CO and splitters and fiber between splitters and ONUs have the same cost per unit of distance. In reality, this is not always the case, since the feeder fiber between the CO and a splitter may have additional shielding and protection, for a higher cost per unit of distance. This is due to the fact that if the feeder fiber fails, all downstream links will fail, whereas a failed fiber between the splitter and ONU will affect only the connected ONU.

The reverse can also be true, since feeder fibers may be bundled in larger cables, reducing the overall cost per fiber per unit of distance. To account for these differences in cost between feeder and distribution fibers, non-symmetrical fiber cost can be introduced into the model.

To split the fiber and trench costs as specified in equations (5.2) and (5.3) to account for different types of fiber, the concept of the commodity pairs can be utilized. Firstly, let k_{ij}^{SP-ONU} refer to the commodity pair of the i-th splitter and the j-th ONU, with $i \in \mathbf{S}, j \in \mathbf{U}$. Similarly, let k_i^{CO-SP} refer to the commodity pair of the CO and the i-th splitter. Next, define separate edge costs $\sigma_e^{CO-SP}, \sigma_e^{SP-ONU}$ and separate path costs $\sigma_p^{CO-SP}, \sigma_p^{SP-ONU}$.

Defining $\mathbf{P}(k) \subseteq \mathbf{P}$ as the subset of paths for commodity pair k allows one to specify $\mathbf{P}(k_i^{CO-SP})$, which is all paths from the CO to the i-th splitter. The same can be done for the splitter to ONU fibers by setting $k = k_{ij}^{SP-ONU}$ to get all paths from the i-th splitter to the j-th ONU. By summing for all splitters and ONUs, we arrive at equations (5.4) and (5.5).

$$C_{fiber}^{CO-SP} = \sum_{i \in \mathbf{S}} \sum_{p \in \mathbf{P}(k_i^{CO-SP})} \sigma_p^{CO-SP} y_p \tag{5.4}$$

$$C_{fiber}^{SP-ONU} = \sum_{j \in \mathbf{U}} \sum_{i \in \mathbf{S}} \sum_{p \in \mathbf{P}(k_{ij}^{SP-ONU})} \sigma_p^{CO-SP} y_p \tag{5.5}$$

Let $\mathbf{E}(k) \subseteq \mathbf{E}$ now be the subset of edges contained within paths of commodity pair k. Stated differently, $\mathbf{E}(k)$ is the set of edges contained within all paths in the set $\mathbf{P}(k)$. Therefore $\mathbf{E}(k_i^{CO-SP})$ is the set of all edges contained within all the paths from the CO to the i-th splitter. $\mathbf{E}(k_{ij}^{SP-ONU})$ is the set of all edges contained within all the paths between the i-th splitter and the j-th ONU. The edge costs can then be defined as in equations (5.6) and (5.7).

$$C_{trench}^{CO-SP} = \sum_{i \in \mathbf{S}} \sum_{e \in \mathbf{E}(k_i^{CO-SP})} \sigma_e^{CO-SP} x_e \tag{5.6}$$

$$C_{trench}^{SP-ONU} = \sum_{j \in \mathbf{U}} \sum_{i \in \mathbf{S}} \sum_{e \in \mathbf{E}(k_{ij}^{SP-ONU})} \sigma_e^{SP-ONU} x_e \tag{5.7}$$

Equivalently, by utilizing the commodity concept and a subset $\mathbf{P}(k,e) \subseteq \mathbf{P}(k)$, consisting of all paths for the k-th commodity pair that contains the e-th edge, the constraint defined in equation (5.1) can be rewritten as equation (5.8). This standardizes the notation by defining subsets by commodity pairs.

$$\sum_{k \in \mathbf{K}} \sum_{p \in \mathbf{P}(k,e)} y_p \leq \Delta x_e, \quad \forall e \in \mathbf{E} \tag{5.8}$$

Through the introduction of non-symmetrical fiber cost and fiber duct sharing into the model, the rest of the basic model constraints can be converted, with the equivalent versions of constraints (4.8), (4.11) and (4.12) given in equations (5.9), (5.10) and (5.11) respectively. This is done by substituting ϕ_{ij} for $\sum_{p \in \mathbf{P}(k_{ij}^{SP-ONU})} y_p$, which now specifies the *total number* of links between the i-th splitter and j-th ONU.

$$\sum_{i \in \mathbf{S}} \sum_{p \in \mathbf{P}(k_{ij}^{SP-ONU})} y_p = 1, \quad \forall j \in \mathbf{U} \tag{5.9}$$

$$\sum_{j \in \mathbf{U}} \sum_{p \in \mathbf{P}(k_{ij}^{SP-ONU})} y_p \leq \Delta \psi_i, \quad \forall i \in \mathbf{S} \tag{5.10}$$

$$\sum_{j \in \mathbf{U}} \sum_{p \in \mathbf{P}(k_{ij}^{SP-ONU})} y_p \leq \kappa_{ONU}^{SP}, \quad \forall i \in \mathbf{S} \tag{5.11}$$

5.1.4 Multiple central offices

Instances where more than one CO is available allows the model to optimize cost by splitting the available POIs into two disjunct PONs. To expand the model for these instances the addition of a new binary usage variable, ϕ_ℓ, indicating the usage of the ℓ-th CO, along with minor adjustments to the current constraints are necessary.

Firstly, summation variables Φ, Ψ and Π are defined to make the formulation of equations that include the total number of used Central Offices (COs), splitters or ONUs less obtuse. Therefore:

$$\sum_{i \in \mathbf{S}} \psi_i = \Psi \tag{5.12}$$

$$\sum_{j \in \mathbf{U}} \pi_j = \Pi \tag{5.13}$$

$$\sum_{\ell \in \mathbf{C}} \phi_\ell = \Phi \tag{5.14}$$

with the set $\mathbf{C}$ defined as the set of all CO locations. Now, define $k_{i\ell}^{CO-ONU}$ as the commodity pair between the ℓ-th CO and the i-th splitter. To ensure that each splitter is connected to one and only one CO, equation (5.9) can be adapted by substituting the correct sets and commodity parameters as done in equation (5.15). Since not all splitters are used, the right side of the equation cannot be 1 as it would force a connection between a CO and unused splitters. Instead, ψ_i is used, which takes on a value of 1 if that splitter is used and 0 otherwise. Therefore the constraint only applies to used splitters.

$$\sum_{\ell \in \mathbf{C}} \sum_{p \in \mathbf{P}(k_{i\ell}^{CO-SP})} y_p = \psi_i, \quad \forall i \in \mathbf{S} \tag{5.15}$$

To ensure that when a path exists between a CO and splitter, the CO is marked used, a constraint similar to equation (5.10) is used. Once again the variables and sets are substituted as done in (5.16) below.

$$\sum_{i \in \mathbf{S}} \sum_{p \in \mathbf{P}(k_{i\ell}^{CO-SP})} y_p \leq \Delta \phi_\ell, \quad \forall \ell \in \mathbf{C} \tag{5.16}$$

Now, to ensure that at least one, but no more than κ_{CO}, COs are used, two additional constraints are necessary, as specified by the inequalities in equations (5.17) and (5.18) below.

$$\sum_{\ell \in \mathbf{C}} \phi_\ell \geq 1 \tag{5.17}$$

$$\sum_{\ell \in \mathbf{C}} \phi_\ell \leq \kappa_{CO} \tag{5.18}$$

where the summation of all CO usage variables ϕ_ℓ in $\sum_{\ell \in \mathbf{C}} \phi_\ell$ represents the total number of COs used.

Finally, all constraints containing the commodity pair parameter k_i^{CO-SP} needs to be substituted with $k_{i\ell}^{CO-SP}$, since the assumption of one CO does not hold any more. Due to the extra index ℓ, the constraints also need to be defined either for all COs or summed for all COs. At this stage in the model, only the constraints defined in equations (5.4) and (5.6) contain the parameter k_i^{CO-SP} and are thus the only constraints that need to be modified.

Since both equations calculate a total cost, they can be easily converted by summing over the additional index ℓ. Therefore they can be written as:

$$C_{fiber}^{CO-SP} = \sum_{\ell \in \mathbf{C}} \sum_{i \in \mathbf{S}} \sum_{p \in \mathbf{P}(k_{i\ell}^{CO-SP})} \sigma_p^{CO-SP} y_p \tag{5.19}$$

$$C_{trench}^{CO-SP} = \sum_{\ell \in \mathbf{C}} \sum_{i \in \mathbf{S}} \sum_{e \in \mathbf{E}(k_{i\ell}^{CO-SP})} \sigma_e^{CO-SP} x_e \tag{5.20}$$

5.1.5 Coverage

Network coverage, or number of homes passed, is the percentage of potential customer premises connected to the network. For PONs, it can then be defined as the percentage of ONUs connected to a splitter. Coverage can be used by SPs when deploying greenfield FTTH networks to determine the percentage of homes to connect to minimize their CAPEX per home. Since the expected income per home is usually fixed, a lower expenditure per home results in a higher expected Return on Investment (ROI). However, this only applies to SPs that are not mandated by regulation or government to connect all homes to the network.

To allow for network coverage, the assumption of the basic model that all ONUs is used is removed. Next, a coverage factor ω, with $0 \leq \omega \leq 1$, is introduced into the model, specifying the percentage of ONUs to use. Ideally, the model should optimize for the minimum cost per ONU, but the objective function formulation thereof:

Minimize:

$$C_{perONU} = C_{total} / \Pi \tag{5.21}$$

will make the function non-linear, as both C_{total} and Π are variable. To avoid this, another approach is followed by making the coverage factor, ω, a fixed value parameter. The model can then be solved externally for a number of different values for ω, comparing the cost per ONU of each solution until an approximately optimal value for ω can be found.

Incorporating the coverage factor entails the modification of one constraint and the addition of another. Equation (5.9), which specifies that all ONUs must be connected to a splitter, must be altered to apply only to used ONUs. This can be done simply by substituting π_j in the right side of the equation, which takes on a value of 1 if the

j-th ONU is used and 0 otherwise, activating the constraint only for used ONUs as required. This modification is shown in equation (5.22) below.

$$\sum_{i \in \mathbf{S}} \sum_{p \in \mathbf{P}(k_{ij}^{SP-ONU})} y_p = \pi_j, \quad \forall j \in \mathbf{U} \tag{5.22}$$

Finally, to ensure that a certain percentage of the available ONUs is used, the total number of used ONUs, Π, can be constrained to be $\omega|\mathbf{U}|$. This constraint can therefore be combined with equation (5.13) as follows:

$$\sum_{j \in \mathbf{U}} \pi_j = \omega|\mathbf{U}| = \Pi \tag{5.23}$$

5.1.6 Network constraints

Recall from chapter 2 that all fiber networks have a limited fiber reach distance due to attenuation, insertion loss and splices. To avoid synchronization issues, the PON standards also specify a differential distance - the maximum difference between the minimum and maximum fiber distances between the CO and ONUs. Once again, for better correlation to real-world PONs, the refined model should incorporate these network constraints.

To formulate the network constraints as model constraints, a similar approach to Li. et al [2] is followed. In this paper, the authors define an intermediary *if-then* binary variable that activate or deactivate two constraints, based on whether a link exists between a splitter and an ONU. For this model, the binary variable d_{ij}^{SP} will be defined for this very purpose, which will be 0 if a path exists between the i-th splitter and the j-th ONU and 1 otherwise.

