Graph Algorithms and Real-Time Telemetry for Intelligent Plant Operations

A Technical Paper prepared for SCTE by

Bob Lutz
Senior Engineer
Comcast
1800 Arch St, Philadelphia, PA
Bob_Lutz@comcast.com

Matthew Stehman
Principal Engineer
Comcast
1800 Arch St, Philadelphia, PA
Matthew_Stehman@comcast.com

Table of Contents

List of Figures

List of Tables

1. Introduction

Real-time telemetry analysis has always been a core requirement for cable plant operators to detect customer-impacting events and dispatch the appropriate field teams. In the age of 10G networks, the cable plant's margin for error is slimmer than ever to offer state-of-the-art technology to potential subscribers. Minimizing the time field technicians need to troubleshoot issues will be key to maintaining a high cadence for 10G node deployments and conversions. To this end, telemetry-based alerts should not only identify problems in the plant but should recommend potential solutions as well.

We present an approach for combining node-level telemetry data and graph algorithms to help technicians resolve plant issues more efficiently and reduce mean time to repair. This approach has been successfully applied to use cases on the road to 10G, including distributed access architecture (DAA)-based 2G service deployments, streamlining demand maintenance with reduced truck rolls, and improving proactive maintenance by detecting likely network impairments. The algorithms involved have been field tested, incorporating technician feedback, and are now being integrated into production operations. This paper will highlight the importance and impact of combining network telemetry with plant topology to support the continued rollout of 10G.

2. Problem Statement and Use Cases

For multiple systems operators (MSOs), processes to efficiently identify and isolate network impairments are key to running a high-performance network and ensuring that customers receive the level of service they expect. Today's 10G networks can produce vast amounts of real-time telemetry from customer premise equipment (CPE). This paper will discuss approaches for combining CPE telemetry with a graph database of the hybrid fiber-coaxial (HFC) network to identify and resolve multi-home network impairments.

When multiple customer devices on a node show a degraded level of service, a decision needs to be made on what type of resource is needed to resolve the issue. Some scenarios require a network technician to address impairments in the HFC plant, while others require an in-home technician for issues inside the customer premise. From an operations standpoint, each ticket generated for field teams should ideally address as many customer issues as possible. This will reduce the overall number of jobs while maintaining or even increasing the number of customer issues resolved. Thus it is critical to be able to identify network impairments that are impacting multiple customers or homes at once. The following sections will discuss use cases for multi-home network impairment detection algorithms that are being used daily as the development of the 10G network continues.

2.1. Legacy Plant Equipment Detection and Isolation

The requirements of the network are changing as analog architectures are converted to digital on the road to 10G. As laid out in Harb et al. (2023) and Sundaresan (2022), the path to 10G requires potentially redesigning the spectrum and channel layouts to increase speeds. If legacy analog components in the HFC plant are not designed compatibly with the updated spectrum plans, they can cause issues with CPE bonding on these new channels.

Two areas of the spectrum that demand particular attention are:

1. 45–200 MHz with transitions from sub-split to mid/high-split designs and
2. > 750 MHz with the expansion of the existing downstream frequencies to include more channels.

In the first case, upstream frequencies are expanding into what were previously downstream frequencies. If certain components of the legacy network attenuate signals in this frequency range, customer devices will be unable to bond on these new channels when the digital network is turned up. In the second case, physical limitations of network equipment, such as amplifiers causing signal power roll-off, can prevent signals at high frequencies from reaching customers.

Typically, field teams perform a network sweep for incompatible legacy equipment when preparing for digital conversion. However, some network elements can be missed—especially if they are undocumented. If this happens, all customers downstream of incompatible network equipment will be unable to bond on any channels in the new frequency ranges. These bonding issues can occur immediately after the digital cutover or introduction of new channels.

In Section 4.1, we present an approach to identifying these issues and directing field teams to specific network elements of interest. This ability to isolate incompatible network devices enables efficient resolutions that minimize negative customer impact.

2.2. Degraded Modulation Error Ratio

Customers might continue to experience degraded signal quality, meaning packet loss and underperforming speeds, even after the HFC plant is upgraded to 10G-compatible equipment. To identify customers with degraded signal quality, we can look for CPE with low Modulation Error Ratio (MER) on certain channels. Figure 1 shows example device populations with high and low MERs for downstream single-carrier quadrature amplitude modulation (SC-QAM) channels.

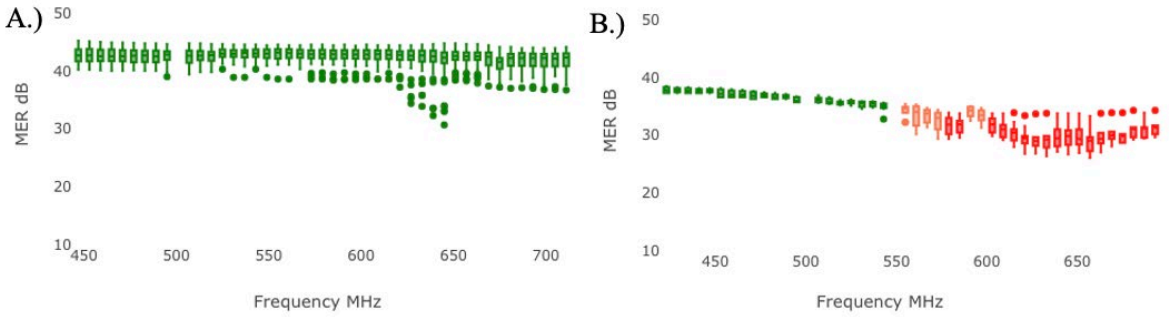

Figure 1 – Downstream SC-QAM MER distributions. A.) Majority of devices with good MER B.) Majority of devices with low MER

As in the previous use case, our goal is to identify impairments affecting multiple homes and direct field teams to root causes of the impairments. One challenge in this use case is that a radio frequency (RF) antenna in the vicinity of a node can cause CPE in separate legs of the plant topology to share similar MER characteristics. If this happens, a collection of in-home issues degrading MER, such as loose connections allowing ingress, can be mistaken for a shared multi-home issue when viewing telemetry alone. Another challenge is that devices only report MER values for channels on which they are bonded, but the CMTS can allow different CPE to bond on different channels. Thus, even if multiple devices are impacted by a single network impairment, their MER values might be incomparable if the devices are bonded on different channels.

