A Machine Learning Pipeline for D3.1 Profile Management

A Technical Paper prepared for SCTE•ISBE by

Maher Harb
Senior Principal Data Scientist
Comcast
1800 Arch Street, Philadelphia, PA 19103
267.260.1846
maher_harb@comcast.com

Jude Ferreira, Comcast

Dan Rice, Comcast

Bryan Santangelo, Comcast

Rick Spanbauer, Comcast

Table of Contents

Title **Page Number**

List of Figures

Title **Page Number**

List of Tables

Title **Page Number**

Introduction

We have entered an era where leveraging Machine Learning to optimize the performance of cable access networks is possible and, perhaps, even a must. The fast arising opportunities in this realm are due to advances in cable technology and increasing investment in data science functions across organizations. Specifically, DOCSIS 3.1 (D3.1) includes Orthogonal Frequency Division Multiplexing (OFDM), enabling the possibility of tailoring the modulation of OFDM channels to realize much improved spectral efficiency and impairment resiliency. Additionally, due to the nature of the wider OFDM channels, Comcast identified several opportunities and key deployment challenges affecting network stability & performance. As part of operationally hardening D3.1, it became clear that an effective modulation Profile Management Application (PMA) is essential for operating D3.1 to its full potential. The initial perspective -- that PMA was an optimization technique to maximize capacity in the future -- changed to a conclusion that PMA is really a table-stakes feature required to ensure network stability, manage operational metrics, and ensure a great customer experience. This document describes the profile management solution developed to address these challenges. As a primer to the ensuing discussion, consider the distribution (shown in Figure 1) of Modulation Error Ratio (MER) collected from Comcast's entire population of D3.1 devices. The distribution, while encapsulating information aggregated across billions of subcarriers, hints at the core idea of PMA: since the quality of the spectrum varies across the network, customizing modulation across subcarriers and devices holds the opportunity to enhance network performance in terms of increasing both capacity and resiliency. The goal of PMA is to pursue this ideal.

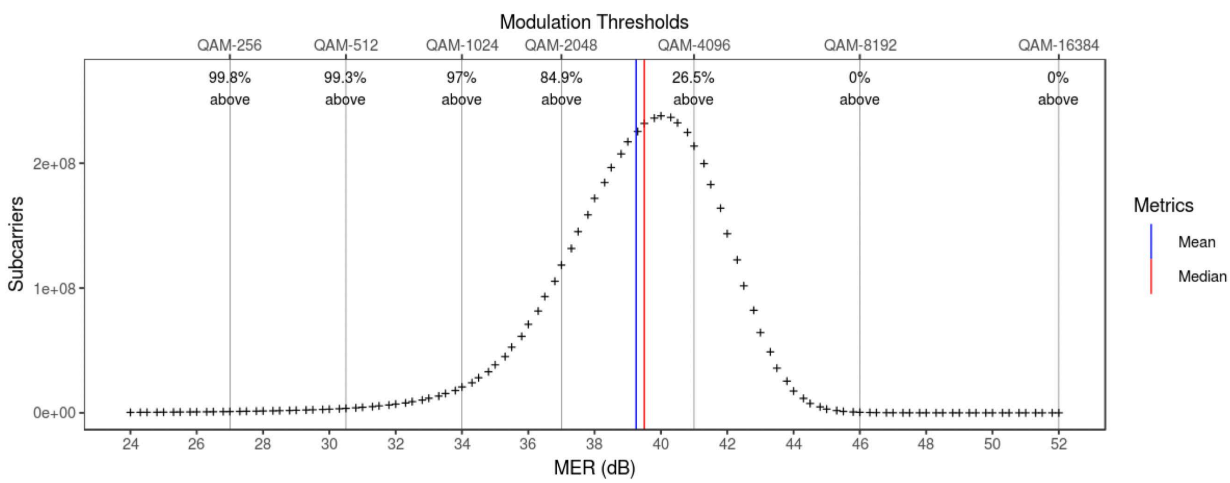


Figure 1. Distribution of MER across billions of D3.1 subcarriers (plus shaped data points). The vertical gray lines indicate recommended thresholds for the respective modulation efficiency. The vertical blue and red lines represent distribution mean and median respectively.

The paper is organized as follows: **DOCSIS 3.1 Hardening** describes our efforts in addressing D3.1 deployment challenges; these efforts are an important precursor to PMA. **Problem Statement** introduces formulation of the PMA problem. **Overview of Solution** presents the high-level PMA solution architecture. **MER & Time** describes our solution for addressing MER variation in time. **Core Algorithm** introduces the algorithm developed for constructing D3.1 Profiles. **Lab Environment** describes establishing a lab for testing the PMA solution. **Pattern Detection** describes a host of algorithms aimed at detecting impairments; these are complimentary to the PMA effort. **Future Work** comments on the future evolution of the PMA algorithm.

DOCSIS 3.1 Hardening

After initially deploying D3.1 across the Comcast network, the operational service performance was evaluated and compared to DOCSIS 3.0 (D3.0) customer metrics. The initial analysis indicated that D3.1 was under-performing D3.0, based on a variety of operational metrics including truck rolls and call-in rates. At first reaction, this seemed very unlikely. All theoretical models and lab testing had shown that D3.1 is fundamentally more robust, with higher capacity based on the principles of OFDM technology, including the advanced Low Density Parity Check (LDPC) algorithm used in Forward Error Correction (FEC). As the root cause of the operational challenges were evaluated, a consistent theme emerged: While D3.1 is fundamentally much more capable than D3.0, it requires operational consideration when deploying, in order to fully realize its potential. Therefore, operational hardening of the D3.1 solution is required.

We began the D3.1 hardening effort with the goal of achieving parity of operational key performance measures relative to our mature D3.0 services. Along the way, we identified several opportunities to improve processes, configurations, and in some cases, entirely new techniques that ultimately would improve the service and the customer experience. While many of these were related to software maturity of the gateway, the operational metrics were equally challenged for customer-owned cable modems (CM). Upon our initial allocation of root cause, we believed that about 40% of the issues were related to D3.1 technology. We quickly moved to develop a collection infrastructure, to gather data and improve our visibility of the network, and diagnosed many specific customer challenges to isolate the key D3.1 operational challenges. In parallel, we began working on a solution for PMA.

Several key areas were identified for operationalizing D3.1 and are briefly described here, supporting the essential nature of profile management for D3.1.

1. Standardized CMTS configurations and software

Inconsistencies were discovered in detailed D3.1 CMTS (Cable Modem Termination System) configurations and software (SW) across the network and across CMTS platforms. In addition to SW updates, the primary configuration adjustments included setting the Cyclic Prefix to 512 samples, the roll-off period to 256 samples, and the time-interleaver to 16 OFDM frames.