Realizing this *if-then* variable as a constraint can be done by using proposition 4.1, with $B = (1 - d_{ij}^{SP})$. Therefore, if $B = 1$, then d_{ij}^{SP} must equal 0. To determine if a link exists between the i-th splitter and the j-th ONU, the path usage variable y_p can be

used, along with the possible paths for commodity pair k_{ij}^{SP-ONU}. Putting this together and defining the constraint for all splitters, $i \in \mathbf{S}$, and ONUs, $j \in \mathbf{U}$, we are left with equation (5.24).

$$\sum_{p \in \mathbf{P}(k_{ij}^{SP-ONU})} y_p \leq \Delta(1 - d_{ij}^{SP}), \quad \forall i \in \mathbf{S}, \forall j \in \mathbf{U} \tag{5.24}$$

Next, the minimum and maximum distances between a CO and ONU can be defined as done in Li. et al [2] as follows:

$$\ell_i^{MIN} - (\ell_i^s + \ell_i^j) \leq \Delta d_{ij}^{SP}, \quad \forall i \in \mathbf{S}, \forall j \in \mathbf{U} \tag{5.25}$$

$$(\ell_i^s + \ell_i^j) - \ell_i^{MAX} \leq \Delta d_{ij}^{SP}, \quad \forall i \in \mathbf{S}, \forall j \in \mathbf{U} \tag{5.26}$$

where ℓ_i^s represents the fiber distance between the i-th splitter and its associated CO and ℓ_i^j represents the fiber distance between the i-th splitter and j-th ONU. For these constraints, it is clear that when $d_{ij}^{SP} = 0$, the inequalities become $(\ell_i^s + \ell_i^j) \leq \ell_i^{MAX}$ and $\ell_i^{MIN} \leq (\ell_i^s + \ell_i^j)$ respectively, ensuring that the minimum and maximum distances ℓ_i^{MIN} and ℓ_i^{MAX} are set accordingly. Since ℓ_i^s and ℓ_i^j can be defined as:

$$\ell_i^s = \sum_{\ell \in \mathbf{C}} \sum_{p \in \mathbf{P}(k_{i\ell}^{CO-SP})} y_p \ell_p^{CO-SP} \tag{5.27}$$

$$\ell_i^j = \sum_{p \in \mathbf{P}(k_{ij}^{SP-ONU})} y_p \ell_p^{SP-ONU} \tag{5.28}$$

with ℓ_p^{CO-SP} the length of the path p between CO and splitter and ℓ_p^{SP-ONU} the length of the path p between splitter and ONU. Now, by substituting (5.27) and (5.28) into equations (5.25) and (5.26), the minimum and maximum distance constraints can be constructed as follows:

$$\ell_i^{MIN} - \Big(\sum_{\ell \in \mathbf{C}} \sum_{p \in \mathbf{P}(k_{i\ell}^{CO-SP})} y_p \ell_p^{CO-SP} + \sum_{p \in \mathbf{P}(k_{ij}^{SP-ONU})} y_p \ell_p^{SP-ONU} \Big) \leq \Delta d_{ij}^{SP}, \quad \forall i \in \mathbf{S}, \forall j \in \mathbf{U} \tag{5.29}$$

$$\Big(\sum_{\ell \in \mathbf{C}} \sum_{p \in \mathbf{P}(k_{i\ell}^{CO-SP})} y_p \ell_p^{CO-SP} + \sum_{p \in \mathbf{P}(k_{ij}^{SP-ONU})} y_p \ell_p^{SP-ONU} \Big) - \ell_i^{MAX} \leq \Delta d_{ij}^{SP}, \quad \forall i \in \mathbf{S}, \forall j \in \mathbf{U} \tag{5.30}$$

Finally, now that ℓ_i^{MIN} and ℓ_i^{MAX} has been determined for each splitter i, the network constraints can be formulated:

$$\ell_i^{MAX} \leq \ell_{MAX}^{TOTAL}, \quad \forall i \in \mathbf{S} \tag{5.31}$$

$$\ell_i^{MAX} - \ell_i^{MIN} \leq \ell_{MAX}^{DIFF}, \quad \forall i \in \mathbf{S} \tag{5.32}$$

where ℓ_{MAX}^{TOTAL} is the maximum fiber reach and ℓ_{MAX}^{DIFF} is the maximum differential fiber distance.

5.1.7 Splitter types

As one would expect, passive splitters come in a range of sizes or split ratios, from 1:2 all the way up to 1:128, with higher split ratio splitters fetching a higher premium due to the additional output ports and required optics. Furthermore, it would be suboptimal to use only a single type of splitter throughout the entire deployment, as some splitters might have 100 ONUs connected to them while others have only 5. This would necessitate the use of expensive 1:128 splitters in situations where a 1:8 splitter might have sufficed.

Therefore, by assigning the *best* type of splitter for a given location, in this case the one

with the lowest split ratio that can still satisfy the requirements, the total cost of the deployment can be minimized.

Incorporating splitter types into the model requires the definition of a new set $\mathbf{T} \neq \varnothing$, consisting of all the available types of splitters. To differentiate between the splitter types, a parameter N_t^{SP} is required, specifying the number of output ports for each splitter type $t \in \mathbf{T}$. Finally, the new binary usage variable τ_{it}^{SP} is defined, taking on a value of 1 if the i-th splitter is of type t and 0 otherwise.

Two constraints are now required to fully define the behaviour of τ_{it}^{SP}. Firstly, each splitter must be of one and only one type and secondly, the splitter type must have sufficient capacity to connect to the number of allocated ONUs. The first constraint can be formulated easily, by noting that the summation of τ_{it}^{SP} over all splitter types $t \in \mathbf{T}$ produces the number of splitter types associated with the i-th splitter. Therefore, to ensure each used splitter is of one and only one type, we can define:

$$\sum_{t \in \mathbf{T}} \tau_{it}^{SP} = \psi_i, \quad \forall i \in \mathbf{S} \tag{5.33}$$

Secondly, note that the product $N_t^{SP}\tau_{it}^{SP}$ takes on a value equal to the t-th splitter type's capacity if the i-th splitter is of type t. Since equation (5.33) ensures only a single type can be active for a given splitter, the summation of this product over all splitter types t will produce the capacity of the i-th splitter, CAP_i^{SP}, as defined in equation (5.34).

$$CAP_i^{SP} = \sum_{t \in \mathbf{T}} N_t^{SP}\tau_{it}^{SP} \tag{5.34}$$

Now that the capacity for the i-th splitter is known, the number of ONUs associated with the i-th splitter needs to be determined. Conveniently, this value was already calculated in equation (5.11), where the number of ONUs per splitter was constrained. Therefore, we can ensure that the capacity of each splitter is always sufficient to serve the number of ONUs connected to it by defining constraint (5.35).

$$\sum_{j \in \mathbf{U}} \sum_{p \in \mathbf{P}(k_{ij}^{SP-ONU})} y_p \leq \sum_{t \in \mathbf{T}} N_t^{SP} \tau_{it}^{SP}, \quad \forall i \in \mathbf{S} \tag{5.35}$$

The total splitter cost in the objective function can now be modified to include different splitter types. Similar to the definition of CAP_i^{SP}, the product $c_t^{SP}\tau_{it}^{SP}$ takes on a value equal to the cost of the t-th type of splitter if the i-th splitter is of type t, with c_t^{SP} the cost of a splitter of type t. The summation of this product over all splitter types produces the cost of the i-th splitter. Finally, the cost of *all* splitters can be determined by summing over all splitters as follows:

$$C_{splitter} = \sum_{i \in \mathbf{S}} \sum_{t \in \mathbf{T}} c_t^{SP} \tau_{it}^{SP} \tag{5.36}$$

5.1.8 Economies of scale

A topic requiring almost no introduction, Economies of Scale (EOS) is the concept that as the production volume of a product increase, its unit cost decreases. The same phenomena is encountered when buying equipment such as splitters or ONUs. As more of a specific piece of equipment is bought, each unit becomes cheaper due to the leverage of EOS. Since EOS is so prevalent in real-world network deployments where equipment is purchased in large volumes, it is important to incorporate it into the refined model.

EOS is usually illustrated by plotting total product cost against product volume as shown in figure 5.5. This is also an apt way to visualize the cost of equipment in the refined model, as splitter and ONU costs are defined as total costs in the objective function. Immediately though, a problem in the formulation is apparent, since the relationship between total cost and volume is inherently non-linear. A linear relationship would imply no EOS and fixed unit costs.

Fortunately, since real-world pricing is usually done in price brackets based on volume,

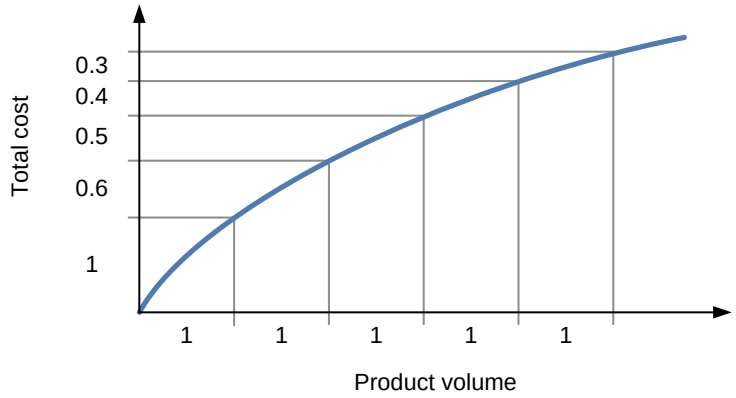


Figure 5.5: Effect of economies of scale on total product cost

EOS can be approximated very well using stepped or piecewise linear approximations. This method relies on the fitting of straight lines on a curve to approximate the underlying curve's changing gradient. Stated differently, the method samples a curve and interpolates linearly between the points, as seen in figure 5.6.

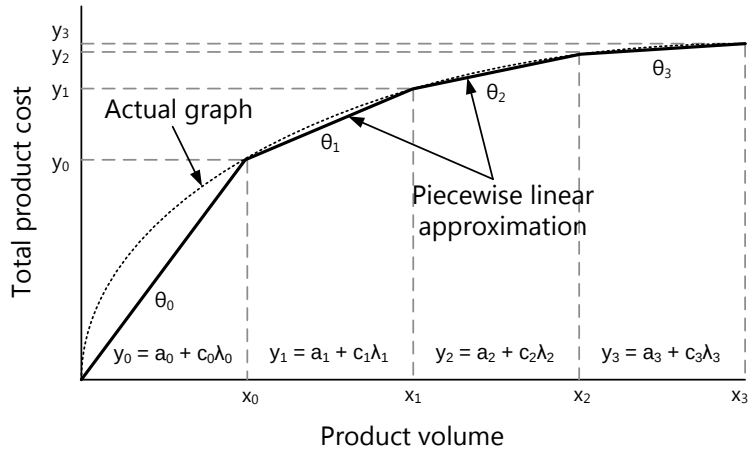


Figure 5.6: Piecewise linear approximation of total product cost

A number of ways exist to formulate a piecewise linear function, usually through utilization of the offsets and gradients of the individual straight line functions. An elegant way of modelling a piecewise linear function is given in [58], where the output of the approximated non-linear function can be accessed directly and where the offsets and gradients of the linear functions are not required. Instead, the formulation relies on discrete sets of points (x, y) describing the function, which is then interpolated linearly through the use of convex combinations.