In Section 4.2, we present a hybrid statistical and graph-based approach that overcomes these challenges by combining MER data with the plant topology. As in the previous use case, our algorithm identifies a particular network element as the likely root cause of each detected impairment.

2.3. Downstream Full Band Spectrum Clustering

Full band capture (FBC) data can reveal network impairments even when channel bonding and MER values are nominal. As discussed in Dugan et al. (2022), certain patterns in the FBC waveform indicate known network impairment types. Some of these patterns are illustrated in Figure 2 for FBC of downstream receive power at the modem.

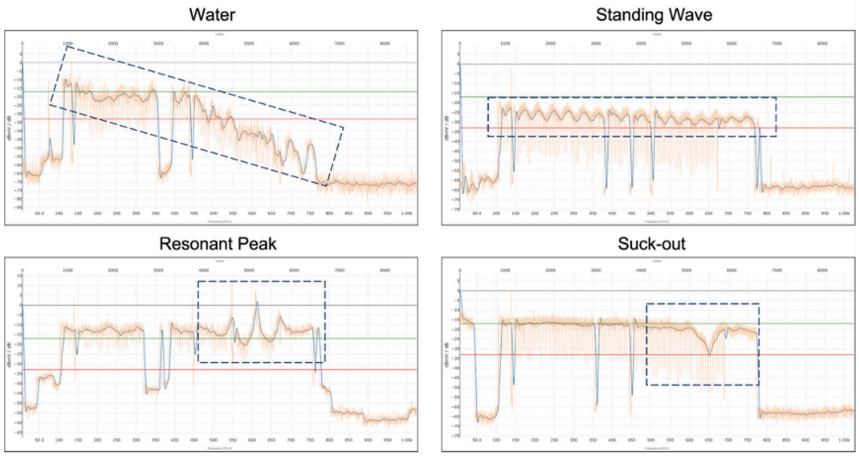

Figure 2 – Common impairment patterns on full band capture of downstream receive power at the modem

In this use case, output from pre-existing classification and clustering models processing FBC data is combined with a graph topology view of the network to identify network elements as root causes of multi-home impairments. These root cause elements are combined with results from the previous use case to find correlation between MER impairments and spectrum impairments, bundling demand maintenance (DM) events with proactive maintenance (PM) events. Our approach to this problem is presented in Section 4.3.

3. Setup

3.1. Network As a Graph

Traditional approaches to batch telemetry analysis are generally either implicit (e.g. inferring relationships between network elements based on latitude/longitude) or manual (e.g. performing root cause analysis by visual inspection of plant maps). Graph-structured data helps us outperform these baseline approaches by codifying explicit connections between network elements and allowing for full automation.

The algorithms in this paper operate on data from routing of cable infrastructure (ROCI), a graph database representing the access network. Vertices in the database represent the logical and physical entities that make up the network, from the cable modem termination system (CMTS) down to CPE. Edges represent either physical connections (e.g. by coaxial cable) or logical relationships (e.g. customer account to street address). Vertices and edges can have attributes representing properties of the given network element or

relationship, such as IP address, cable length, or latitude/longitude. To support real-time analysis, the data in ROCI is refreshed automatically as the topology of the access network evolves, by e.g. the addition of new customers, the conversion of analog nodes to digital, or the redrawing of plant maps. For more detail on ROCI, see Narayanaswamy et al. (2021).

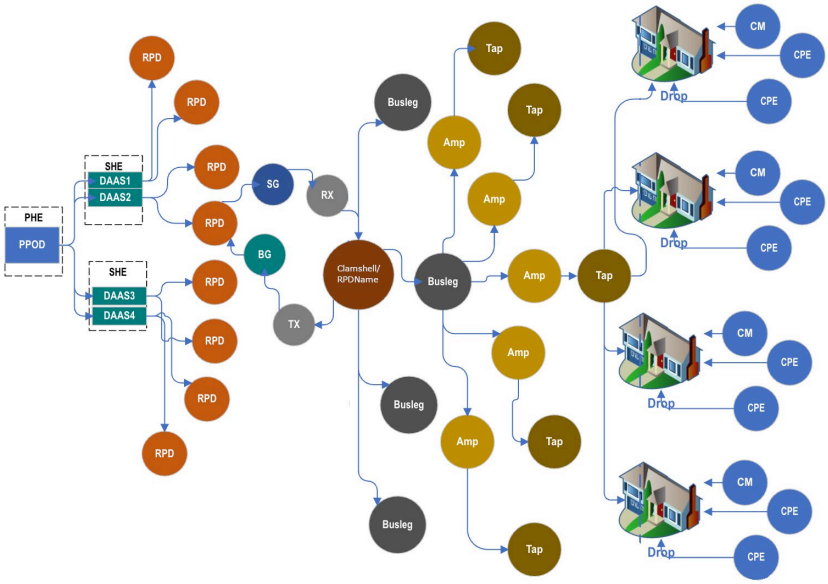

Figure 3 – Graph database schema example for virtual CMTS

3.2. Telemetry Summary

Addressing the use cases in Section 2 requires up-to-date telemetry from millions of customer devices. Fortunately, DOCSIS management information bases (MIBs) allow vast amounts of telemetry to be polled from CPE in near-real time. We briefly summarize the telemetry relevant to our use cases.