2. HFC Roll-off Reduction

The initial spectrum deployment targets were 96 MHz to be placed in the highest spectrum of the Hybrid Fiber Copper (HFC) network above the existing video and D3.0 channels. In many service groups, a 96 MHz OFDM channel required that some of the OFDM channel be located in the HFC design roll-off area; where roll-off refers to spectrum above formal plant design. For example, in a 96 MHz channel, in some instances, the highest frequency was at 774 MHz for a 750 MHz HFC network, resulting in 24 MHz of channel bandwidth in the roll-off. In many cases, this was just fine, as this spectrum had been vetted and used previously for additional SC-QAM channels. However, in other cases, such as the example in Figure 2, the roll-off spectrum had reduced MER with the higher attenuation and response of the network, depending on N+x cascade length, where x represents the number of amplifiers in the signal path between the node (N) and the customer. In Figure 2, the red line represents the level for a flat QAM-256 modulation. The solid colored lines on each chart represent select percentiles in the distribution of MER per subcarrier. As can be seen in many of the channels, 75% of the MER samples for the modems will support modulation higher than QAM-256, but some will not support QAM-1024 or 2048. However, a small number of modems have degraded MER level in the roll-off region. To mitigate this, in the short term, we reduced the size of the channel back to the HFC roll-off edge, thereby reducing capacity. For the longer term we are able to run into the roll-off spectrum effectively by using the PMA solution to modulate the subcarriers consistent with the network roll-off characteristics.

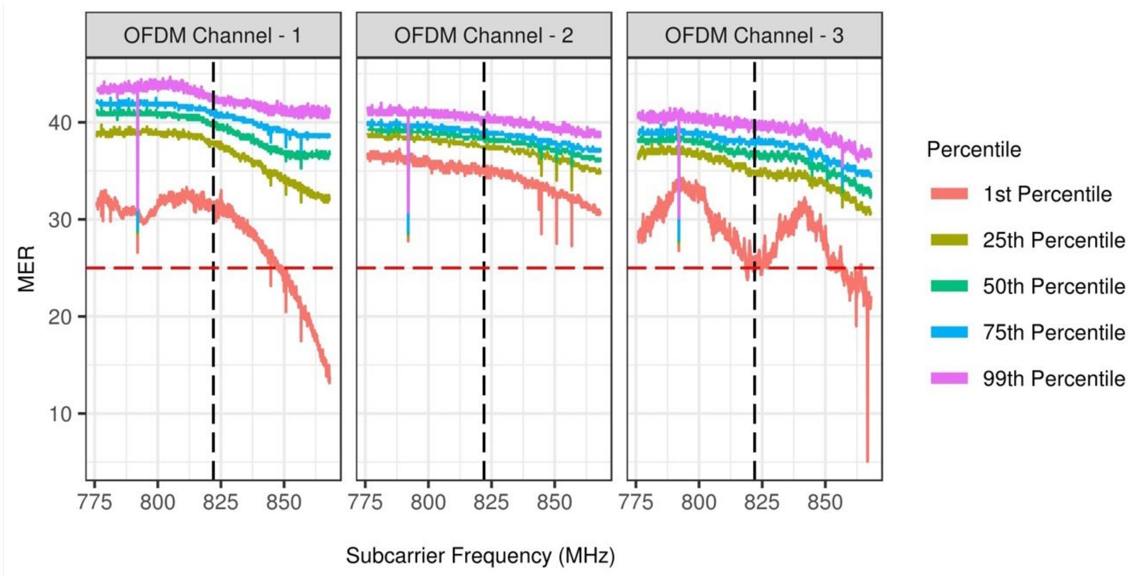


Figure 2 - MER distribution for select OFDM Interfaces. The dashed red line represented the QAM-256 threshold. While the spectrum supports higher modulation overall, the roll-off region (upper edge) has MER levels falling below QAM-256.

3. PHY-Link Channel (PLC) Location

On some of the CMTS population we noticed a high number of PHY-Link Channel (PLC) CM-STATUS 21 failure messages, along with PLC error rates. Some of these were resolved by fixing some CMTS and CM software. In several other cases, we noticed that the PLC channel was placed within spectrum used by the mobile carriers and was failing due to ingress. To resolve these issues, we performed analysis described later in this paper to join together OFDM channel locations with spectrum licenses obtained from a spectrum allocation database. As a result, we were able to find "gaps" in the spectrum allocation for each county associated with each HFC node and, through a rule set, identify the most common spectrum locations to reduce the probability of overlap between our PLC channel and offending mobile carriers. In most regions we were able to find a single spectrum location based on the HFC bandwidth (625, 750, 860, and 1000 MHz) and OFDM channel location to consistently place the PLC in a lower risk location. In several network regions, different localities required different PLC options where the mobile spectrum is widely licensed and deployed. We also developed an algorithm based on MER that could be used to find the best possible spot for the PLC channel, when there was not a great opportunity for lower risk, due to mobile spectrum licensing. We moved all of the PLC locations across the network to mobile spectrum gaps and reduced the rate of PLC failure events. The PMA software will be able to recommend optimal PLC locations based on statistical analysis of MER per subcarrier. The PLC current locations are shown in charts as described in Section 7 (Pattern Detection), and alternate locations are identified by finding the best possible 400 kHz or 6 MHz of bandwidth based on MER.

4. Ingress

We identified a variety of ingress sources in the network; many were based on wireless ingress such as LTE (Long Term Evolution). Several wireless ingress issues were caused by uncoordinated configuration of access network systems. For example, a small number of ingress sources traced to the use of our sweep generator systems placing insertion points or QAM carriers into the service group, which were not moved out of the spectrum when the OFDM channel was added. As a result, the sweep generator moving through the spectrum with a transient tone every 6 MHz introduced errors that were difficult for the CM-STATUS messaging-based management to deal with.

For example, in Figure 3, the LTE interference in this case was at a level that periodically caused the OFDM channel to go into partial service or move back and forth between profiles at such a high rate that traffic forwarding performance was severely degraded. Many of the subcarriers would easily support QAM-2048 or QAM-4096, but for a small section of the channel, even QAM-256 was not supported well all of the time. With many of these types of smaller impairments, and from a spectrum perspective, it was clear that a flat modulation profile would not be as effective in achieving the high capacity performance potential of D3.1, and a PMA solution would be needed to achieve the anticipated D3.1 capacity benefits and stabilize the network.