Given the function in figure 5.6 for example, any point between (x_i, y_i) and (x_{i+1}, y_{i+1}) can be determined through the use of the convex combinations:

$$x = \lambda x_i + (1 - \lambda) x_{i+1} \tag{5.37}$$

$$y = \lambda y_i + (1 - \lambda) y_{i+1} \tag{5.38}$$

Using this principle, a set of equations describing any point (x, y) on the piecewise linear function defined by N points (x_i, y_i), with $i = \{0, 1, \ldots, N-1\}$, can be constructed as given in [58]:

$$x = \sum_{i=0}^{N-1} \lambda_i x_i \tag{5.39}$$

$$y = \sum_{i=0}^{N-1} \lambda_i y_i \tag{5.40}$$

$$0 \leq \lambda_0 \leq \theta_1 \tag{5.41}$$

$$0 \leq \lambda_n \leq \theta_n + \theta_{n+1}, \quad \forall n \in \{1, \ldots, N-2\} \tag{5.42}$$

$$0 \leq \lambda_{N-1} \leq \theta_{N-1} \tag{5.43}$$

$$\sum_{i=1}^{N-1} \theta_i = 1, \quad \theta_i \in \{0, 1\} \tag{5.44}$$

$$\sum_{i=0}^{N-1} \lambda_i = 1, \quad \lambda_i \geq 0 \tag{5.45}$$

To include EOS for ONUs in the model, the above equations are used with some modifications. Firstly, since the x-axis of the function is the ONU volume, x is substituted with Π, the number of used ONUs. Similarly, the y-axis is the total cost, so y is substituted with $\tilde{c}^{ONU}$, the total ONU cost. The parameters x_i and y_i are replaced with v_m^{ONU} and c_m respectively, with v_m^{ONU} denoting the volume of ONUs and c_m^{ONU} the total ONU cost at the m-th point of the curve. Finally, after defining binary decision variables θ_m^{ONU} and continuous variables λ_m^{ONU}, the set of equations defining EOS for ONUs can be formulated as below:

$$\sum_{m \in \mathbf{M}} \lambda_m^{ONU} v_m^{ONU} = \Pi, \tag{5.46}$$

$$\sum_{m \in \mathbf{M}} \lambda_m^{ONU} c_m^{ONU} = \tilde{c}^{ONU}, \tag{5.47}$$

$$0 \leq \lambda_0^{ONU} \leq \theta_1^{ONU} \tag{5.48}$$

$$0 \leq \lambda_m^{ONU} \leq \theta_m^{ONU} + \theta_{m+1}^{ONU}, \quad m = \{1, \ldots, M-2\} \tag{5.49}$$

$$0 \leq \lambda_{M-1}^{ONU} \leq \theta_{M-1}^{ONU} \tag{5.50}$$

$$\sum_{m=1}^{M-1} \theta_m^{ONU} = 1 \tag{5.51}$$

$$\sum_{m \in \mathbf{M}} \lambda_m^{ONU} = 1 \tag{5.52}$$

$$\tag{5.53}$$

where **M** is the set of all points for the piecewise linear cost function of ONUs.

Similarly, to include EOS for every splitter type in the model, the variable x is substituted with the number of type t splitters used, which was defined earlier as $\sum_{i \in \mathbf{S}} \tau_{it}^{SP}$. The function value y is replaced with the total cost variable $\tilde{c}_t^{SP}$, denoting the total cost of type t splitters in the model. The parameter x_i is substituted with v_{nt}^{SP}, the volume of type t splitters at point n. y_i is substituted with c_{nt}^{SP}, the total cost of type t splitters at point n.

It is assumed that the piecewise linear cost function for every splitter type t has the same number of points, N, with the set of all points being defined as **N**. Then, with binary decision variables θ_{nt}^{SP} and continuous variables λ_{nt}^{SP}, the piecewise linear functions can be defined for all splitter types $t \in \mathbf{T}$ as follows:

$$\sum_{n \in \mathbf{N}} \lambda_{nt}^{SP} v_{nt}^{SP} = \sum_{i \in \mathbf{S}} \tau_{it}^{SP}, \quad \forall t \in \mathbf{T} \tag{5.54}$$

$$\sum_{n \in \mathbf{N}} \lambda_{nt}^{SP} c_{nt}^{SP} = \tilde{c}_t^{SP}, \quad \forall t \in \mathbf{T} \tag{5.55}$$

$$0 \leq \lambda_{0,t}^{SP} \leq \theta_{1,t}^{SP} \quad \forall t \in \mathbf{T} \tag{5.56}$$

$$0 \leq \lambda_{nt}^{SP} \leq \theta_{nt}^{SP} + \theta_{n+1,t}^{SP}, \quad n = \{1, \ldots, N-2\}, t \in \mathbf{T} \tag{5.57}$$

$$0 \leq \lambda_{N-1,t}^{SP} \leq \theta_{N-1,t}^{SP} \quad \forall t \in \mathbf{T} \tag{5.58}$$

$$\sum_{n=1}^{N-1} \theta_{nt}^{SP} = 1, \quad \forall t \in \mathbf{T} \tag{5.59}$$

$$\sum_{n \in \mathbf{N}} \lambda_{nt}^{SP} = 1, \quad \forall t \in \mathbf{T} \tag{5.60}$$

5.2 Final model

All the refinements of section 5.1 are now combined into one model. The sets, subsets, variables and parameters introduced can now be summarized before defining the final MILP model.

5.2.1 Sets

For the refined model, a number of new sets were defined, including sets for fiber duct sharing, splitter types and EOS.

- **U** - The set of all possible locations for ONUs. $\mathbf{U} \neq \varnothing$.
- **S** - The set of all possible locations for splitters. $\mathbf{S} \neq \varnothing$.
- **C** - The set of all possible sites for COs. $\mathbf{C} \neq \varnothing$.
- **P** - The set of all possible paths between all commodities. $\mathbf{P} \neq \varnothing$.
- **E** - The set of all edges used in all paths. $\mathbf{E} \neq \varnothing$.
- **K** - The set of all commodity pairs i.e. all possible pairs of ONUs, splitters and COs. $\mathbf{K} \neq \varnothing$.
- **M** - The set of all points of the piecewise linear cost function for ONUs. $\mathbf{M} = \{0, \ldots, M-1\}$.

- **N** - The set of all points of the piecewise linear cost function for all splitter types. $\mathbf{N} = \{0, \ldots, N-1\}$.
- **T** - The set of all splitter types.

5.2.2 Subsets

The three subsets included to allow for fiber duct sharing and non-symmetrical fiber cost are defined as follows:

- $\mathbf{P}(k) \subseteq \mathbf{P}$ - The set of all paths for commodity pair $k \in \mathbf{K}$ i.e. all possible paths between the two points referenced by k.
- $\mathbf{P}(k, e) \subseteq \mathbf{P}(k)$ - The set of all paths for commodity pair $k \in \mathbf{K}$ that contain edge $e \in \mathbf{E}$.
- $\mathbf{E}(k) \subseteq \mathbf{E}$ - The set of all edges used in all possible paths of commodity pair $k \in \mathbf{K}$.

5.2.3 Variables

Dwarfing the number of variables contained in the basic model, the refined model's newly defined variables include:

- ψ_i - Binary variable used to indicate usage of the i-th splitter, $i \in \mathbf{S}$. Takes on a value of 1 if the i-th splitter is used and 0 otherwise.
- π_j - Binary variable used to indicate usage of the j-th ONU, $j \in \mathbf{U}$. Takes on a value of 1 if the j-th ONU is used and 0 otherwise.
- ϕ_ℓ - Binary variable to indicate usage of ℓ-th CO, $\ell \in \mathbf{C}$. Takes on a value of 1 if the ℓ-th CO is used and 0 otherwise.

- x_e - Binary variable indicating the usage of the e-th edge, $e \in \mathbf{E}$. Takes on a value of 1 if the e-th edge is used and 0 otherwise.
- y_p - Binary variable indicating the usage of the p-th path, $p \in \mathbf{P}$. Takes on a value of 1 if the p-th path is used and 0 otherwise.
- τ_{it}^{SP} - Binary variable indicating the usage of a type t splitter for splitter i, $t \in \mathbf{T}$, $i \in \mathbf{S}$. Takes on a value of 1 if the i-th splitter is of type t and 0 otherwise.
- $\lambda_{nt}^{SP}, \lambda_{m}^{ONU}$ - Continuous variables used as coefficients in the convex combination when determining the correct cost segment to use given a volume of ONUs or splitters. $n \in \mathbf{N}$, $m \in \mathbf{M}$, $t \in \mathbf{T}$.
- $\theta_{nt}^{SP}, \theta_{m}^{ONU}$ - Intermediary binary selection variables used in the stepped cost graph selection. Takes on 1 if the n-th or m-th segment is selected respectively. $n \in \{1, \ldots, N-1\}$, $m \in \{1, \ldots, M-1\}$, $t \in \mathbf{T}$.
- $\tilde{c}_t^{SP}, \tilde{c}^{ONU}$ - The total cost for the currently selected cost point for splitter type t and ONUs respectively. $t \in \mathbf{T}$.
- ℓ_i^{MIN} - Minimum distance of ONU to CO associated with splitter $i \in \mathbf{S}$.
- ℓ_i^{MAX} - Maximum distance of ONU to CO associated with splitter $i \in \mathbf{S}$.
- d_{ij}^{SP} - Intermediary *if-then* binary variable used to determine minimum and maximum distances between ONUs and CO. $i \in \mathbf{S}$, $j \in \mathbf{U}$.
- Ψ - The total number of splitters used.
- Π - The total number of ONUs used.
- Φ - The total number of COs used.

5.2.4 Parameters

A more extensive set of parameters are required for the refined model, pertaining to splitter types, EOS cost functions and predefined network constraint distances.

- C_{OLT} - The fixed OLT cost incurred for each CO.
- κ_{ONU}^{SP} - The maximum number of ONUs that can connect to each splitter.
- κ_{CO} - The maximum number of COs to deploy.
- ω - The coverage factor percentage. $0 \leq \omega \leq 1, \omega \in \mathbb{R}$.
- σ_e^{CO-SP} - The cost of edge $e \in \mathbf{E}$ between CO and splitter. Refers to trenching cost in this instance.
- σ_e^{SP-ONU} - The cost of edge $e \in \mathbf{E}$ between splitter and ONU. Refers to trenching cost in this instance.
- σ_p^{CO-SP} - The cost of path $p \in \mathbf{P}$ between CO and splitter. Refers to fiber cost in this instance.
- σ_p^{SP-ONU} - The cost of path $p \in \mathbf{P}$ between splitter and ONU. Refers to fiber cost in this instance.
- ℓ_p^{CO-SP} - The length of path $p \in \mathbf{P}$ between CO and splitter.
- ℓ_p^{SP-ONU} - The length of path $p \in \mathbf{P}$ between splitter and ONU.
- ℓ_{MAX}^{TOTAL} - The maximum total reach between an ONU and CO.
- ℓ_{MAX}^{DIFF} - The maximum differential distance between two ONUs connected to a splitter.
- Δ - A large number.
- N_t^{SP} - The number of ports for splitter type $t \in \mathbf{T}$.
- N, M - The total number of points for the piecewise linear cost functions of splitters and ONUs respectively. $M > 0, N > 0$.
- v_{nt}^{SP}, v_m^{ONU} - The volume of the n-th and m-th cost point for splitter type t and ONUs respectively. $n \in \mathbf{N}$, $m \in \mathbf{M}$, $t \in \mathbf{T}$.

- c_{nt}^{SP}, c_{m}^{ONU} - The total cost for the n-th and m-th cost point for splitter type t and ONUs respectively. $n \in \mathbf{N}$, $m \in \mathbf{M}$, $t \in \mathbf{T}$.
- $k_{ij}^{SP-ONU} \in \mathbf{K}$ - The commodity pair referring to splitter $i \in \mathbf{S}$ and ONU $j \in \mathbf{U}$.
- $k_{i\ell}^{CO-SP} \in \mathbf{K}$ - The commodity pair referring to splitter $i \in \mathbf{S}$ and CO $\ell \in \mathbf{C}$.