Most CPE have onboard spectrum analyzers and support the ability to report a FBC of downstream receive power across the entire frequency range. This view of receive power versus frequency gives a detailed view of the signals reaching the CPE and can be processed in a variety of ways based on project needs. Frequency-specific variations in the FBC indicate impairments related to amplification and attenuation of the signal power levels as the signals traverse the network. The FBC data is used heavily to detect HFC plant-related impairments as well as determining if certain frequencies can reach given CPE.

Even if the CPE receives signal power at a sufficiently high level, it may not be able to decode the DOCSIS code words if the signal quality is poor. In these cases, the CPE will drop packets, causing slower speeds and degraded customer experience. Channel-level MER can be used to identify these types of impairments, where the signal level is high, but noise corrupts the signal before it reaches the CPE. Processing MER data at the channel level is key, as it allows for clustering of devices based on frequency-specific patterns. This is not possible with aggregated MER values.

DOCSIS overcomes some amount of noise in a signal by including redundancy in its code words to help account for uncertainty in the packet. However, if the MER is too low and packets are lost, then the noise will impact the customer. Packet error rate (PER) can be used to measure customer experience in this situation. As will be seen in Section 5, MER and PER can be used together to efficiently identify customer-impacting, multi-home network impairments.

4. Algorithms

We present a suite of algorithms that address the use cases from Section 2 by combining device telemetry with the network topology to identify multi-home network impairments. Broadly speaking, our goal is to group together CPE or homes whose degraded service is likely due to the same underlying issue and identify the network element that is the most likely root cause.

We first establish some terminology. A *path* in a graph is a sequence $(v_0, v_1, \dots, v_n)$ of vertices such that there is an edge between v_i and v_{i+1} for all i. A graph is a *tree* if, for each pair of vertices u and v, there is exactly one path from u to v. The *leaves* of a tree are the vertices incident to exactly one edge. When designing algorithms for the access network, the key data structure is a *rooted tree*, i.e. a tree in which a single vertex is designated the *root*. In this section, the root of our graphs is always the RF node, and the leaves are CPE. A *crawl* in a rooted tree is a path $(v_0, v_1, \dots, v_n)$ where v_0 is a leaf and v_n is a root.

Rooted trees enjoy a natural notion of hierarchy, where the root is considered the unique common ancestor of all other vertices. Algorithms on rooted trees can leverage this structure in contexts where vertices inherit behavior from their ancestors. For example, a misconfigured amplifier can pass a resonant peak to all downstream CPE. By identifying the relevant set of CPE and working backwards, the amplifier can be identified.

Our setup presents two main challenges. First, trees are the sparsest connected graphs, making them poor candidates for algorithms that rely on interconnectedness to simulate message passing or information spread. Many out-of-the-box solutions to problems like clustering and classification fall into this class of algorithms. Second, most approaches to analyzing graphs with vertex attributes assume that *all* vertices of the graph are attributed. But that is not the case here; while we have telemetry for the CPE, we have none for the internal vertices. In this way, access network graphs can be considered discrete "sensor networks," where the sensors are the CPE.

4.1. Binary Attribute Clustering: Combinatorial Approach

We start with the simplest possible case, when the device telemetry consists of a single binary variable, and apply our technique to the use case from Section 2.1. In practice, this variable could be truly binary (e.g. online vs. offline) or could express whether a continuous variable meets a certain threshold (e.g. whether signal-to-noise ratio is above or below 35 dB). For uniformity, we will call the values of this binary variable *impaired* and *unimpaired*. We will use the terms "CPE" and "device" interchangeably, so "device" necessarily means a customer device.

Let $(v_0, v_1, \dots, v_n)$ be a crawl in an access network graph, so that v_0 is a device, i.e. a leaf, and v_n is the RF node, i.e. the root. At each step of the crawl, we take a quantitative measurement $m(i)$ of the devices downstream of v_i. For example, let D_i denote the set of all devices downstream of v_i, and let I_i denote the set of devices in D_i that are impaired. We could take $m(i)$ to be the *precision*, i.e. the fraction of devices downstream of v_i that are impaired:

$$p(i) = \frac{|I_i|}{|D_i|}.$$

Alternatively, we could take $m(i)$ to be the *recall*, i.e. the fraction of all impaired devices that are downstream of v_i:

$$r(i) = \frac{|I_i|}{|I_n|}.$$

For further discussion on these measurements, see Section 6.1 of Dugan et al. (2022).

We fix a value M to act as an inclusive lower threshold for the measurement $m(i)$. Simple modifications can be made for upper or exclusive thresholds. Typically, we will assume that the threshold is satisfied at the starting device, i.e. $m(0) \geq M$. The *terminal* is the last vertex in the crawl whose measurement satisfies the threshold. In other words, the terminal is v_{s-1}, where s is the smallest value of i for which $m(i) < M$, or $s = n + 1$ if no such value exists. The *target* is the first vertex in the crawl whose set of downstream impaired devices is the same as the terminal's. In other words, the target is v_t, where t is the smallest index such that $|I_t| = |I_{s-1}|$.

An example crawl is illustrated in Figure 4. Each vertex v_i of the crawl is labeled by its index i. There are 11 impaired devices, colored blue, and 4 unimpaired devices, colored gray. We take $m(i) = p(i)$ to be the precision and $M = 1$. The terminal, colored red, is the last vertex in the crawl for which all downstream devices are impaired. The target is colored green. The remaining vertices of the crawl are colored orange.