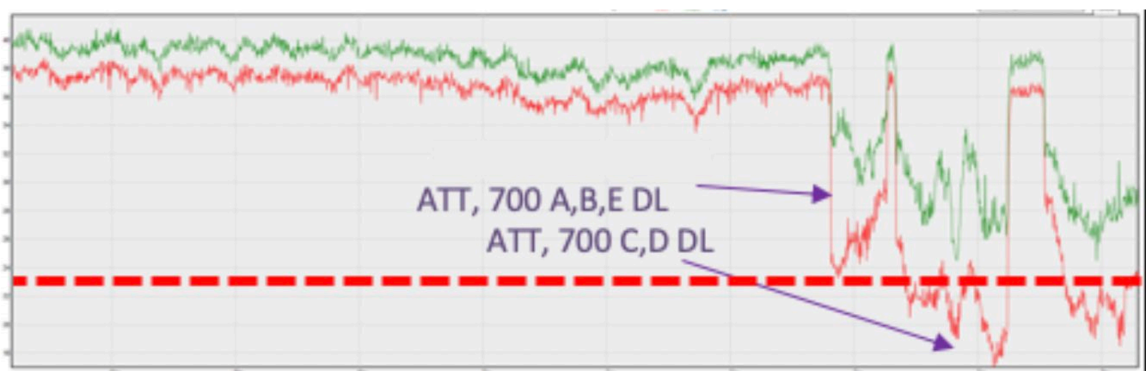


Figure 3 – MER spectra for two cable modems showing LTE Interference. The dashed red line corresponds to the MER threshold that supports QAM-256.

A high percent of OFDM channels had some subset of subcarriers across some subset or all cable modems in the service group that had RF physical layer impairments. For D3.0, when these impairments occur, the partial service and bonding mechanisms, along with a very mature capacity planning process, protects the customer experience very effectively. If one or two channels are impacted, there are more than 30 additional D3.0 channels available to provide a seamless customer experience. Respectively, without a PMA solution on an OFDM channel, a flat modulation profile is very inefficient from both a capacity and robustness perspective. Reducing modulation for all subcarriers, or losing 96 or 192 MHz of capacity, is a much more significant capacity and network stability challenge than a corresponding single D3.0 SC-QAM 6 MHz channel.

5. Partial Channel/Service impairment handling

Often, in the presence of transient ingress, or noise levels near the modulation boundary, we discovered stability issues with "profile flapping" (the condition where traffic is moved between two different profiles, back and forth, sometimes leading to instability or slow performance) based on our configuration for managing CM-STATUS messages. The CM-STATUS approach (illustrated in Figure 4) for notification from the CM to the CMTS on the health of the OFDM profile and channel elements is fairly complicated, with 3 primary state machines that must be tuned to work together, and are still immature in their implementation. The state machine, on the silicon System on Chip (SoC), makes decisions based on vendor-proprietary algorithms as to when the OFDM channel is impaired or when the channel is considered recovered from an impairment. Additionally, the time scale for consideration within those proprietary algorithms can vary. In one of our gateways, the event messages were by default sent on time frames of milliseconds or 100s of microseconds, depending on the noise characteristics. In the other gateway, the event messages were sent in timeframes of seconds or 10s of seconds. In either case, after the SoC decides on the condition, it messages the DOCSIS MAC layer, which sends it to the CMTS based on a configurable hold-off timer and state machine. These hold-off timers were set differently across CMTS platforms, creating variable behavior once the event is received. The CMTS decides how to react to the CM-STATUS event, and may apply proprietary controls in response to that message.

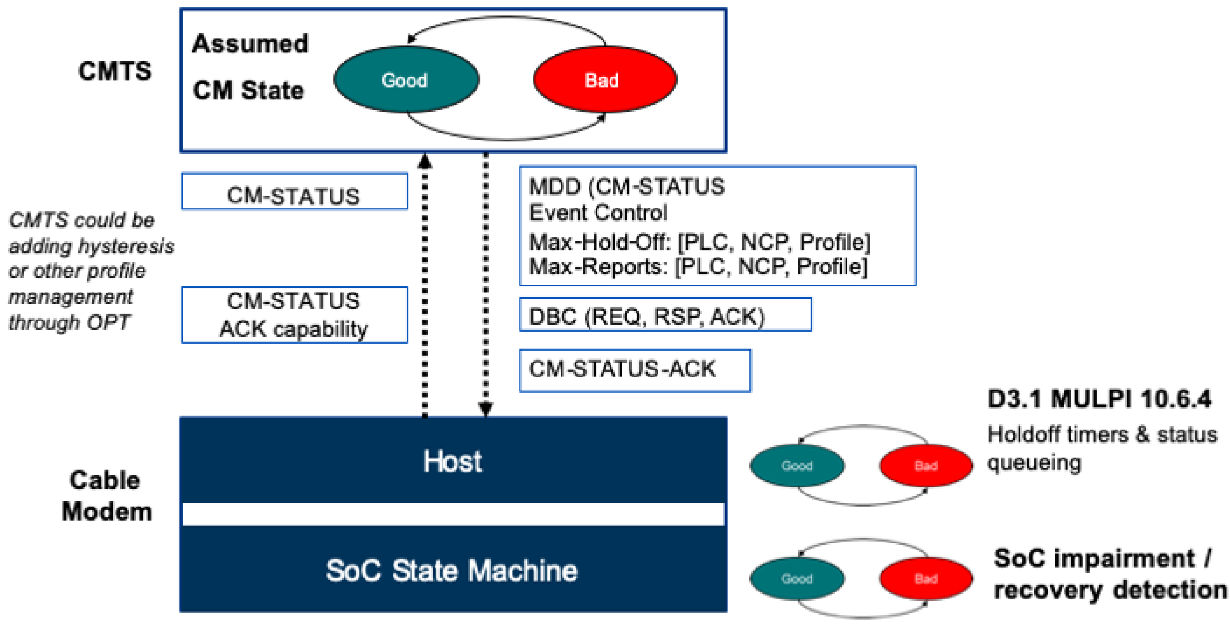
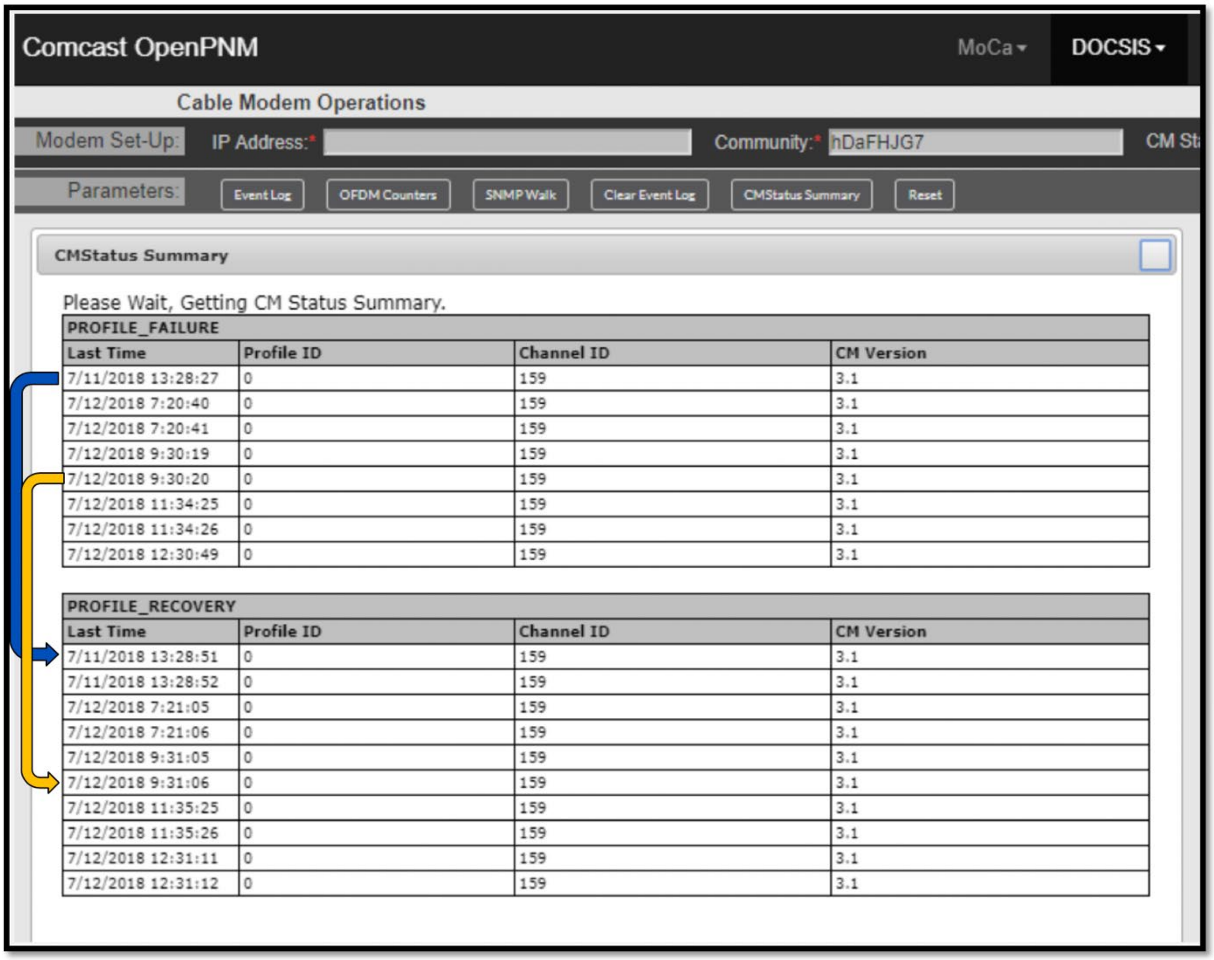