5.2.5 MILP model

As was done in the basic ILP model, the modified objective function components and the constraints as described in section 5.1 are put together to form a MILP model that can be described as follows:

Minimize

$$C_{total} = C_{CO} + C_{splitter} + C_{ONU} + C_{fiber}^{CO-SP} + C_{fiber}^{SP-ONU} \tag{5.61}$$

where

$$C_{CO} = C_{OLT}\Phi \tag{5.62}$$

$$C_{splitter} = \sum_{t \in \mathbf{T}} \tilde{c}_t^{SP} \tag{5.63}$$

$$C_{ONU} = \tilde{c}^{ONU} \tag{5.64}$$

$$C_{fiber}^{CO-SP} = \sum_{\ell \in \mathbf{C}} \sum_{i \in \mathbf{S}} \Big[\sum_{e \in \mathbf{E}(k_{i\ell}^{CO-SP})} \sigma_e^{CO-SP} x_e + \sum_{p \in \mathbf{P}(k_{i\ell}^{CO-SP})} \sigma_p^{CO-SP} y_p \Big] \tag{5.65}$$

$$C_{fiber}^{SP-ONU} = \sum_{i \in \mathbf{S}} \sum_{j \in \mathbf{U}} \Big[\sum_{e \in \mathbf{E}(k_{ij}^{SP-ONU})} \sigma_e^{SP-ONU} x_e + \sum_{p \in \mathbf{P}(k_{ij}^{SP-ONU})} \sigma_p^{SP-ONU} y_p \Big] \tag{5.66}$$

subject to

$$\sum_{j \in \mathbf{U}} \pi_j = \omega |\mathbf{U}| = \Pi \tag{5.67}$$

$$\sum_{i \in \mathbf{S}} \psi_i = \Psi \tag{5.68}$$

$$\sum_{\ell \in \mathbf{C}} \phi_\ell = \Phi \tag{5.69}$$

$$\sum_{\ell \in \mathbf{C}} \phi_\ell \leq \kappa_{CO} \tag{5.70}$$

$$\sum_{\ell \in \mathbf{C}} \phi_\ell \geq 1 \tag{5.71}$$

$$\sum_{\ell \in \mathbf{C}} \sum_{p \in \mathbf{P}(k_{i\ell}^{CO-SP})} y_p = \psi_i, \quad \forall i \in \mathbf{S} \tag{5.72}$$

$$\sum_{i \in \mathbf{S}} \sum_{p \in \mathbf{P}(k_{ij}^{SP-ONU})} y_p = \pi_j, \quad \forall j \in \mathbf{U} \tag{5.73}$$

$$\sum_{i \in \mathbf{S}} \sum_{p \in \mathbf{P}(k_{i\ell}^{CO-SP})} y_p \leq \Delta \phi_\ell, \quad \forall \ell \in \mathbf{C} \tag{5.74}$$

$$\sum_{j \in \mathbf{U}} \sum_{p \in \mathbf{P}(k_{ij}^{SP-ONU})} y_p \leq \Delta \psi_i, \quad \forall i \in \mathbf{S} \tag{5.75}$$

$$\sum_{k \in \mathbf{K}} \sum_{p \in \mathbf{P}(k,e)} y_p \leq \Delta x_e, \quad \forall e \in \mathbf{E} \tag{5.76}$$

$$\sum_{j \in \mathbf{U}} \sum_{p \in \mathbf{P}(k_{ij}^{SP-ONU})} y_p \leq \kappa_{ONU}^{SP}, \quad \forall i \in \mathbf{S} \tag{5.77}$$

$$\sum_{j \in \mathbf{U}} \sum_{p \in \mathbf{P}(k_{ij}^{SP-ONU})} y_p \leq \sum_{t \in \mathbf{T}} N_t^{SP} \tau_{it}^{SP}, \quad \forall i \in \mathbf{S} \tag{5.78}$$

$$\sum_{t \in \mathbf{T}} \tau_{it}^{SP} = \psi_i, \quad \forall i \in \mathbf{S} \tag{5.79}$$

$$\ell_i^{MIN} - \Big(\sum_{\ell \in \mathbf{C}} \sum_{p \in \mathbf{P}(k_{i\ell}^{CO-SP})} y_p \ell_p^{CO-SP} + \sum_{p \in \mathbf{P}(k_{ij}^{SP-ONU})} y_p \ell_p^{SP-ONU} \Big) \leq \Delta d_{ij}^{SP}, \quad \forall i \in \mathbf{S}, \forall j \in \mathbf{U} \tag{5.80}$$

$$\Big(\sum_{\ell \in \mathbf{C}} \sum_{p \in \mathbf{P}(k_{i\ell}^{CO-SP})} y_p \ell_p^{CO-SP} + \sum_{p \in \mathbf{P}(k_{ij}^{SP-ONU})} y_p \ell_p^{SP-ONU} \Big) - \ell_i^{MAX} \leq \Delta d_{ij}^{SP}, \quad \forall i \in \mathbf{S}, \forall j \in \mathbf{U} \tag{5.81}$$

$$\sum_{p \in \mathbf{P}(k_{ij}^{SP-ONU})} y_p \leq \Delta (1 - d_{ij}^{SP}), \quad \forall i \in \mathbf{S}, \forall j \in \mathbf{U} \tag{5.82}$$

$$\ell_i^{MAX} \leq \ell_{MAX}^{TOTAL}, \quad \forall i \in \mathbf{S} \tag{5.83}$$

$$\ell_i^{MAX} - \ell_i^{MIN} \leq \ell_{MAX}^{DIFF}, \quad \forall i \in \mathbf{S} \tag{5.84}$$

$$\sum_{m \in \mathbf{M}} \lambda_m^{ONU} v_m^{ONU} = \Pi, \tag{5.85}$$

$$\sum_{m \in \mathbf{M}} \lambda_m^{ONU} c_m^{ONU} = \tilde{c}^{ONU}, \tag{5.86}$$

$$\sum_{n \in \mathbf{N}} \lambda_{nt}^{SP} v_{nt}^{SP} = \sum_{i \in \mathbf{S}} \tau_{it}^{SP}, \quad \forall t \in \mathbf{T} \tag{5.87}$$

$$\sum_{n \in \mathbf{N}} \lambda_{nt}^{SP} c_{nt}^{SP} = \tilde{c}_t^{SP}, \quad \forall t \in \mathbf{T} \tag{5.88}$$

$$0 \leq \lambda_0^{ONU} \leq \theta_1^{ONU} \tag{5.89}$$

$$0 \leq \lambda_m^{ONU} \leq \theta_m^{ONU} + \theta_{m+1}^{ONU}, \quad m = \{1, \ldots, M-2\} \tag{5.90}$$

$$0 \leq \lambda_{M-1}^{ONU} \leq \theta_{M-1}^{ONU} \tag{5.91}$$

$$0 \leq \lambda_{0,t}^{SP} \leq \theta_{1,t}^{SP} \quad \forall t \in \mathbf{T} \tag{5.92}$$

$$0 \leq \lambda_{nt}^{SP} \leq \theta_{nt}^{SP} + \theta_{n+1,t}^{SP}, \quad n = \{1, \ldots, N-2\}, t \in \mathbf{T} \tag{5.93}$$

$$0 \leq \lambda_{N-1,t}^{SP} \leq \theta_{N-1,t}^{SP} \quad \forall t \in \mathbf{T} \tag{5.94}$$

$$\sum_{m=1}^{M-1} \theta_m^{ONU} = 1 \tag{5.95}$$

$$\sum_{n=1}^{N-1} \theta_{nt}^{SP} = 1, \quad \forall t \in \mathbf{T} \tag{5.96}$$

$$\sum_{m \in \mathbf{M}} \lambda_m^{ONU} = 1 \tag{5.97}$$

$$\sum_{n \in \mathbf{N}} \lambda_{nt}^{SP} = 1, \quad \forall t \in \mathbf{T} \tag{5.98}$$

$$\psi_i \in \{0,1\}, \quad \forall i \in \mathbf{S} \tag{5.99}$$

$$\pi_j \in \{0,1\}, \quad \forall j \in \mathbf{U} \tag{5.100}$$

$$\phi_\ell \in \{0,1\}, \quad \forall \ell \in \mathbf{C} \tag{5.101}$$

$$x_e \in \{0,1\}, \quad \forall e \in \mathbf{E} \tag{5.102}$$

$$y_p \in \{0,1\}, \quad \forall p \in \mathbf{P} \tag{5.103}$$

$$\tau_{it}^{SP} \in \{0,1\}, \quad \forall i \in \mathbf{S}, \forall t \in \mathbf{T} \tag{5.104}$$

$$\theta_{nt}^{SP} \in \{0,1\}, \quad \forall n \in \mathbf{N}, \forall t \in \mathbf{T} \tag{5.105}$$

$$\theta_m^{ONU} \in \{0,1\}, \quad \forall m \in \mathbf{M} \tag{5.106}$$

$$d_{ij}^{SP} \in \{0,1\}, \quad \forall i \in \mathbf{S}, \forall j \in \mathbf{U} \tag{5.107}$$

$$\lambda_{nt}^{SP} \in \mathbb{R} \quad \forall n \in \mathbf{N}, \forall t \in \mathbf{T} \tag{5.108}$$

$$\lambda_{m}^{ONU} \in \mathbb{R} \quad \forall m \in \mathbf{M} \tag{5.109}$$

$$\tilde{c}_{t}^{SP} \in \mathbb{R} \quad \forall t \in \mathbf{T} \tag{5.110}$$

$$\tilde{c}^{ONU}, \ell_{i}^{MIN}, \ell_{i}^{MAX} \in \mathbb{R} \tag{5.111}$$

$$\Pi, \Phi, \Psi \in \mathbb{Z} \tag{5.112}$$

5.3 Testing methodology

As with the basic model, a testing methodology for the refined model is required, specifying the testing environment, the different types of input data and model parameters. A list of evaluation points whereby the relative *goodness* of the model is determined, is also needed. Therefore, this section will now discuss the testing methodology in detail.

5.3.1 Input datasets

In contrast to the theoretical, randomly generated datasets used in the evaluation of the basic model, the refined model will use actual GIS-mapped test data, generously provided by *atesio GmbH* [59]. This ensures that solutions generated by the refined model will have practical significance, allowing for much more practical conclusions to be drawn.

The data consisted of four individual scenarios, ranging in size from 1000 to close to 5000 ONUs, as detailed in table 5.1. The first scenario, a medium-sized area containing settlement clusters, will be termed *MedNet*, while the second, *SubNet*, contains a central town with outlying suburban areas. The third dataset, *CityNet*, has a large number of densely distributed nodes contained within a smaller region, while the largest dataset, termed *HugeNet*, contains clusters of densely populated areas, connected through sparser suburban or agricultural neighbourhoods.

Table 5.1: Detailed GIS-mapped dataset information

		Dataset			
Parameter	Unit	*MedNet*	*SubNet*	*CityNet*	*HugeNet*
Topology					
Nodes		7,935	4,389	6,698	24,018
Edges		8,607	4,785	7,660	26,800
PON					
ONUs		1,056	1,079	1,951	4,825
Splitters		402	149	239	1,034
COs		1	1	1	1
Splitter ratio		2.63	7.24	8.16	4.67
Geography					
Height	km	14.4	10.1	7.1	24.8
Width	km	13.2	11.6	2.3	16.1
Area	km^2	190.1	117.2	16.3	399.3
Node density	per km^2	41.7	37.4	410.9	60.2
ONU density	per km^2	5.6	9.2	119.7	12.1

Figures 5.8a to 5.8d show the ONU distribution in blue and the CO placement in red for each dataset. Splitter placements are shown in magenta, although distinguishing them from ONUs is difficult at this scale. Therefore, the central region of *MedNet* is shown at a larger scale in figure 5.7, with magenta triangles representing the possible splitter placements.

Lastly, to reduce the complexity of any of the original datasets, subsets can be generated by only taking the first n defined splitters or ONUs, treating all other points of interest as normal nodes. This allows the use of the topology of larger sets without having to deal with the complexity of additional splitters and ONUs, which can be useful when a test needs to be repeated a large number of times.