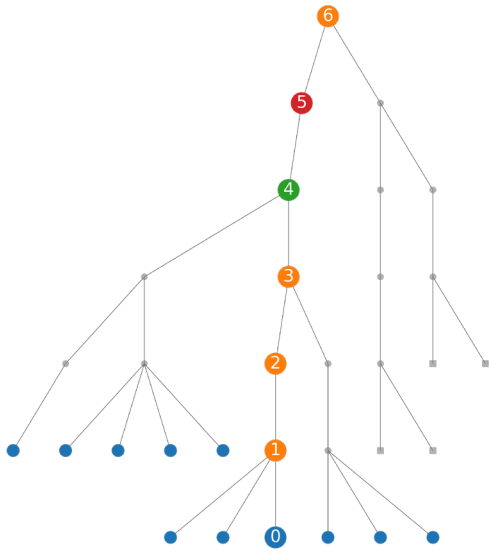

Figure 4 – Precision-based crawl in an access network graph with terminal and target highlighted

To cluster the impaired devices on the node, we perform a crawl starting at each impaired device. In this way, each impaired device is associated with a target. Sometimes the set of all targets is an *antichain*, in the sense that no target is an ancestor of any other target. In this case, we can partition the impaired devices into disjoint clusters by grouping together all devices with the same target. The clusters can be interpreted as separate underlying network impairments, each equipped with a corresponding *root cause vertex*, i.e. the target itself, that is interpreted as the source of the cluster's impairment. The validity of these interpretations depends on data quality and the choices of measurement and threshold.

The targets do not always form an antichain, however. If some targets are ancestors of other targets, measures must be taken to prevent the clusters from overlapping. One such measure is to discard any

targets that are descendants of another target in the set. The resulting "minimal" set of targets will guarantee a partition into disjoint clusters. Another remedy is to order the targets (or, equivalently, the impaired devices), and to define each cluster as the set of impaired devices downstream of the target that do not belong to any previous cluster. These "exclusionary" clusters will depend on the ordering; choosing an appropriate ordering is a potentially subtle problem.

This procedure can be used to detect incompatible legacy equipment as described in Section 2.1. Here a customer device is considered unimpaired if it is bonded on any new channels and impaired otherwise. Since all devices downstream of incompatible legacy network elements are guaranteed to be impaired, we can take $m(i) = p(i)$ to be the precision and $M = 1$. Each crawl will then terminate at the last step for which every downstream device is impaired.

The terminals always form an antichain in the case $M = 1$. They do not necessarily form an antichain, however, if $M < 1$. This threshold could be a more appropriate choice if the impairment in question were not guaranteed to impact the telemetry of every single downstream device. An example result for the case $M = 1$ is shown in Figure 5, where the impaired devices and root cause vertices are colored according to their corresponding cluster (red, green, orange or blue).

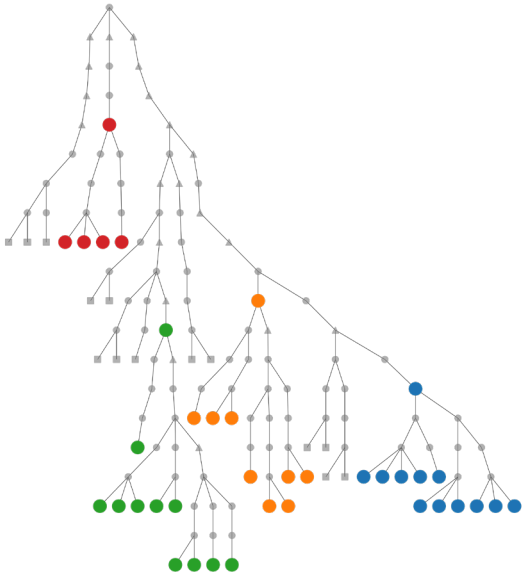

Figure 5 – Device clusters and corresponding root cause vertices

In practice, additional context is often needed to generate meaningful tickets for field teams. For example, how many homes are impacted by a given cluster? Is the root cause a cable drop in a multi-dwelling unit (MDU)? Is it an "end-of-line" element? Answers to these questions can help decide how to prioritize different network events and which personnel to dispatch. The algorithm's ability to provide these answers depends on what data is available in the graph database.

4.2. Continuous Attribute Clustering: Statistical Approach

We now consider the case where the device telemetry takes continuous values in order to address the use case from Section 2.2. We will continue to use the notation from Section 4.1, where $(v_0, v_1, \dots, v_n)$ is a crawl in an access network graph and $m(i)$ is a measurement at each step i of the crawl. It is convenient to think of this process from a statistical point of view, where at each vertex v_i the attribute values of the

downstream devices constitute a sample from a probability distribution, and the measurement $m(i)$ is a descriptive statistic of this sample. With binary or categorical attributes, the distribution is discrete, and the statistic is enumerative. With continuous attributes, the distribution is continuous, and the statistic can take more familiar forms like the mean, standard deviation, etc.

In addition to allowing continuous attributes, we will generalize the previous setup in several ways. First, instead of a single measurement $m(i)$, we consider a family of measurements $m_k(i)$. For example, $m_1(i)$ could be the maximum attribute value across all devices downstream of v_i, and $m_2(i)$ could be the median. Second, instead of a fixed threshold M_k for each measurement, we define dynamic thresholds $M_k(i)$ that can depend on the current step i or on previous steps. For example, $M_1(i)$ could be the 50^{th} percentile of the attribute values of all devices downstream of v_i.