Figure 4 – Schematic illustrating the CM-STATUS message operation.

We were able to identify and reproduce a variety of scenarios that resulted in profile flapping. One example of profile flapping in the field is shown in Figure 5. Due to the holdoff settings, the CM state machine, and the CMTS configuration, the traffic would be moved from a profile that was marginal to a more conservative profile -- but then immediately moved back to the marginal profile based on the CM-STATUS messaging. With the 30 second holdoff timers configured on the CMTS, the modem would be stuck on the marginally performing profile for most of the time, with very brief transitions back and forth. The end result was traffic not getting forwarded through the CM, further demonstrating why a PMA solution is important. Profile decisions based on statistics from a data lake of channel performance, as opposed to the microsecond and second-by-second decisions that can be made by a CM and CMTS, helps to ensure network stability.

After a lot of testing and experimenting with different noise impairments recorded from the network, we were able to model and test CMTS- and DOCSIS-based settings that added the appropriate controls and hysteresis to stabilize the “profile flapping” challenges. The next section introduces the main idea behind PMA and the challenges that need to be addressed for a successful PMA deployment.



PROFILE_FAILURE			
Last Time	Profile ID	Channel ID	CM Version
7/11/2018 13:28:27	0	159	3.1
7/12/2018 7:20:40	0	159	3.1
7/12/2018 7:20:41	0	159	3.1
7/12/2018 9:30:19	0	159	3.1
7/12/2018 9:30:20	0	159	3.1
7/12/2018 11:34:25	0	159	3.1
7/12/2018 11:34:26	0	159	3.1
7/12/2018 12:30:49	0	159	3.1

PROFILE_RECOVERY			
Last Time	Profile ID	Channel ID	CM Version
7/11/2018 13:28:51	0	159	3.1
7/11/2018 13:28:52	0	159	3.1
7/12/2018 7:21:05	0	159	3.1
7/12/2018 7:21:06	0	159	3.1
7/12/2018 9:31:05	0	159	3.1
7/12/2018 9:31:06	0	159	3.1
7/12/2018 11:35:25	0	159	3.1
7/12/2018 11:35:26	0	159	3.1
7/12/2018 12:31:11	0	159	3.1
7/12/2018 12:31:12	0	159	3.1

Figure 5 - Example of Profile Flapping. Profile failure is followed ~25 sec later by recovery. The cycle of failure and recovery continues, and traffic gets blocked because the CM stays on an impaired channel.

Problem Statement

To support the increased capacity and stability of D3.1 OFDM channels, opportunities for optimization were identified. D3.1 supports the use of OFDM, in which the usable spectrum is divided into multiple narrow band (25 or 50 kHz wide) subcarriers. With OFDM, it is possible to tailor the modulation of signal to the specific spectral conditions of each of those subcarriers. There are two obvious benefits to gaining such flexibility in customizing modulation to the device (cable modem) and subcarrier levels: (1) Increasing the total capacity of an OFDM interface since the modulation scheme will no longer be dictated by those devices with relatively poor overall signal-to-noise ratios (S/N). (2) Increasing robustness of all devices by assigning impaired regions of the spectrum (as shown in Figures 2 & 3) a suitable modulation scheme, or even entirely blocking the impaired regions if needed.

To highlight the potential benefits of customizing the modulation profile, consider the plots, shown in Figure 6, of measured MER across the OFDM spectrum for a group of 20 CMs attached to the same OFDM channel. QAM-256 is the recommended modulation scheme for MER falling within the 27 to 30.5 dB range. Without the ability to configure modulation profiles, a QAM-256 modulation plan (8 bits/symbol) would be adopted across the channel to accommodate the lower performing CMs. Yet, most CMs shown in Figure 6 have superior MER characteristics; thus, they could support the use of modulation higher than the assigned QAM-256. Additionally, one device (CM84) with an impaired region, likely due to LTE interference, has otherwise favorable MER. This device may benefit from using a modulation lower than QAM-256, exclusively in the impaired region, for added robustness. Therefore, it is evident that a flat, one-size-fits-all modulation scheme is far from ideal; this is true whether the device has a healthy or an impaired spectrum.