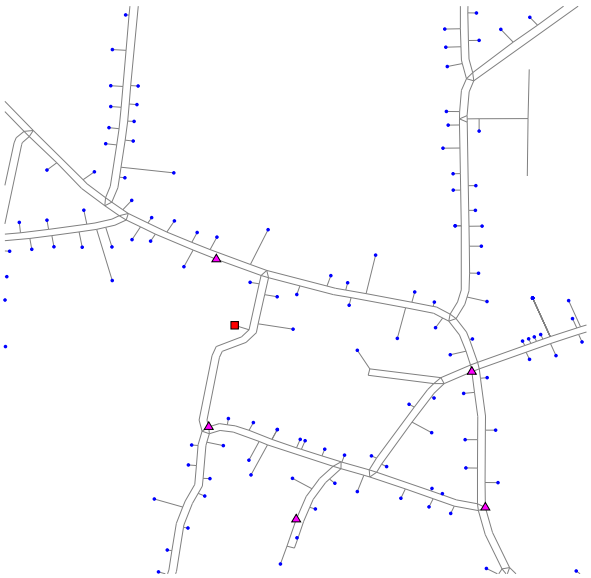


Figure 5.7: Larger scale plot of *MedNet*'s central region

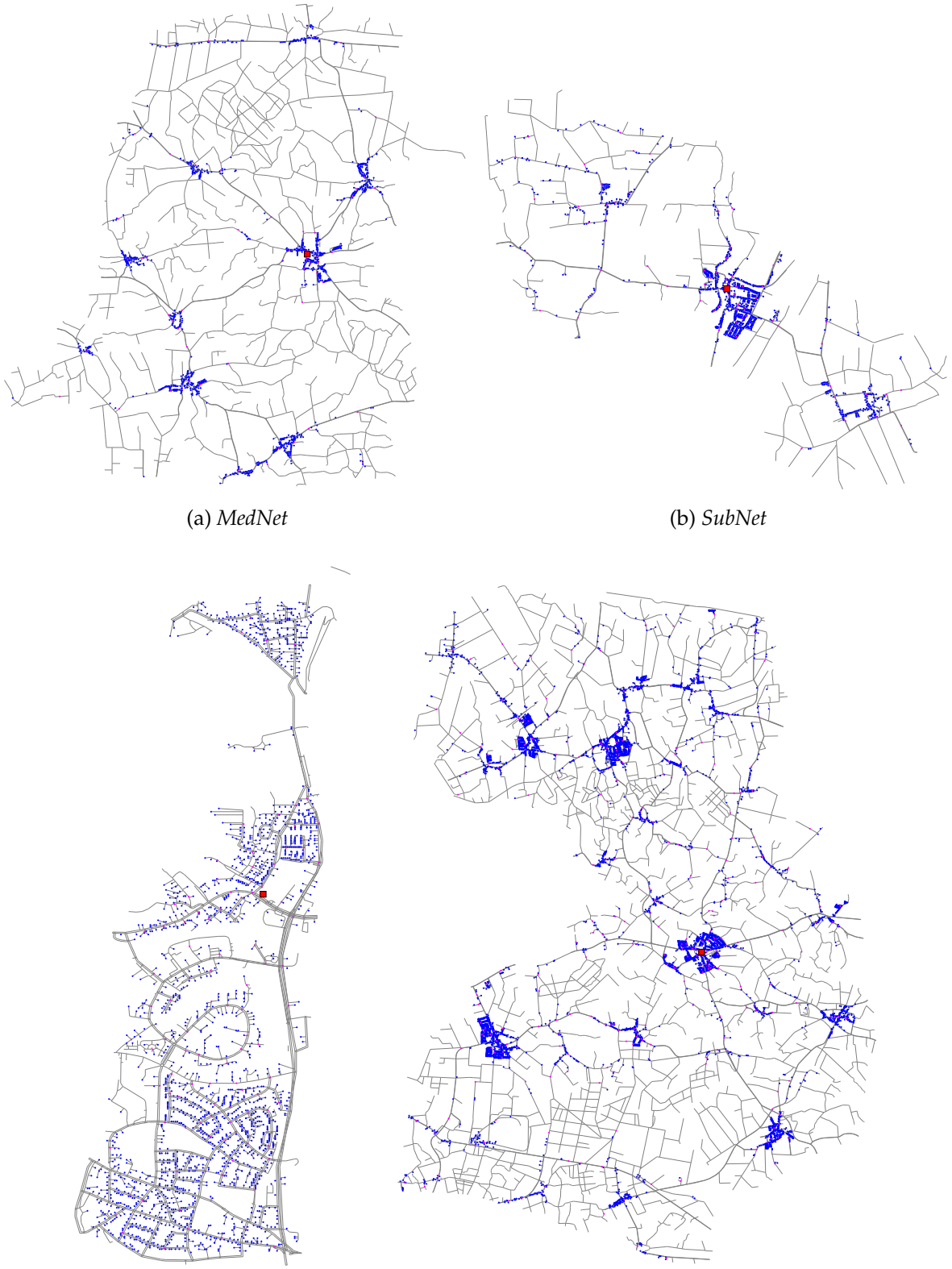
5.3.2 Parameters

With the additional refinements in this chapter, the number of global parameters needed increases. Applicable parameters from the basic model are reintroduced as parameters for the refined model in table 5.2, with the remaining parameter values once again being set to values consistent with typical values from industry and other works.

For parameters formulated with EOS, the unit cost is the baseline cost, i.e. the cost to deploy a single piece of equipment. Depending on the type of EOS cost curve, the baseline will then be adjusted as volume increases. Splitter cost c_{nt}^{SP} will be defined according to the test scenario due to the fact that both EOS and the splitter type influences the cost value. Network reach parameters are defined as specified in the ITU-T G.984 [3] standard.

5.3.3 Result interpretation

The aim of the refined model has shifted from that of the basic model, with solution accuracy and real-world correlation taking center stage. To interpret the results from



(a) *MedNet* (b) *SubNet*

(c) *CityNet* (d) *HugeNet*

Figure 5.8: GIS-mapped datasets used for the refined model

Table 5.2: Refined model global design parameters

Parameter	Symbol	Unit	Value
Costs			
Fixed OLT setup cost	C_{OLT}	Rand (R)	10,000
Baseline ONU unit cost	c_0^{ONU}	R	200
Fiber cost (CO $\rightarrow$ SP)	σ_p^{CO-SP}	R/m	100
Fiber cost (SP $\rightarrow$ ONU)	σ_p^{SP-ONU}	R/m	120
Trenching cost (CO $\rightarrow$ SP)	σ_e^{CO-SP}	R/m	300
Trenching cost (SP $\rightarrow$ ONU)	σ_e^{SP-ONU}	R/m	300
Constraints			
Max network reach	ℓ_{MAX}^{TOTAL}	km	60
Max differential distance	ℓ_{MAX}^{DIFF}	km	20
Max number of COs	κ_{CO}		3
Max ONUs per splitter	κ_{ONU}^{S}		64
Large number	Δ		10^7
Max running time		min	60

the refined model, the following evaluation points are considered:

- **Numerical correlation** - A step beyond numerical feasibility, numerical correlation refers to the numerical outputs of the model correlating with real-world expected values, with more emphasis on absolute values. Distribution of values between datasets are less important for the refined model, since the inherent properties of the datasets are not directly related as was the case with the uniformly distributed data.

- **Deployment cost** - One of the primary aims of the refined model, deployment cost reduction due to model refinements is a key point of evaluation for all test cases, especially in the case of the fiber duct sharing and splitter types refinements.

- **Topological accuracy** - Even though topological accuracy is not a primary evaluation point, as this was already established in the basic model, the PON plan still has to adhere to all topological requirements to have any practical feasibility.

- **Performance** - Since a lot of complexity is added to the refined model, perfor-

mance will undoubtedly decrease, although solution times still need to be feasible.

Now that the tests can be evaluated, the different types of tests along with their respective results will be discussed.

5.4 Results and analysis

The refined model was again implemented in C++ using the Concert extension of IBM ILOG CPLEX [13] and the Digia Qt 4.8.0 framework [52]. All tests were run on an Intel Core i7 processor running at 2.67 GHz with 16 GiB main memory and a 148 GiB page file split over two Intel Solid-State Drives. This resulted in a total of 164 GiB usable memory.

Scenarios are now given in which fiber duct sharing, coverage, splitter types and EOS are tested explicitly. All other refinements will be tested implicitly by noting their operation in the other test scenarios. Firstly, a baseline solution will be calculated for all datasets, before moving on to more detailed testing and validation. Table 5.3 shows a summary of the test scenarios in this section, along with their corresponding parameters. A bullet in this table indicate the inclusion of a specific aspect in a test scenario.

5.4.1 Baseline

Methodology

This test gives a baseline to the model solutions without fiber duct sharing, coverage, splitter types or EOS for all four datasets. Therefore, the solutions of this scenario can be compared to scenarios with these effects included. To ensure an optimal or near-optimal solution can be computed for each dataset, the time limit is extended for this test. The exact testing environment is defined in table 5.3.

Table 5.3: Refined model test environments

		Test scenario					
Parameter	Symbol	Baseline	Duct sharing	Coverage	SP types	EOS	Complete
Datasets used							
MedNet		•	• / 100 *		•	•	•
SubNet		•			•	•	•
CityNet		•		•	•	•	•
HugeNet		•					•
Refinements							
Fiber duct sharing	Paths		1 - 20				1
Non-symmetrical fiber		•	•	•	•	•	•
Multiple COs		•	•	•	•	•	•
Coverage	%			10 - 100			100
Network constraints		•	•	•	•	•	•
Splitter types					•		•
Economies of scale						•	•
Additional parameters							
Splitter base cost	c_0^{SP}	6,000	6,000	6,000	3,000	6,000	3,000
Splitter cost step	c_t^{SP}				1,000		1,000
Splitter types	N	1	1	1	4	1	4
Splitter capacities	N_{nt}^{SP}	64	64	64	8, 16, 32, 64	64	8, 16, 32, 64
Max running time		360	60	60	60	60	60

* A subset of 100 ONUs is used for preliminary tests

Results analysis

The numerical results of the baseline test are given in table 5.4. As was seen in the numerical results from the basic model, the baseline model shows cost per ONU directly correlated to the dataset's ONU density, with higher densities translating to lower cost per ONU as is expected. In contrast with the basic model though, average split ratios for the refined model are much higher, even given the drastic changes in ONU density due to clusters as seen in the data. This illustrates that the possible splitter locations are placed strategically, with no splitters being placed next to each other or far from any ONUs as is possible in randomly generated datasets.

Computational results from the baseline test show decent performance for all but one dataset, with *HugeNet* posing a particular challenge for the baseline model due to its size and complexity. Even with an extended time limit, the solver failed to converge to an optimal integer solution, staying at low optimality gaps of under 2 % from the second hour of the test right up to the six hour termination point.

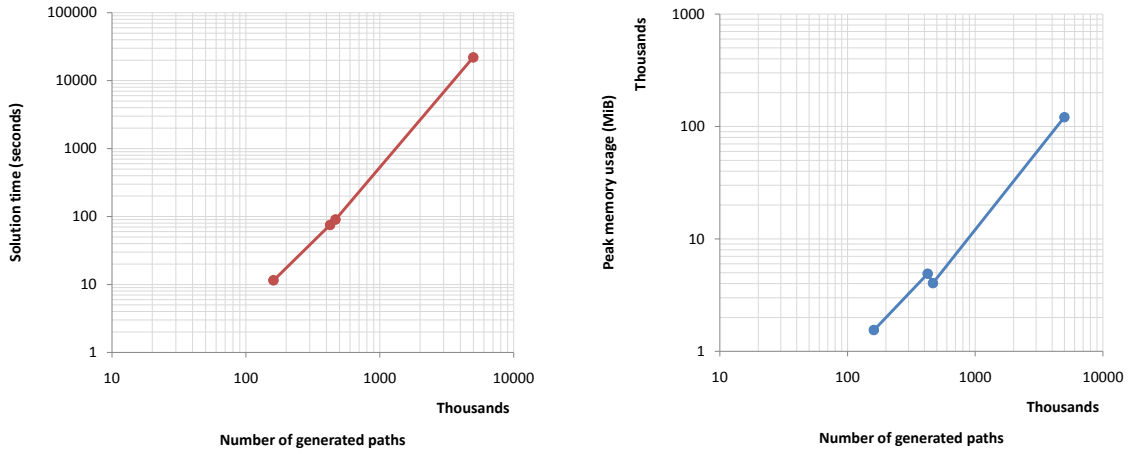
Table 5.4: Refined model results: Baseline test

		Dataset			
Result	Unit	*MedNet*	*SubNet*	*CityNet*	*HugeNet*
Numerical					
Total cost	R (mil)	161.46	124.26	121.08	848.29
Cost per ONU	R (1000)	152.90	115.16	62.06	175.81
Splitters used		35	48	80	134
Avg. split ratio		30.17	22.48	24.39	36.01
Optimality gap	%	0.0	0.0	0.0	0.79
Number of paths	1000	424.9	160.9	466.5	4,990
Computational					
Setup time	s	2.7	1.2	3.0	173.8
Time to solve	s	75.1	11.5	89.7	21,960
CPU time	s	464	57	286	21,358
Path calculation time	s	36.1	3.9	13.8	2,113
Subset generation time	s	0.1	0.1	0.2	248.8
Peak memory	MiB	4,909	1,543	4,038	121,070

It is interesting to note that a relatively good linear relation between the number of paths and the time to solve can be seen in the log-log plot in figure 5.9a, with the number of paths giving a good indication as to at least the relative complexity order of the model. Even with wildly different node densities, *MedNet* and *CityNet* showed similar complexity in the results, having a similar number of paths in their respective datasets.