Depending on the problem, we will have different criteria for where crawls begin and end. For where to begin the crawls, there is no longer a built-in notion of "impaired" devices as in the binary case, so we must define a set of impaired devices based on the problem statement and telemetry. As for terminals, there are many ways to generalize the termination criterion $m(i) < M$ to the multi-measurement case. Typically, a crawl will terminate when $m_k(i) < M_k(i)$ for *any* k. As before, simple modifications can be made for upper or exclusive thresholds $M_k(i)$. Following the same clustering process as in Section 4.1, we obtain disjoint clusters of the impaired devices with a root cause vertex associated to each cluster.

We can use this setup to detect pockets of customer devices with degraded MER as discussed in Section 2.2. Because MER is measured on multiple channels, the telemetry in this example is vector-valued, and the relevant probability distributions are multivariate. For simplicity, we will assume that all devices are bonded on the same channels. In reality, however, devices are commonly bonded on different channels, so the telemetry vectors might contain null values. The null values can be either ignored or imputed.

In this example, a device is impaired if it is bonded on any sufficiently "degraded" channel, where the definition of "degraded" depends on context. We set the following measurements and thresholds:

- $m_1(i)$ the maximum value of $f(d)$ as d ranges over all devices downstream of v_i, where $f(d)$ is the minimum MER value for device d across all degraded channels
- $m_2(i)$ the median of $g(c)$ as c ranges over all degraded channels, where $g(c)$ is the standard deviation of MER values on channel c across all devices downstream of v_i
- $m_3(i)$ the 80^{th} percentile of $h(d)$ as d ranges over all devices downstream of v_i, where $h(d)$ is the minimum MER value of device d across all degraded channels
- $M_1(i)$ a constant value; MER above this value is considered "very good"
- $M_2(i)$ a constant value; MERs differing by more than this value are considered "dissimilar"
- $M_3(i)$ a certain percentile of MER values across all devices downstream of v_{i-1} (the *previous* step in the crawl) and all degraded channels

The crawls terminate when $m_k(i) > M_k(i)$ for any i. For $k = 1$, the termination condition checks whether the MER of downstream devices is too high for the sample to be considered impaired; for $k = 2$ and $k = 3$, the conditions detect whether the current step has added outliers to the previous MER sample. In this example, we take the "minimal" approach to clustering described in Section 4.1. Devices with healthy MER will be clustered by themselves, due to the crawl termination conditions; all singleton clusters and unimpaired devices can be merged to form a cluster of "normal" devices. This is depicted in Figure 6, where the lone impaired cluster is orange, and the merged "normal" cluster is blue.

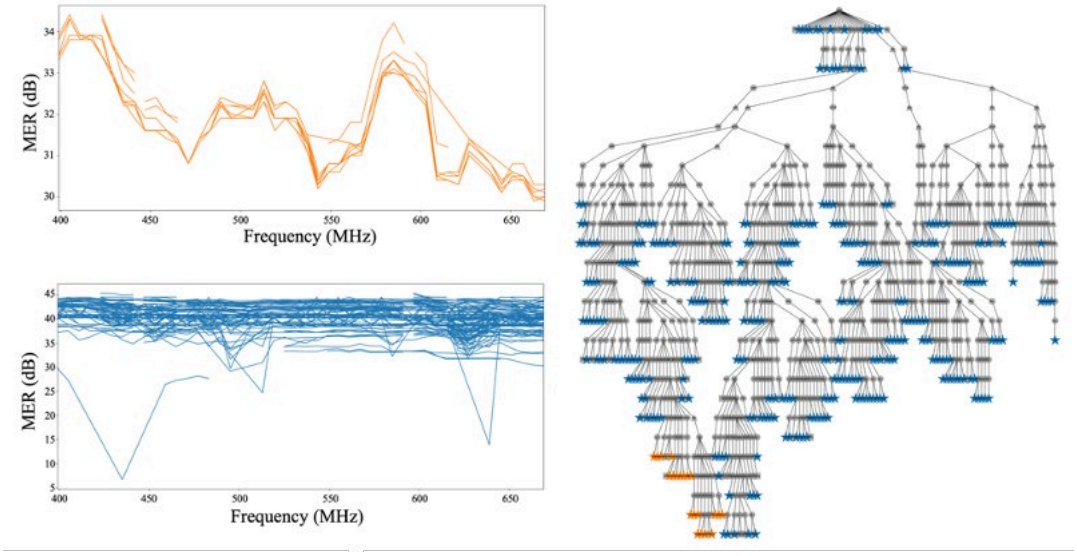

Figure 6 – Clustering impaired downstream SC-QAM MER with network topology and graph algorithms

4.3. Cluster Analysis

So far, we have described graph algorithms that use telemetry and graph structure *simultaneously* to perform clustering and root cause analysis. But this is not always the preferred approach. For some problems, it makes sense to operate on telemetry alone, or on the graph structure alone, and to combine the results post hoc. Alternatively, we might wish to leverage existing solutions, such as legacy clustering or classification algorithms that do not make use of topology, and enrich them with graph data. In this section, we apply these approaches to the use cases of Sections 2.2 and 2.3.

4.3.1. Root Cause Analysis

In some cases, we are given existing clusters of impaired devices, and our goal is to identify the network element that most likely caused each impairment cluster. This is essentially the reverse of the problems in Sections 4.1 and 4.2, where we identified a root cause vertex *first*, and the clusters are obtained as a corollary. Hence a different approach is needed.