Ideally, each device would be assigned its own fully custom-matched modulation scheme. In practice, however, customizing modulation for each device is not practical due to protocol efficiencies and current CMTS capabilities. D3.1 allows customizing modulation through the concept of *modulation profiles*, where a profile defines a specific modulation scheme for each subcarrier, over the entire channel spectrum, for which devices within an OFDM channel can share a set of tailored profiles. While the number of allowed profiles per OFDM channel with the current D3.1 specification is 16, current vendor-specific implementations limit this figure -- in some instances, as low as 3 per OFDM channel (one of which is the control profile. known as profile 0 or profile A.) Restricting the number of profiles introduces one of the main optimization problems, in which the objective is to construct suitable profiles such that the total capacity of the interface is maximized within known MER conditions, so as to not negatively affect customer experience.

Let's define the problem mathematically by first introducing some useful notation. Assume that the usable spectrum contains L frequency subcarriers, and that there are M devices/cable modems (CM) attached to the same channel and N profiles (P) to work with. The measured MER of a device is a vector of MER values of length L. A constructed profile on the other hand is defined as a vector of modulation efficiency values, also of length L. These are represented below for device j and profile k:

$$CM^{(j)} = \left[x_1^{(j)},\ \ x_2^{(j)},\ \ \cdots,\ \ x_i^{(j)},\ \ \cdots,\ \ x_L^{(j)}\right], \qquad i \in \{1,..,L\} \text{ and } j \in \{1,..,M\}$$

$$P^{(k)} = \left[y_1^{(k)},\ \ y_2^{(k)},\ \ \cdots,\ \ y_i^{(k)},\ \ \cdots,\ \ y_L^{(k)}\right], \qquad i \in \{1,..,L\} \text{ and } k \in \{1,..,N\}$$

A useful property of a profile is its total symbol size B defined as:

$$B^{(k)} = y_1^{(k)} + y_2^{(k)} + \cdots + y_L^{(k)}$$

The problem at hand is to maximize the OFDM interface capacity. Following the convention in CableLabs seminal paper on PMA [1], the capacity is defined as:

$$C = S \times \left(\sum_{k=1}^{N} \frac{R^{(k)}}{B^{(k)}}\right)^{-1},$$

where S is the symbol rate (fixed to 25 or 50 Ksym/s), $B^{(k)}$ is the total symbol size for profile k, and $R^{(k)}$ is the ratio of number of devices assigned to profile k. The optimization problem has several constraints that will be discussed in this paper. The most basic constraint is that for any device j assigned to a profile k, the MER of the device across the spectrum must support the chosen modulation efficiency values. Let's assume that the MER thresholds for assigning modulation efficiency values in the downstream are represented by the mappings listed in Table 1 (obtained from the D3.1 specification [2]).

Table 1 - Minimum MER values that support the corresponding modulation.

MER Threshold (dB) ($x_{threshold}$)	Modulation efficiency (y)
0	0
9	2
15	4
21	6
24	7
27	8
30.5	9
34	10
37	11
41	12
46	13
52	14

Then the constraint above is mathematically expressed as:

$$\forall x_i^{(j)},\ x_i^{(j)} \geq x_{threshold} \text{ for the assigned } y_i^{(k)}$$

The statement of the problem so far is naïve and is used for illustrative purposes. In reality, the thresholds listed in the table are not hard thresholds but rather recommendations that have to do with keeping the symbol error rate below a certain level. Revisiting the thresholds and allowing some subcarriers to operate above their recommended modulation efficiency values should be permissible as long as it does not result in generation of uncorrectable errors and packet loss above some desired value (refer to Section 8: Future Work for a discussion of how we plan to integrate error rates into PMA).

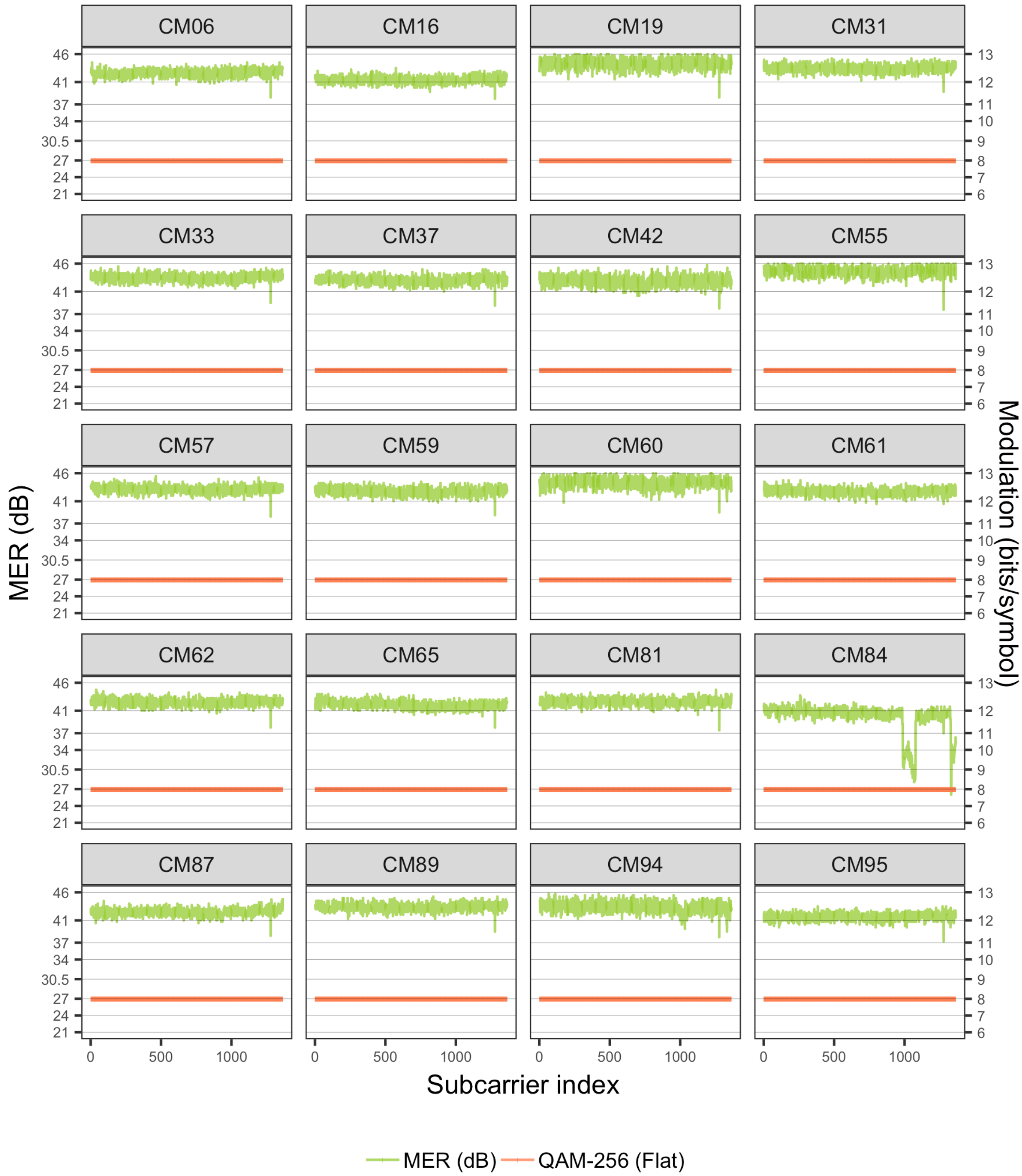


Figure 6 - MER measurements for a group of 20 CMs shown on a dual y-axis plot. The left y-axis indicates the MER value in dB and the right y-axis the corresponding modulation level (bits/symbol). In the current configuration, all devices use a QAM-256 modulation across the entire OFDM spectrum (1,364 subcarriers in total). It is clear from the shape and level of the MER curves that most devices benefit from using higher modulation while one device (CM84) exhibits impairments (possibly in two regions of the spectrum). Note that the device identifiers shown in the headers of panels are anonymized versions of the MAC addresses of devices.