Memory usage remained relatively moderate for the *MedNet*, *SubNet* and *CityNet* datasets with up to 4 GiB usage, spiking to a staggering 120 GiB for the *HugeNet* dataset. Even with the page file placed on extremely fast SSDs, it is still not nearly as fast as the 16 GiB main memory, and the time to solve this dataset will not be comparable to the other three. Looking at the log-log plot in figure 5.9b of the peak memory usage given the number of paths, a linear relation can be seen, with a line with equation $y = 0.025x - 5040$ showing an R^2 value of 0.998, where x is the number of paths and y the peak memory usage.

Preprocessing times remained acceptable, with even the 5 million shortest paths of



(a) Time to solve

(b) Peak memory usage

Figure 5.9: Log-log plots of time to solve and peak memory usage for refined model given number of paths

HugeNet being calculated in just over 35 minutes. This clearly attests to the efficiency of the multi-threaded implementation of Dijkstra's algorithm, with over 2,000 shortest paths calculated each second, given a graph with 24,000 nodes and 26,000 edges.

5.4.2 Fiber duct sharing

Methodology

The fiber duct sharing test compares the objective value of a model formulated with fiber duct sharing to the objective value of the baseline. The test therefore aims to determine the difference in total deployment cost. As a secondary aim, the test tries to determine a good value for the number of paths to generate for each commodity, where the total cost reduction is sufficient to warrant the increased complexity and much longer preprocessing times.

Firstly, a 100 ONU, 50 splitter subset of *MedNet* will be solved with the number of paths per commodity ranging between 1 and 10 to note the increased complexity and change in total cost. Even though diminishing returns are expected for larger path numbers,

especially considering the added complexity, 15 and 20 path subsets will also be solved to be sure.

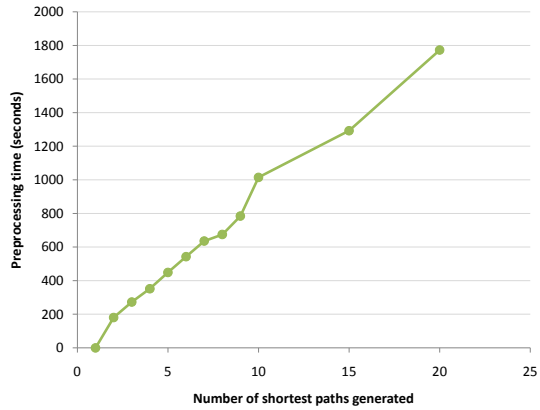
Finally, the entire *MedNet* dataset will be solved using the *best* number of paths as determined in the subset tests. This can then be compared to the baseline result to determine the change in deployment cost. As indicated in table 5.3, a single splitter type with a split ratio of 1:64 will be used for the test, along with full coverage and no economies of scale.

Results analysis

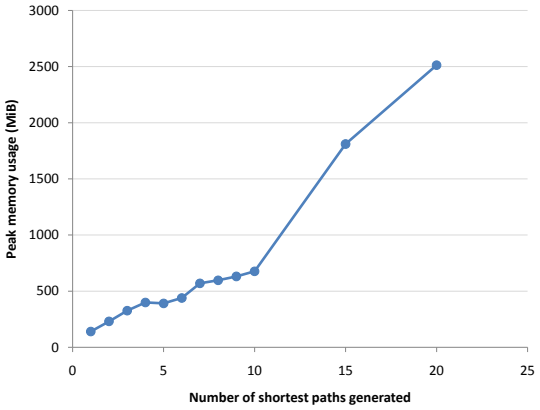
The results of the fiber duct sharing test are illustrated in figure 5.10. As expected, an increase in the number of paths not only results in a linear increase in preprocessing time, but also a seemingly exponential increase in model solution time. Conversely, the incremental reduction in total deployment cost is comparatively small, peaking at 5.33 % using 20 shortest paths. Given the complexity increase each additional path introduces to the model, with a 167 fold increase in computation time when moving from 1 to 20 paths, it would suggest that the previous projection of diminishing returns holds.

Looking at preprocessing times, the difference is especially noticeable when looking at the switch between algorithms, with a single shortest path calculated in 0.3 seconds with Dijkstra's algorithm while calculating two paths with Yen's algorithm took more than three minutes. Even though this is still feasible, it will manifest in a major increase in processing times for larger datasets, as seen when solving the full *MedNet* dataset.

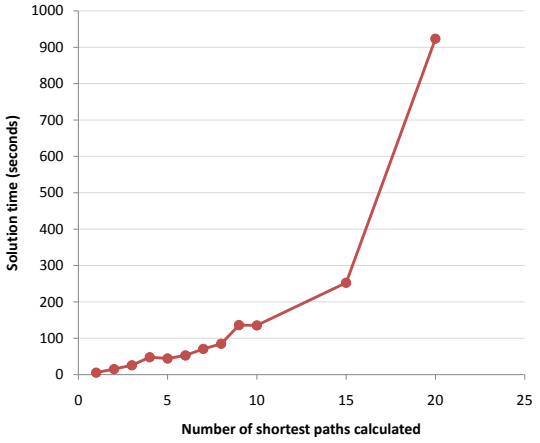
Even though complexity increases drastically with additional paths, peak memory usage shows less of a spike, with a linear increase while path storage is still the largest contributor scaling up when solver memory usage becomes the dominant factor. It must be noted though that at 20 paths, even this small 100 ONU subset of *MedNet* used close to 2.5 GiB, which will once again ramp up with the full dataset and become infeasible.



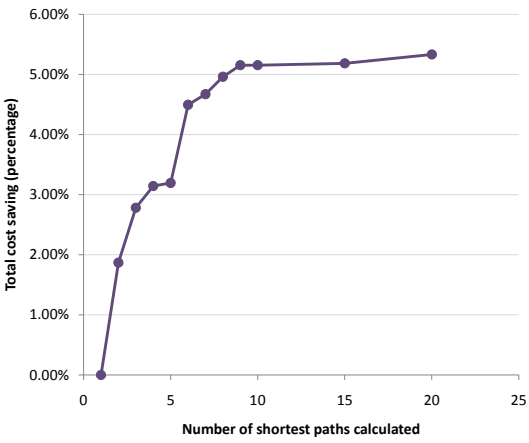
(a) Preprocessing time



(b) Peak memory usage



(c) Time to solve



(d) Percentage cost saving

Figure 5.10: Plots of refined model performance for different number of shortest paths

Finally, when looking at the full *MedNet* dataset's deployment cost and solution time results as illustrated in figure 5.11, it is clear that the introduction of fiber duct sharing, even using only the shortest path, results in a large deployment cost reduction. In this test, the reduction is at least 42.6 %. Unfortunately, the time to solve the model increases drastically, with the single shortest path model exiting after an hour with an optimality gap of 24.72 %.

The test was also started with more shortest paths, but preprocessing was deemed infeasible with even the 2 path model's time being estimated at 27 hours, increasing linearly with k. Since the introduction of another path only resulted in a deployment cost reduction of 1.87 % in the subset tests, the additional complexity is not deemed

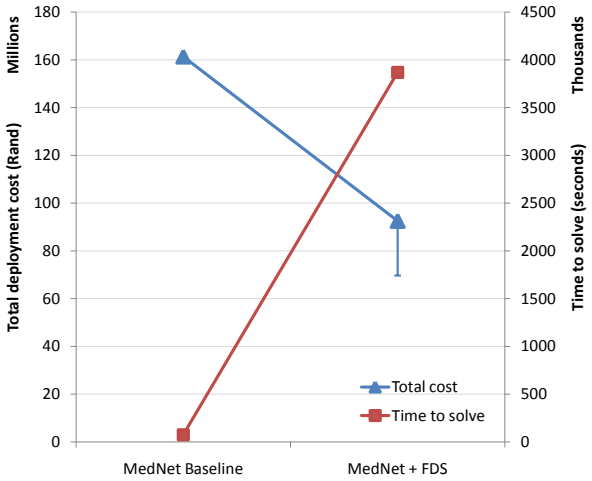


Figure 5.11: Total deployment cost and time to solve for the refined model with and without fiber duct sharing - 1 path

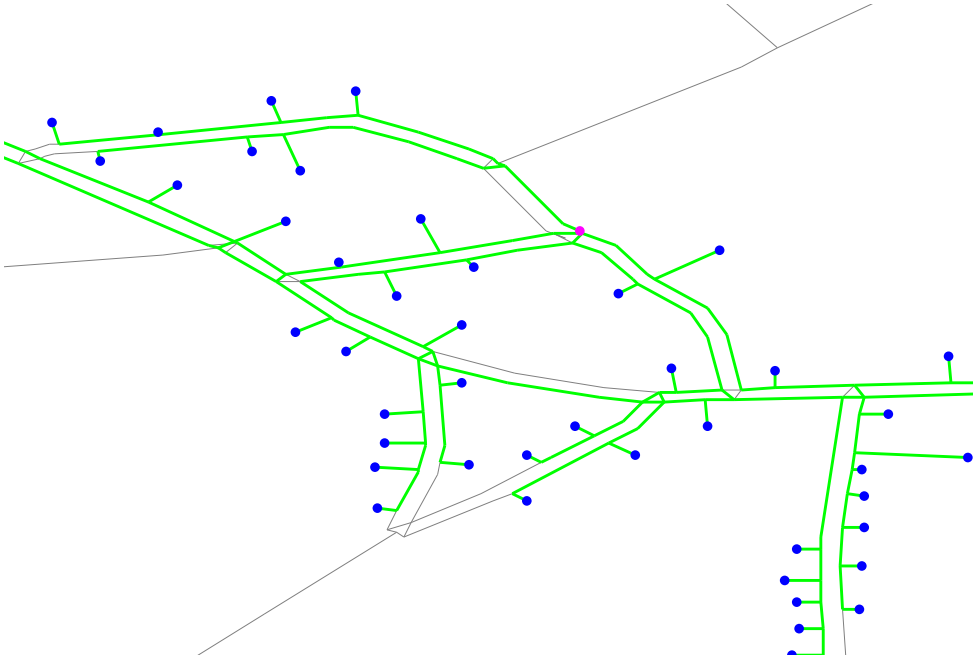
worthwhile for these preliminary tests. When solving a real-world model in practice however, an additional day or week of processing time may be tolerated, since even a small percentage of the overall cost would result in a significant monetary amount.

To illustrate the topological differences between the number of paths, two extracts of the PON plan are shown in figure 5.12, the first using only the shortest path and the second using 20 shortest paths. It is clear that the increased number of paths results in a more intuitive plan, where a single trench would be used instead of multiple trenches in close proximity. Another noticeable aspect of the plots is that even though the deployment cost difference between using a single path versus using 20 paths is relatively small, the resulting plan at this small scale differs quite dramatically, indicating that a large number of configurations exist that will result in a similar deployment cost.