For example, suppose that we want to perform root cause analysis for impairment patterns in FBC data, as described in Section 2.3. In this scenario, devices are classified by impairment type (e.g. resonant peak), and devices with the same impairment type(s) are clustered together if their impairment patterns exhibit similar characteristics (e.g. resonant peaks with the same resonant frequency). This clustering step is needed to distinguish separate impairments of the same type occurring simultaneously. We can perform root cause analysis on these clusters by viewing them in the appropriate access network graph. In addition to providing key diagnostic context, this process also allows us to filter out clusters for which the clustering algorithm has underperformed, or the underlying data quality is poor. In this way, the telemetry and topology can act as checks and balances for one another, instead of as potentially noisy simultaneous input.

Returning to the general setting, let $C_1, C_2, \ldots, C_n$ denote device clusters. These clusters need not be disjoint. In the case of spectrum impairments, for example, the clusters might overlap if some devices are

subject to multiple underlying network impairments with distinct patterns; ideally, we would see one cluster for each impairment pattern and another cluster comprising all devices with "normal" spectra.

Naively, we could take the root cause vertex of the cluster C_k to be the *lowest common ancestor* of all devices in C_k. However, this approach gives disproportionate influence to "topological outliers," i.e. devices in C_k that are topologically distant from the other devices in the cluster. We need an approach that can discriminate some topological outliers, since they are common in practice.

We will focus on a particular cluster C_k and consider a device "impaired" if it belongs to C_k or "unimpaired" otherwise. Recall the definitions of *precision* $p(i)$ and *recall* $r(i)$ from Section 4.1. Instead of taking these measurements at a step i of a crawl, we now take them at any vertex v in the graph. Thus, for example, $p(v)$ is the fraction of devices downstream of v that are impaired. We can define a corresponding *F-score* at each vertex as follows:

$$F_\beta(v) = \frac{(1+\beta^2)\cdot p(v)\cdot r(v)}{\beta^2\cdot p(v) + r(v)},$$

where β is a positive parameter. This score can be interpreted as a weighted harmonic mean of precision and recall, where recall is considered roughly β times as important as precision. It is generally high if both precision and recall are high, and low otherwise. The usual F_1 score is the special case $\beta = 1$.

The root cause vertex of cluster C_k is taken to be the vertex v that maximizes $F_\beta(v)$. Higher values of β will give more importance to topological outliers, resulting in root cause vertices closer to the RF node. Lower values of β will give less importance to topological outliers, resulting in root cause vertices closer to impaired devices. Choosing an appropriate value of β depends on the problem statement, the quality of the clusters, and the nature of the impairments in question. Figure 7 shows a cluster of devices exhibiting similar water wave patterns with a single topological outlier and the associated root cause vertex.

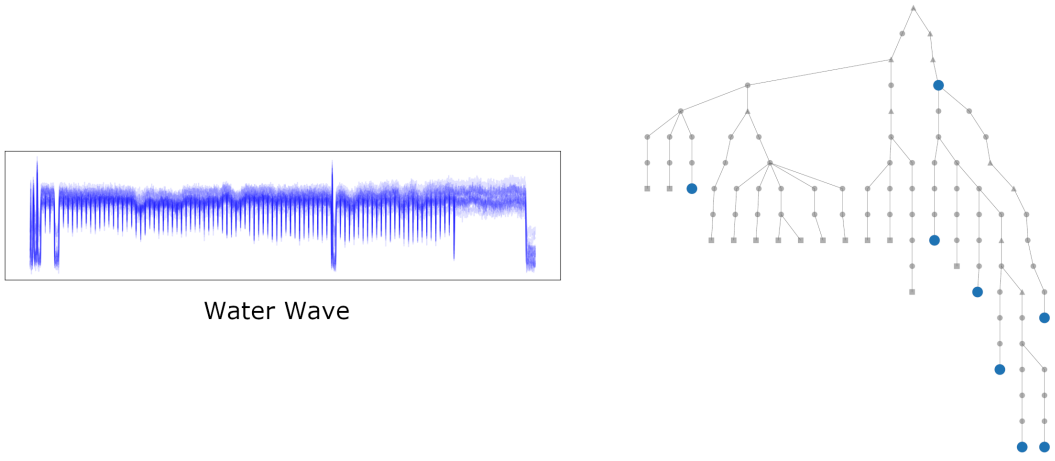

Figure 7 – Cluster of device spectra with similar water wave patterns and corresponding root cause vertex

It is not always possible to adequately characterize a device cluster with a root cause vertex. Clusters can exhibit sparseness, bifurcation, or otherwise low topological correlation when viewed within the network topology. Typically, these phenomena point to one or more of the following issues:

1. Poor clustering performance
2. Inaccurate data, either in the telemetry or in the graph database
3. Indistinguishable telemetry characteristics across distinct impairments.

An example of Item 3 in the case of FBC could be two impairment patterns occurring on the same frequencies, such as a water wave superimposed with a standing wave, or two water waves overlapping. This poses a separate issue from poor clustering performance, since even an ideal clustering algorithm could not necessarily distinguish devices subject to the different underlying impairments without additional data.

To detect clusters with low topological correlation, we enforce lower thresholds for precision and recall. Specifically, we set numbers P and R and check whether $p(v) \geq P$ and $r(v) \geq R$, where v is the candidate root cause vertex described above. If either condition fails, the cluster is rejected as having low topological correlation and receives no root cause. The cluster can then be reported and evaluated to determine the influence of Items 1–3 above.

4.3.2. Multi-Cluster Correlation

When clustering impaired devices based on multiple categories of telemetry, different clusters can point to related network impairments, or even the same underlying impairment. For example, a cluster of devices with degraded MER and a cluster of devices exhibiting water wave could point to the same issue if the water wave is *causing* the low MER. Thus we need a way to determine if clusters and root cause vertices from different sources are correlated, independent of the individual algorithms.