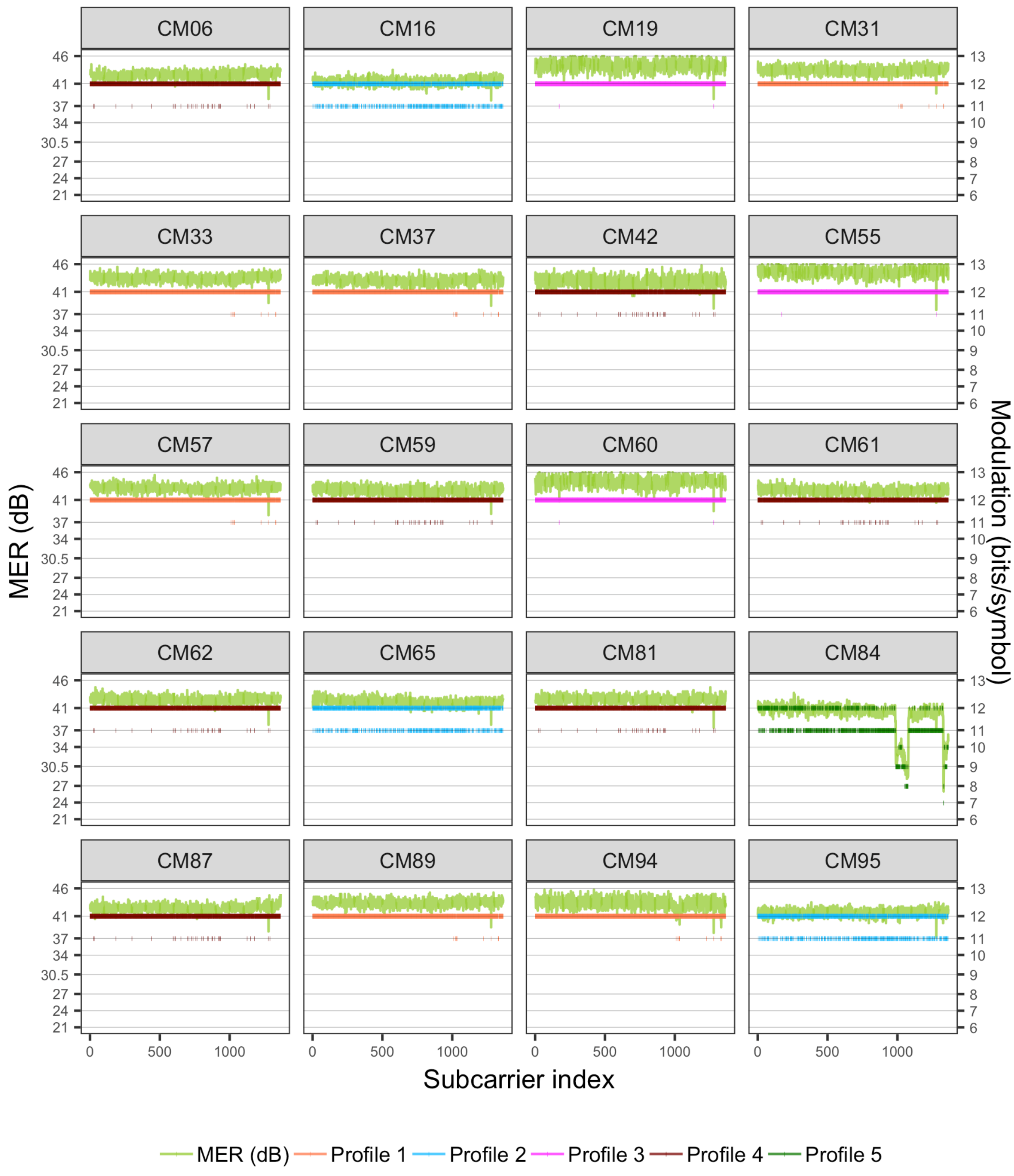


Figure 7 - Each of the same 20 CMs is now assigned a modulation profile from a pool of a total 5 profiles. The profiles were algorithmically constructed to maximize the interface capacity. Notice that the impaired device (CM84) gets assigned its own unique profile (Profile # 5), whereas the rest of the devices share the remaining profiles. The control profile (Profile #0), which is common to all devices, is not shown on this plot.

Following this illustrative representation of the problem, one may come up with a suitable algorithm to construct profiles and assign devices appropriately. For example, Figure 7 shows the same group of 20 modems, assigned to 5 profiles yielding ~42% gain in capacity over a flat QAM-256 (benchmark) configuration. The chosen profiles also highlight the problem with treating the values in Table 1 as hard decision boundaries. Notice that Profile #2 in Figure 7 fluctuates between modulation values of 11 and 12 bits/symbol. The observed fluctuation is due to MER curves for the devices assigned to Profile #2, tracking the 41 dB decision boundary line. This scenario is not uncommon, and highlights that basing the construction of profiles on single point-in-time MER likely leads to unstable profiles that follow the inherent noise in the MER measurement.

The mechanics of the algorithm used to construct the profiles shown in Figure 7 will be discussed in a later section (Core Algorithm). But before diving into the details of the algorithm, we'll argue next that the problem is far more challenging than this illustration due to the following factors:

- **The time dimension**: MER measurements have inherent noise; a single-point-in-time MER curve does not capture the "true" but rather the hidden state of the S/N for a given device. The variance in MER should be addressed by the algorithm. In addition to the inherent noise in the measurement, true changes in S/N occur over time: Impairments may be introduced in the spectrum; impairments may be fixed; some may be seasonal, weather, and/or temperature related; some effects may transiently show up and disappear. These factors should be considered to ensure that the recommended profiles function properly and do not become outdated/obsolete shortly following their application.