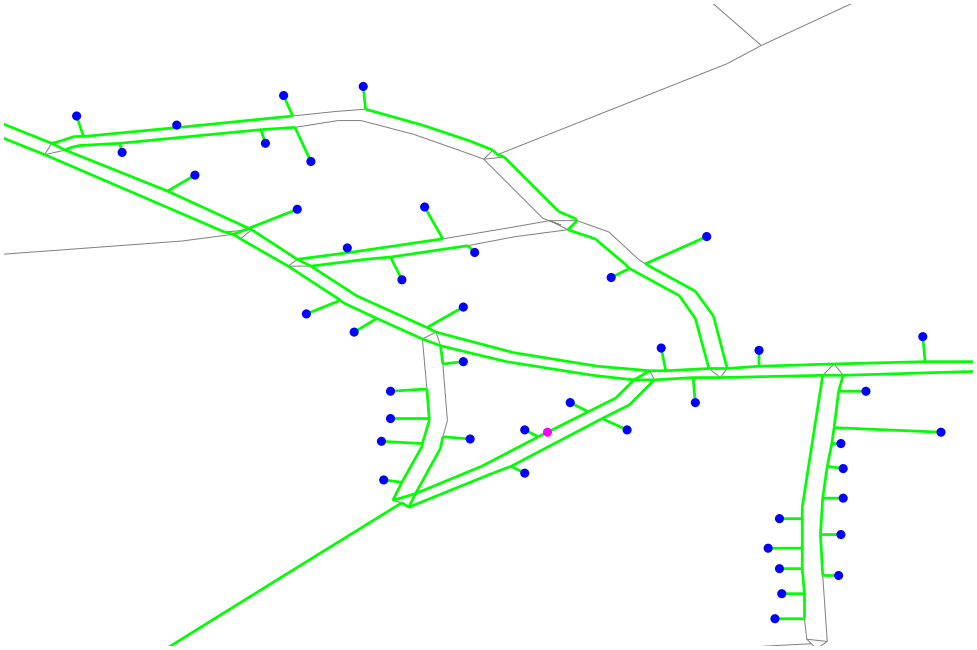
5.4.3 Coverage

Methodology

To determine correct operation of the coverage refinement, the cost per ONU will be compared for different values of ω by solving the *CityNet* dataset to optimality. Ten



(a) Single shortest path



(b) 20 shortest paths

Figure 5.12: Difference in refined model topological output when increasing number of shortest paths

tests are run, with coverage values ranges from 10 % to 100 % in 10 % intervals. This will allow a decent resolution to compare the results.

As indicated in table 5.3, the coverage tests are run with a single splitter type of 1:64 and no fiber duct sharing. No economies of scale effects are included in the model, effectively making link costs linear with distance. When EOS effects are included, it introduces non-linear effects in the link costs, usually producing a minimum cost per ONU at a coverage percentage of 40 % to 60 % [60]. Therefore, the test is expected to produce costs per ONU that do not necessarily have a minimum for a coverage value larger than the minimum tested.

Results analysis

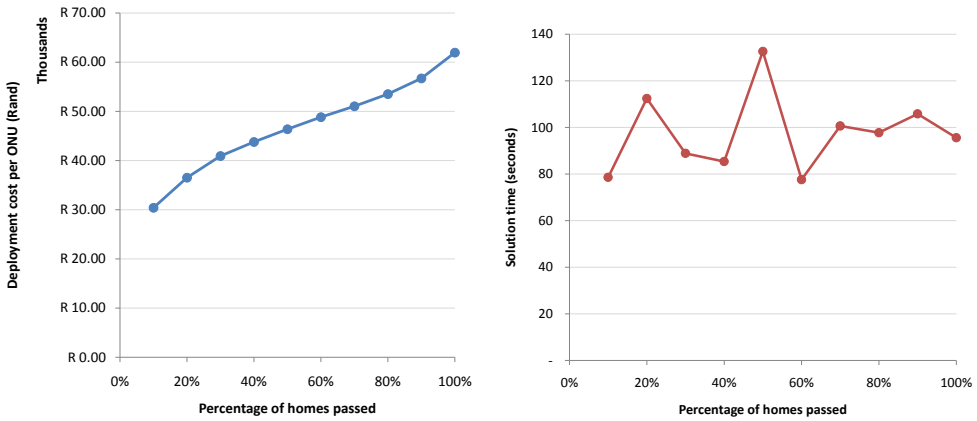
The coverage test results are shown in table 5.5. As this test solves the same dataset, while connecting the specified percentage of possible ONUs for minimum total cost, the primary result of concern is the cost per ONU. This gives a relative measure for SPs in the form of CAPEX per ONU. Minimizing this value will result in the highest ROI for an SP. As expected however, the absence of economies of scale results in a linear graph, with a minimum cost per ONU at the minimum coverage of 10 %.

Table 5.5: Refined model results: Coverage test

		Coverage									
Result	Unit	10 %	20 %	30 %	40 %	50 %	60 %	70 %	80 %	90 %	100 %
Numerical											
Total cost	R (mil)	5.9	14.2	23.9	34.2	45.2	57.1	69.7	83.4	99.5	120.8
Cost per ONU	R (1000)	30.4	36.5	40.9	43.8	46.4	48.8	51.0	53.5	56.7	61.9
Splitters used		13	22	27	34	40	46	54	60	69	80
Avg. split ratio		15.00	17.73	21.67	22.94	24.38	25.43	25.28	26.00	25.43	24.39
Optimality gap	%	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Number of paths	1000	46.8	93.4	140.0	186.7	233.3	279.8	326.5	373.0	419.7	466.5
Computational											
Setup time	s	3.5	3.1	3.1	3.1	3.1	3.1	3.1	3.1	3.1	3.1
Time to solve	s	78.6	112.4	88.9	85.4	132.6	77.6	100.6	97.8	105.8	95.6
CPU time	s	242	282	192	163	423	154	218	194	220	191
Peak memory	MiB	4,511	5,023	4,567	4,492	5,112	4,027	4,801	4,076	4,072	4,036

Of note however is the two extremes at 10 % and 100 % coverage in figure 5.13a. At lower coverage values, a plan can be constructed that contains a close proximity of clusters, decreasing the number of long distance fiber links to other clusters. This then translates to a lower overall cost per ONU as seen in the figure. Conversely, the fully

connected plan must connect all outlying clusters, even if they consist of only a few ONUs, increasing the overall cost per ONU.



(a) Deployment cost per ONU (b) Solution time

Figure 5.13: Plots of refined model performance with regards to coverage

Since coverage is specified as a parameter, it should not increase the relative complexity of the model as the value is varied, and therefore both the computation time and memory usage should remain constant. Solution time, as plotted in figure 5.13b, shows a constant relationship, albeit with large variances arising from the form of the data and different cut and presolve techniques used by the solver. With a sample mean of 97.5 seconds and a standard deviation of 16.7 seconds, this closely resembles the baseline results of the *CityNet* dataset, which had a solution time of 89.7 seconds. This indicates that as expected the addition of coverage into the model does not significantly add to the model complexity.

Finally, the topological output of the model for different coverage values can be analysed. Results with coverages of 10 %, 50 % and 100 % can be seen in figure 5.14. As previously mentioned, it is clear that the lower coverage plans start with connecting the highest density clusters close to the CO. With an increase in coverage, additional ONUs in already connected clusters or high density clusters are connected first, before moving to outlying or sparsely distributed clusters.

(a) 10 % coverage (b) 50 % coverage (c) 100 % coverage

Figure 5.14: Topological output of *CityNet* for different coverage values

5.4.4 Splitter types

Methodology

The correct operation of splitter types will be tested using a comparison between the baseline results and the results with multiple splitter types incorporated into the model. Since the model is solved by optimizing different splitter types, a direct verification cannot be done. Splitters may be allocated differently, or multiple smaller splitters may replace a larger one in the baseline solution if fiber cost can be reduced sufficiently.

As indicated in figure 5.3, this test will comprise the solving of all but the *HugeNet* dataset with multiple splitter types, excluding fiber duct sharing, coverage and EOS. Table 5.6 shows the different splitters used as well as their respective unit costs. The baseline results will use a single splitter type with a split ratio of 1:64 at a unit cost of R 6,000, as specified in the baseline test parameters.

Table 5.6: Refined model tested splitter types

Splitter type	Split ratio	Unit cost (Rand)
Type A	1:8	3,000
Type B	1:16	4,000
Type C	1:32	5,000
Type D	1:64	6,000

Results analysis

The numerical results of the splitter type test are shown in table 5.7. Comparing these results with that of the baseline results in table 5.4, there are a number of observations that can be made. Firstly, the total deployment cost is reduced as expected, although the difference is almost insignificant, with a 0.05 %, 0.02 % and 0.09 % reduction for the *MedNet*, *Subnet* and *CityNet* datasets respectively.

Table 5.7: Refined model results: Splitter type test

		Dataset		
Result	Unit	*MedNet*	*SubNet*	*CityNet*
Numerical				
Total cost	R (mil)	161.43	124.20	120.98
Cost per ONU	R (1000)	152.86	115.10	62.00
Splitters used		35	48	81
Avg. split ratio		30.17	22.48	24.09
Optimality gap	%	0.0	0.0	0.0
Splitters				
Type A		3	7	16
Type B		6	12	17
Type C		13	19	27
Type D		13	10	21
Computational				
Setup time	s	3.0	1.0	3.2
Time to solve	s	93.4	13.2	123.1
CPU time	s	212	31	252
Peak memory	MiB	4,981	1,550	4,166

Whereas the number of splitters used remained unchanged for the *MedNet* and *SubNet*

datasets, the *CityNet* dataset solution shows an additional splitter being used. This is likely due to the disintegration of the coverage area of a large splitter into two distinct clusters, each with its own splitter of sufficient size. Furthermore, the number of small splitters of Type A to C suggests that the inclusion of splitter types alleviated a large number of cases where an allocated splitter had excessive capacity.

Computational results in comparison with the baseline results suggest that the inclusion of splitter types increased the complexity of the model, with solution times increasing by a significant margin. The *SubNet* dataset solution exhibited an increase of 15.39 %, the least of the three datasets tested, while the *MedNet* and *CityNet* datasets increased by 24.35 % and 36.97 % respectively. Fortunately, this should be a relatively fixed increase, since the number of splitter types available in practical applications should be of the same order of magnitude as the four tested here. Peak memory usage remained largely unchanged with the addition of splitter types, with the relative disparity remaining below 3 %.

5.4.5 Economies of scale

Methodology

Economies of scale on the cost of products can be implemented in two different ways. The first way is to structure discounts on any additional units purchased, where a discounted price only applies to the number of units that exceed the normal volume bracket. For example, if a buyer gets a 10 % discount on all units over 100 and he then purchases 101, only one unit will receive the discount. The second and more common way is to structure discounts on all units purchased. If the same buyer in the previous example now purchases 101 units, he will receive a 10 % on all 101 units.

Therefore, to test economies of scale effects in the model, a test is run for each type of EOS structure. EOS is only implemented on ONU and splitter unit cost, as EOS on fiber will distort the solution, resulting in a plan that may use a longer path to shift the

fiber length into the discounted region. Fiber economies of scale can be implemented externally to the model, where the SP can decide if it is worthwhile to buy additional fiber at a discounted rate without wasting fiber by deploying excessive lengths.

The test involves solving the *MedNet*, *SubNet* and *CityNet* datasets, using the unit prices as indicated in table 5.8. The resulting EOS total deployment cost curves for both discount structures are illustrated in figure 5.15. The unit price values are calculated using a simple linear decrease in price, with a R 20,00 reduction for every 400 ONUs and a R 500,00 reduction for every 20 splitters. As before, fiber duct sharing and coverage are excluded from the test and a single splitter type with a 1:64 split ratio is used.