Let K and L denote two clusters in the same access network graph, potentially from different algorithms, with associated root cause vectors u and v, respectively. A classical measure of correlation between clusters K and L is the *Jaccard index*:

$$\frac{|K \cap L|}{|K \cup L|},$$

Where ∩ denotes intersection and ∪ denotes union. This measure is appropriate when the clusters are generally "comprehensive," in the sense that they contain most of the devices they should, and few they should not. In practice, however, clustering results are often sensitive, noisy and non-deterministic. This can be due to inconsitent data quality, the behavior of the algorithm, or both.

To address these limitations, we focus on root cause vertices instead of the clusters themselves. Ultimately, we want to know if a technician can solve multiple issues by visiting a single network element. The particular devices implicated in a cluster often have no bearing on this result, especially taking into account the performance of the clustering algorithm. For each vertex w, let $S(w)$ denote the set of all CPE downstream of w. We define the *Jaccard index* of root cause vertices u and v as

$$J(u, v) = \frac{|S(u) \cap S(v)|}{|S(u) \cup S(v)|},$$

In other words, $J(u, v)$ is the number of common downstream devices between u and v divided by the total number of downstream devices. Higher values indicate higher correlation between the root cause

vertices. Identical root cause vertices will have Jaccard index 1, while root cause vertices from distinct branches of the graph will have Jaccard index 0. A Jaccard index strictly between 0 and 1 occurs when one root cause vertex is an ancestor of the other. This definition can be adapted to arbitrarily many sets of clusters.

The Jaccard index of root cause vertices can be used to measure correlation between degraded MER (Section 2.2) and impairment patterns seen on FBC (Section 2.3). Here the root cause vertices come from the algorithms in Sections 4.2 and 4.3.1, respectively. Root cause vertex pairs with sufficiently high Jaccard index are interpreted as expressions of related or identical underlying network impairments. This effectively correlates demand maintenance (DM) events with proactive maintenance (PM) events.

5. Algorithm Performance

The algorithms in Section 4 perform well when applied to the use cases in Section 2. The precision-based algorithm in Section 4.1 has been used to identify legacy equipment preventing mid-split enablement on the path to 10G (Section 2.1). In this application, the root cause vertex identified by the algorithm is within 300 feet of the actual legacy equipment in 95% of cases. A similar approach has also been used to identify root causes of severe roll-off in a region of spectrum intended for orthogonal frequency-division multiplexing (OFDM) expansion; see Harb et al. for further discussion (2023).

The statistical algorithm in Section 4.2 has been used to identify multi-home MER impairments and their root causes (Section 2.2). The results of the algorithm are overlayed with PER telemetry to determine whether the degraded MER is impacting customers. A relevant network impairment is found at the predicted root cause vertex in 85% of customer-impacting cases. To enhance these results, the root cause analysis and cluster correlation algorithms in Section 4.3 are used to identify spectrum impairments (Section 2.3) that are likely related to or even causing multi-home MER issues.

5.1. Graph Algorithms vs. Geospatial Clustering

Network topology data is sometimes unavailable or of insufficient quality for meaningful analysis. When this happens, a non-graph-based fallback approach is necessary. There is a question of how such an approach will perform when compared to the graph algorithms in Section 4. To answer this question, we perform clustering on CPE downstream of 2,071 physical layer (PHY) devices (remote PHY devices (RPDs)) with high-quality network topology data and a single binary attribute (“impaired” vs. “unimpaired”). We cluster the impaired customer devices on each RPD using three approaches:

1. (Baseline) All impaired devices assigned to the same cluster
2. (Geospatial) Density-based spatial clustering of applications with noise (DBSCAN) on latitude/longitude alone
3. (“Ground truth”) Clustering with graph data, precision-based approach described in Section 4.1.

We assess the performance of the baseline and geospatial approaches, treating the results of the graph-based clustering as the ground truth. Let n be one of the RPDs analyzed. Define S to be the set of all pairs of impaired devices on RPD n such that both devices belong to the same cluster, according to the graph-based approach. Let T be defined similarly but with one of the non-graph-based approaches instead. The *precision* of the non-graph-based approach for node n can be defined as

$$p = \frac{|S \cap T|}{|T|}.$$

In words, p is the fraction of device pairs sharing the same non-graph-based cluster that also share the same graph-based cluster. The *recall* is then

$$r = \frac{|S \cap T|}{|S|}.$$

In words, r is the fraction of device pairs sharing the same graph-based cluster that also share the same non-graph-based cluster. The results are illustrated in Figure 8, where we plot histograms and cumulative distribution functions (CDFs) of precision, recall and F_1 score across all 2,112 nodes. Here the F_1 score is the usual harmonic mean of p and r.