- **Vendor constraints**: CMTS vendors introduce a host of additional constraints to the optimization problem, depending on their implementation of the D3.1 specification. In addition to the variation in supported profiles per OFDM interface across CMTS make/models, some of the encountered constraints include the following:
    - Limit on the number of profiles per CMTS. While the optimization problem is defined at the OFDM interface level, we've encountered additional constraints on the total number of unique profiles across the CMTS.
    - Limit on the number of segments within a profile, where a segment is defined as a contiguous block of subcarriers assigned the same modulation efficiency value.
    - Requiring that segment width be a multiple of some fixed frequency value. Example, a segment width that is to be a multiple of 1 MHz translates into assigning the same modulation value for each group of 20 subcarriers (assuming 50 KHz subcarrier width).
    - Requiring that the absolute frequency value at which a segment block starts be divisible by a certain value representing the CMTS "grid spacing". Example, if the grid spacing is 250 KHz and the start frequency is 700.1 MHz, then the first 3 subcarriers are skipped or the segment starts in the excluded subcarrier guard band; resulting in the segment starting at 700.25 MHz to conform with this constraint.
    - Allowing a subset of QAM modulation values from the list in Table 1. Example, certain CMTS models support only square QAM constellations (even number modulation efficiency values).

  These are just few examples of encountered constraints. They do not apply across all CMTS makes and models and are not an exhaustive list. To deal with this challenge, the end-to-end data science pipeline built for managing profiles must be able to dynamically apply constraints depending on the specific make/model of the CMTS. At the same time, the list of known constraints should be curated and kept up-to-date to follow any CMTS hardware, firmware, and software development that may result in change to the constraints.

- **Policy considerations**: with a broadband network as large as Comcast's, managing the application of profiles goes beyond the pure data science problem. Maintaining a good degree of control over algorithmic recommendations is a must. Dictating parameters that directly influence the algorithm -- such as time and frequency aggregation statistic thresholds, applying overrides on the output, deciding how frequently profiles should be updated, monitoring performance of the program, and managing notifications & alerts -- are just few examples of the control capabilities that are a must for operationalizing PMA.

- **Forward Error Correction (FEC) consideration**: D3.1 uses a different error correction mechanism (LDPC) compared to D3.0 (Reed-Solomon). The recommended mappings between MER values and modulation efficiency values shown in Table 1 are not hard truths and will vary based on the statistics and nature of the noise and network linearity. Ultimately, the response of the system to the application of profiles is manifested in the rate of Corrected and Uncorrectable Codewords (CCER & CER). Therefore, FEC has to be measured and considered within the PMA program in some to-be-defined form.

With all this complexity in mind, we will venture next into describing an implementation of a PMA solution from a high-level "architectural" viewpoint.

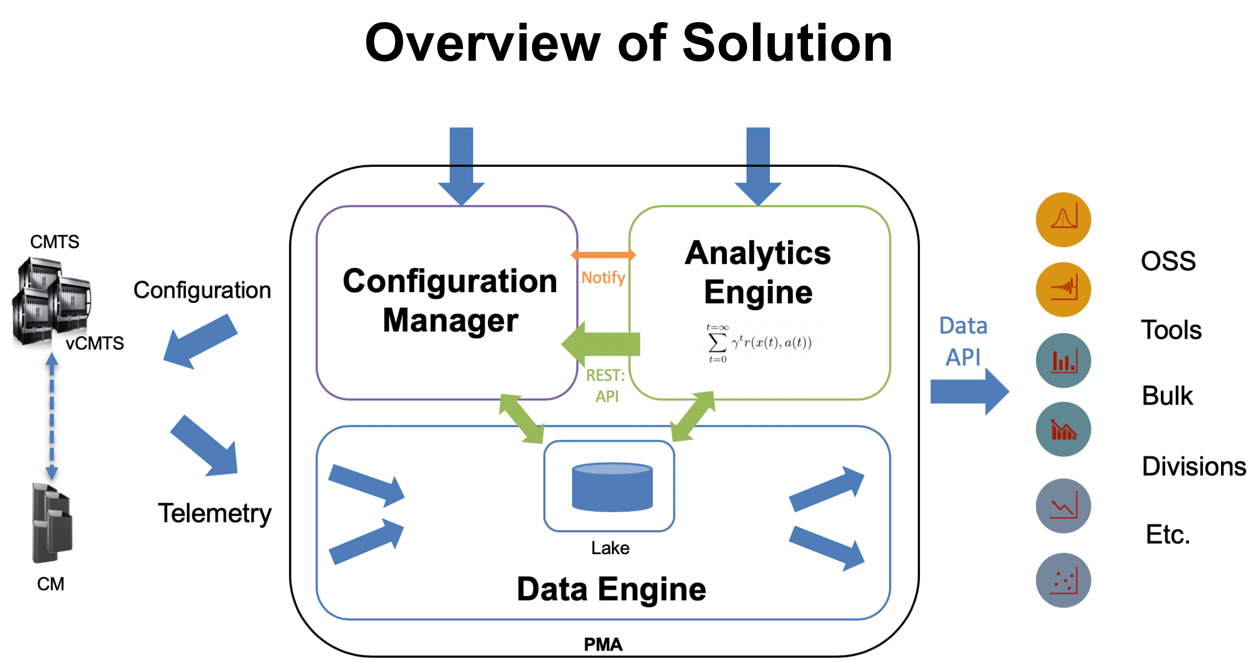


Figure 8 - Architecture view of the PMA systems. It includes 3 components: Data Engine, Analytics Engine, and Configuration Manager. These are described in the text below.

At a high level, the solution is comprised of the 3 systems shown in Figure 8. The focus of this paper is on the Analytics Engine (AE); but here's a brief description of the 3 systems:

1. **Data Engine**: Collects and maintains all data required for the creation and management of modulation profiles. Data includes:
   - **Network Topology**: connects a CM to an OFDM channel, MAC Domain, CMTS, and related Comcast Region & Division.

- **OFDM Channel Characteristics**: contains OFDM channel configuration such as channel width, subcarrier width, start frequency, active & excluded regions, position of PLC channel, and other configurations.
- **OFDM Subcarrier Configuration**: The modulation efficiency for each subcarrier and subcarrier type
- **CMTS Characteristics**: contains information on the make, model, hardware & software versions for each CMTS in the network.
- **Telemetry data**: includes MER, FEC, traffic and other network metrics for each CM and CMTS in the network.

Each of the different data sources is retrieved/updated with a certain frequency, depending on the dynamic nature of the data. For example, MER data from each device in the network may be acquired at n-hour intervals, while FEC data per profile is acquired at n-min intervals, with the topology acquired on yet a different frequency. The data architecture places the raw data in our data lake, with structure designed for ease of retrieval. The Data Engine has the responsibly of ensuring scalability of collection, performing validations to ensure integrity of the data, and managing retention policies. Note that the Data Engine is the foundation for D3.1 Proactive Network Maintenance (PNM) activities in addition to the PMA solution.