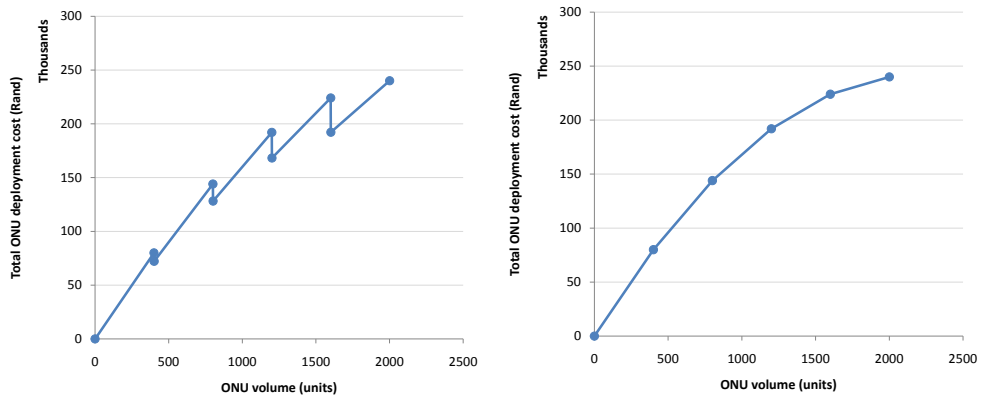
Table 5.8: Refined model pricing with economies of scale

Volume	Unit cost (Rand)
ONU pricing	
0 - 400	200
401 - 800	180
801 - 1200	160
1201 - 1600	140
1601+	120
Splitter pricing	
0 - 20	6,000
21 - 40	5,500
41 - 60	5,000
61 - 80	4,500
81+	4,000

Results analysis

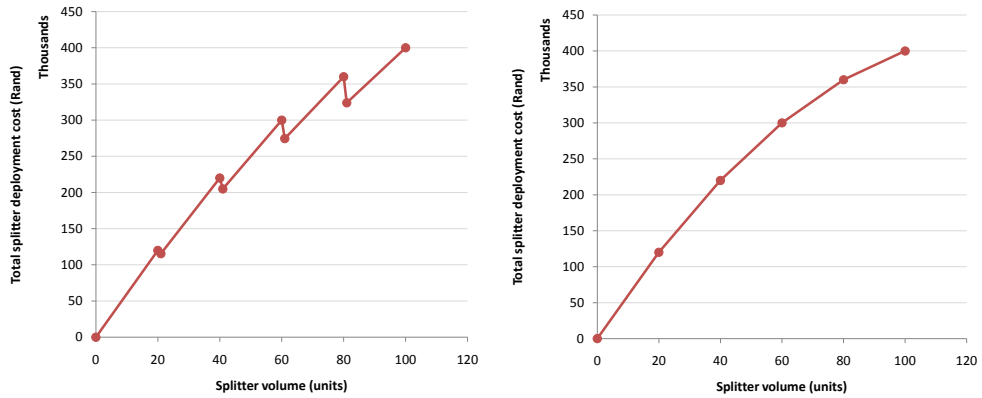
The numerical and computational results of the total discount structure are given in table 5.9. It shows a decrease in deployment cost of 0.04 %, 0.07 % and 0.26 % for the *MedNet*, *SubNet* and *CityNet* respectively as compared to the baseline results. This is as expected, as both ONUs and splitters are now cheaper due to economies of scale effects.

The *CityNet* dataset results shows an additional splitter being used, likely due to the



(a) ONU - total discount

(b) ONU - incremental discount



(c) SP - total discount

(d) SP - incremental discount

Figure 5.15: Economies of scale total deployment graphs for ONUs and splitters

fact that when moving from 80 to 81 splitters, each splitter is R 500,00 cheaper. This single extra splitter results in a total splitter deployment cost saving of R 36,000, covering the additional cost to deploy it as well as its connecting fiber.

The numerical and computational results of the incremental discount structure are given in table 5.10. The total deployment cost for the incremental discount scenario is reduced as expected, with the reduction compared to the baseline results amounting to 0.03 %, 0.06 % and 0.22 % for the *MedNet*, *SubNet* and *CityNet* datasets respectively. These are lower than the total discount structure results since a higher discount is achieved when it applies to all units.

Computational results for both scenarios are similar to the baseline results, with a

Table 5.9: Refined model results: Economies of scale test - Total discount

		Dataset		
Result	Unit	*MedNet*	*SubNet*	*CityNet*
Numerical				
Total cost	R (mil)	161.40	124.17	120.77
Cost per ONU	R (1000)	152.84	115.08	61.90
Splitters used		35	48	81
Avg. split ratio		30.17	22.48	24.09
Optimality gap	%	0.0	0.0	0.0
Computational				
Setup time	s	2.8	1.1	3.7
Time to solve	s	80.0	17.1	94.8
CPU time	s	188	36	195
Peak memory	MiB	4,909	1,543	4,039

Table 5.10: Refined model results: Economies of scale test - Incremental discount

		Dataset		
Result	Unit	*MedNet*	*SubNet*	*CityNet*
Numerical				
Total cost	R (mil)	161.41	124.19	120.81
Cost per ONU	R (1000)	152.85	115.09	61.92
Splitters used		35	48	80
Avg. split ratio		30.17	22.48	24.09
Optimality gap	%	0.0	0.0	0.0
Computational				
Setup time	s	3.0	1.1	3.1
Time to solve	s	72.7	17.3	128.6
CPU time	s	165	38	248
Peak memory	MiB	4,825	1,680	4,476

slight increase in solution time for all but the *MedNet* dataset in the incremental discount scenario. This would suggest that complexity is increased when incorporating economies of scale, although disparities remained below 40 %. Once again, the actual complexity increase is difficult to pin down exactly, as the solution time measurements have a large error margin due to background activity and specific solver decisions.

5.4.6 Complete model

Methodology

Now that the individual refinements to the model have been tested, the complete model, incorporating all the refinements, can be tested. This ensures all the refinements integrate correctly and serves as a basis for comparison with the established baseline. All four datasets are tested as specified in table 5.2. Using the insight gained from the fiber duct sharing test, the model will be solved using only the shortest path in an effort to produce feasible solutions given the resource constraints. Coverage is locked at 100 % to allow comparison with the baseline.

Economies of scale effects on ONU unit cost will be conducted as specified in table 5.8. Splitter types and EOS on splitters will be integrated as indicated in table 5.11. The same linear reduction will be applied to the other splitter types as was done in the previous section, with Type A, B and C reduced by R 200,00, R 300,00 and R 400,00 at each step respectively. The economies of scale is structured as total discounts, with each bracket implying the cost per unit for all units purchased.

Table 5.11: Complete refined model splitter pricing and types

	Unit cost of splitter type (Rand)			
Volume	Type A	Type B	Type C	Type D
0 - 20	3,000	4,000	5,000	6,000
21 - 40	2,800	3,700	4,600	5,500
41 - 60	2,600	3,400	4,200	5,000
61 - 80	2,400	3,100	3,800	4,500
81+	2,200	2,800	3,400	4,000

Results analysis

Unfortunately, the *HugeNet* dataset could not be solved with the refined model, with CPLEX giving an *out of memory* error before reaching a feasible solution. The numerical and computational results for the other three datasets are given in table 5.12. While

SubNet shows a small optimality gap of only 2.52 % at the one hour time limit, the other two datasets shows less impressive results, with *MedNet* and *CityNet* only managing optimality gaps of 25.98 % and 15.54 % respectively. Even taking this into consideration, the total deployment cost is reduced considerably in comparison with the baseline results, with an average decrease of 47.8 % across all datasets, as illustrated in figure 5.16.

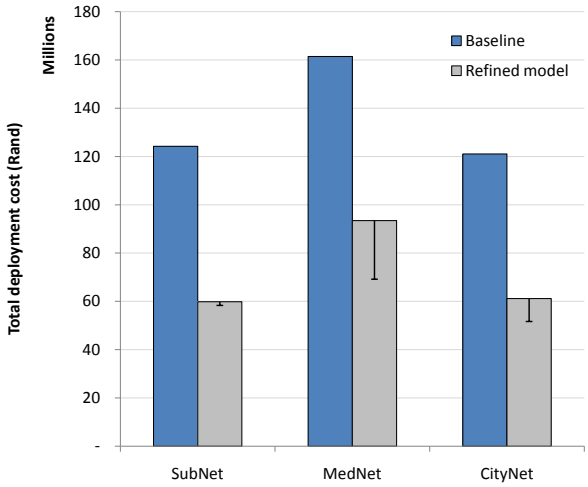


Figure 5.16: Total deployment cost of refined and baseline model

Interestingly, the *MedNet* and *CityNet* datasets show the deployment of an additional 23 and 14 splitters respectively as compared to the baseline results. This is a significant increase but not unexpected at such large optimality gaps, with the optimal solution likely containing less splitters and a larger average split ratio. The influence of splitter types and economies of scale can also be responsible for the increase in the number of deployed splitters, although the *SubNet* results suggest this is unlikely.

Computational timing results are largely inconclusive due to the time limit, although it is clear that the time to solve the complete model is significantly higher than the baseline. Lower solution time limits for *SubNet* suggest an increase of at least 315 fold, while *MedNet* and *CityNet* shows at least 50 and 41 fold increases respectively. However, due to the high optimality gaps of the larger datasets it is likely that these estimates are *very* conservative. Conversely, although the peak memory requirement for the three solvable datasets was high, it remained feasible, with a peak of 20 GiB

Table 5.12: Refined model results: Complete test

		Dataset		
Result	Unit	*MedNet*	*SubNet*	*CityNet*
Numerical				
Total cost	R (mil)	93.46	59.83	61.16
Cost per ONU	R (1000)	88.50	55.45	31.35
Splitters used		58	48	94
Avg. split ratio		18.21	22.48	20.76
Optimality gap	%	25.98	2.52	15.54
Splitters				
Type A		1	2	12
Type B		16	14	1
Type C		20	21	0
Type D		21	11	81
Computational				
Setup time	s	57.0	14.3	35.2
Time to solve	s	3,761	3,627	3,680
CPU time	s	7,023	9,273	7,131
Peak memory	MiB	20,022	11,283	12,943

for the *MedNet* dataset. Finally, the topological output of the model given the three solvable datasets is shown in figure 5.17.

5.5 Conclusion

In this chapter, a number of refinements were made to the basic model introduced in the previous chapter, along with detailed discussions on the mathematical design of each. The first of the refinements, the use of GIS datasets, allowed for a much more practical approach to real-world scenarios, while smaller refinements such as non-symmetrical fiber costs, the support for multiple central offices and network constraints improved the general capabilities of the model.

Arguably the largest improvement, fiber duct sharing showed a significant decrease in

(a) *MedNet*

(b) *CityNet*

(c) *SubNet*

Figure 5.17: Complete refined model topological output

deployment cost, with a 42.6 % reduction as tested with the *MedNet* dataset. Economies of scale on ONU and splitter unit costs as well as the inclusion of different splitter types further reduced the deployment cost by implementing real-world cost reduction effects into the model. The complete model showed an average total deployment cost reduction of 47.8 % across the datasets tested, showing promise as a viable real-world tool. Finally, the incorporation of coverage into the model allows SPs to maximize their ROI by minimizing the CAPEX per ONU.

All these refinements introduced additional complexity into the model, with the complete model having close to a million constraints and half a million variables for all datasets tested. This increase resulted in a significant increase in solution time. Whereas the baseline model for the *SubNet* dataset was solved in 11.5 seconds, the refined model was still unsolved after an hour.

In contrast with the basic model of the previous chapter, the refined model is much more accurate while being close to infeasible to solve. To find a suitable compromise between these two extremes, the next chapter will aim to improve the solution times to feasibility while striving to be as accurate as possible.