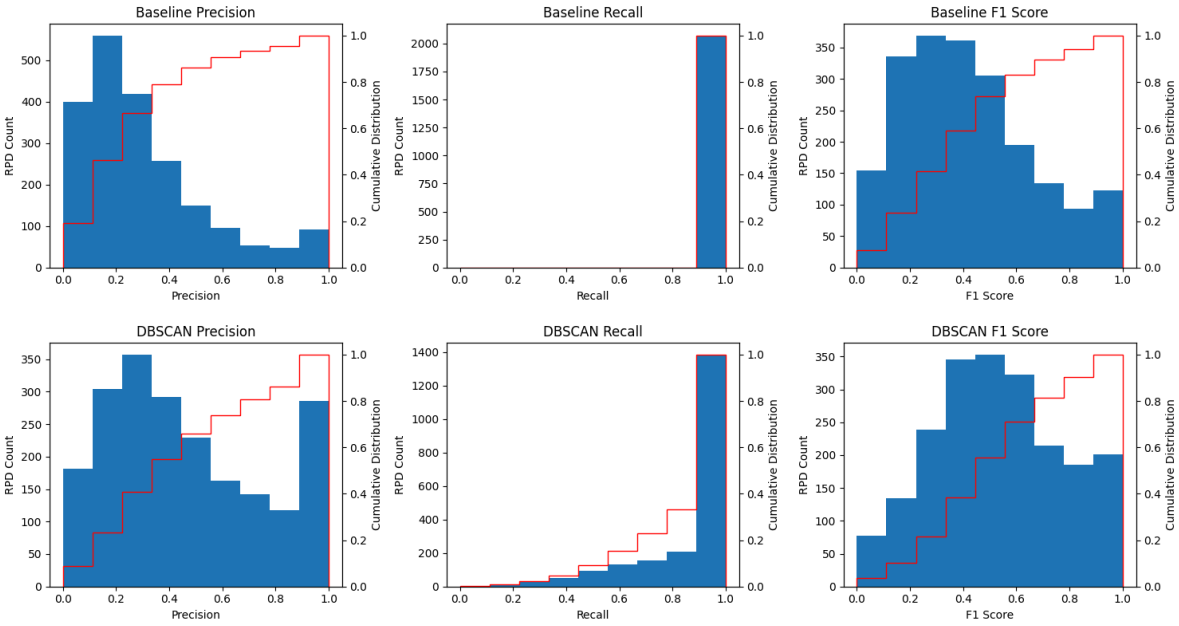

Figure 8 – Performance of baseline and geospatial clustering vs. graph-based clustering

A numerical summary is given in Table 1. We compute the mean and median of precision, recall and F_1 score for the baseline and geospatial approaches across all RPDs, and then for each metric compute the mean and median *difference* between the two approaches across all RPDs. While the baseline approach has perfect recall by design, the geospatial approach outperforms the baseline in both precision and F_1 score, suggesting that it improves on the baseline overall. However, the average F_1 score for the geospatial approach is 0.54, which is still low. This suggests that the geospatial approach does not perform similarly overall to the graph-based clustering.

Table 1 – Average performance metrics and lift of baseline and geospatial clustering vs. graph-based clustering

	Baseline	Geospatial	Geo minus baseline
Precision (mean)	0.30	0.47	0.16
Precision (median)	0.24	0.40	0.11
Recall (mean)	1.00	0.87	−0.13
Recall (median)	1.00	0.98	−0.02
F_1 score (mean)	0.42	0.53	0.11
F_1 score (median)	0.38	0.52	0.10

6. Conclusions and Next Steps

We presented a suite of approaches to combining node-level telemetry with graph algorithms to identify and describe multi-home network impairments. Applying similar algorithms to additional categories of telemetry will help to further reduce the number of jobs generated for field teams while increasing the number of customer issues resolved. Possible candidates for such telemetry include MER for OFDM channels, upstream forward error correction (FEC) rate, and downstream PER.

With more telemetry comes more complexity, however. The approaches in this paper treat each category of telemetry separately, comparing and combining the results at the end (e.g. by cluster correlation in Section 4.3.2). As more telemetry is included in our analyses, it will be critical to evaluate this divide-and-conquer approach against more holistic approaches that attempt to perform clustering and root cause analysis on multiple attributes at once. The goal, ultimately, is a comprehensive data pipeline that ingests all pertinent customer device telemetry and generates jobs that are optimized for customer impact.

7. Acknowledgments

The authors thank Andy Martushev, Doug Sitkin, and Nathaniel Lee for operationalizing the algorithms in this paper and providing critical technical input. The authors also thank Ramya Narayanaswamy for helpful comments on a draft of this paper.

Abbreviations

BG	bonding group
CDF	cumulative distribution function
CMTS	cable modem termination system
CPE	customer premise equipment
DAA	distributed access architecture
DAAS	DAA switch
DBSCAN	density-based spatial clustering of applications with noise
DM	demand maintenance
FBC	full band capture
FEC	forward error correction
HFC	hybrid fiber-coaxial
MDU	multi-dwelling unit
MER	modulation error ratio
MIB	management information base
MSO	multiple systems operator
OFDM	orthogonal frequency-division multiplexing
PER	packet error rate
PHE	primary headend
PHY	physical layer
PM	proactive maintenance
PPOD	physical point of deployment
RF	radio frequency
ROCI	routing of cable infrastructure
RPD	remote PHY device
RX	receive
SC-QAM	single-carrier quadrature amplitude modulation
SG	service group
SHE	secondary headend
TX	transmit

Bibliography & References

Dugan, K., Evans, J. & Harb, M. (2022). A Deep Learning Approach for Detecting RF Spectrum Impairments and Conducting Root Cause Analysis. *SCTE 2022 Fall Technical Forum*.

Harb, M., Shen, W., Stehman, M., Walavalkar, S. & Rice, D. (2023). Accounting for Every MHz of Bandwidth: Data & Algorithms for Artifact Discovery and Close-Packing of QAMs in Support of Spectrum Activation. *SCTE 2023 Fall Technical Forum*

Narayanaswamy, R., Subramanya, K., Prodan, R. & Wolcott, L. (2021). When Physical Layer Simulation Gets Real. *SCTE 2021 Fall Technical Forum*.

Prodan, R. S. (2020). Optimizing the 10G Transition to Full-Duplex DOCSIS® 4.0. *SCTE 2020 Fall Technical Forum*.

Sundaresan, K. (2022). Network Capacity Options on the Path to 10G. *SCTE 2022 Fall Technical Forum*.