2. **Analytics Engine**: Consists of an end-to-end data science pipeline and is primarily responsible for the generation and recommendation of modulation profiles, based on the evolving conditions of the network. It retrieves the relevant raw data from the data lake, then cleans, aggregates, shapes, joins, and transforms the data. It further enables complex construction and invocation of the different algorithms for profile construction, validating the output, and generating useful metrics pertaining to the output. The Analytics Engine maintains its own data on policy and vendor related constraints. It also employs REST APIs for standard access to the recommended profiles and profile assignments as well as for managing constraints data and for making the output easily accessible by the end user. The Analytics Engine is described in more detail below.

3. **Configuration Manager**: The format of the output from the Analytics Engine is agnostic to the CMTS vendor make and model, i.e., it simply defines profiles as contiguous blocks of subcarriers, assigned specific modulation efficiencies, as well as which devices should be assigned to each of the defined profiles. The Configuration Manager is responsible for translating the raw output from the Analytics Engine to the CMTS-specific API and configuring the profiles with transactional integrity to the targeted CMTS/OFDM channel. Additionally, the Configuration Manager serves as a safeguard, conducting validation independent of the Analytics Engine, as well as checking the conditions and state of the CMTS. This is to make sure a profile update would not be disruptive to other activities, such as other configuration operations, 911 calls, CM registration status, and automatically opens and closes operational change management events. The Configuration Manager has the authority to reject or defer profiles recommended by the Analytics Engine based on policy. Finally, the Configuration Manager records the response of the CMTS to the application of profiles and can remediate or rollback, should the change cause adverse effects.

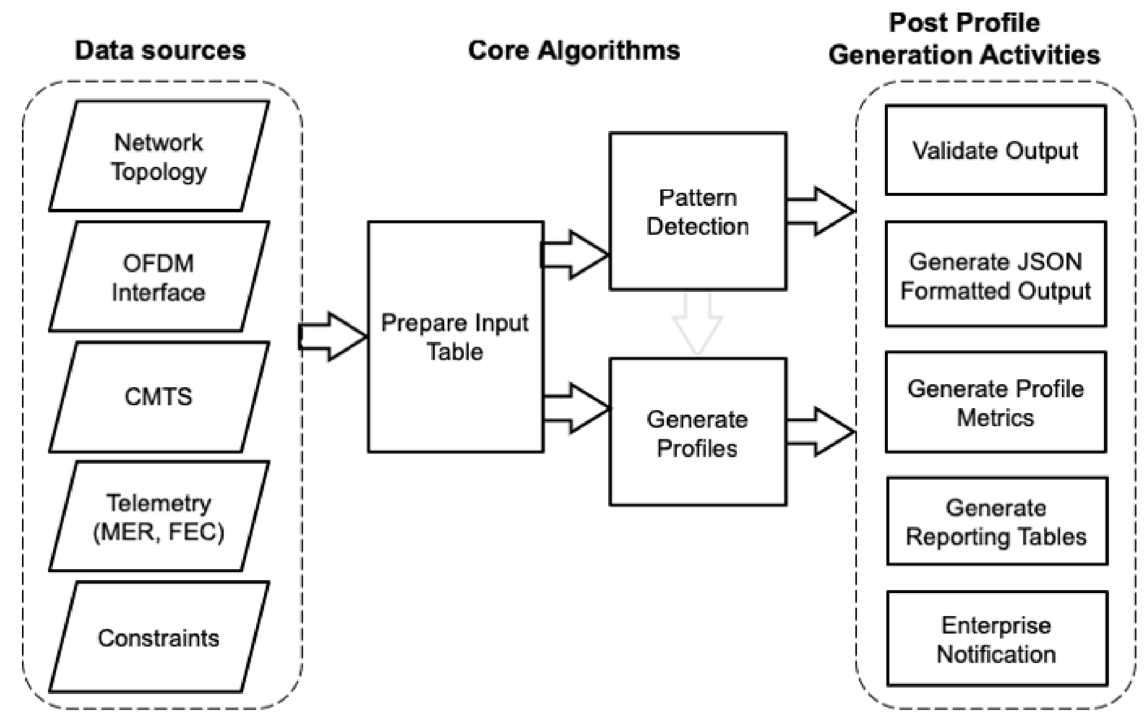


Figure 9 - Block diagram showing the Analytics Engine key data sources, core functions, and post generation activities. Note that Pattern Detection, discussed is a later section, includes a set of algorithms aimed at detecting ingress to ensure appropriate network maintenance.

Figure 9 shows the process flow within the Analytics Engine. The first step involves preparing an input data table that combines all needed information from the different data sources. The generated input table has observations (rows) corresponding to OFDM channels in the network and variables (columns) corresponding to features needed to generate profiles and metrics. The main features are:

- **Organizational Topology**: String representing location where CMTS is located
- **CMTS**: String representing the CMTS domain name
- **OFDM Channel**: String representing OFDM interface (includes port, card, and slot numbers)
- **Devices**: Array of strings containing list of mac addresses of devices attached to the interface
- **MER**: Array of an array of floating numbers, where each inner array contains the corresponding CM's MER vector
- **Constraints**: Data structure (key-value pairs) containing all vendor-related and global policy constraints

In addition to these features, characteristics of the ODFM channel are also included. The input data is used by two processes running asynchronously: Pattern detection and Profile generation. The latter includes the core algorithms for construction of suitable modulation profiles, given the MER characteristics of the devices and the corresponding CMTS constraints. Following the generation of profiles, several processes run in parallel:

- Profiles are validated against the constraints to ensure that the algorithm indeed generated valid profiles. For example, if a certain CMTS make/model allows a maximum of 5 segments and a generated profile

is found to contain segments exceeding this number, the violation is recorded, and the profile is flagged as invalid.
- The raw output describing profiles and device assignment is formatted in accordance to an agreed upon JSON data structure.
- Metrics for the specific profile generation run are calculated. These include the OFDM interface capacity, segmentation impact, various counts of the number of interfaces, profiles, devices, etc.
- Aggregated reporting tables are constructed for metrics of interest (e.g. the total capacity gain over QAM-256 by organizational topology). The main purpose of the reporting tables is to support operations and deployment of dashboards.

The entire pipeline presented in Figure 9 was implemented and run in the Apache Spark environment. A main orchestration function kicks off the pipeline and manages invocation of the other functions. All form of output produced by the pipeline is saved into the data lake under unique run IDs, which identify the specific pipeline run for system traceability.

MER & Time

MER data for D3.1 devices in Comcast's network is polled at frequent intervals and used to conduct exploratory analysis to understand and quantify the amount of variability of MER, the effect of the time dimension on the stability of profiles and, most importantly, to turn these insights into actions that inform how MER data is to be processed by the Analytics Engine. Insights from the analysis are shared in this section.

First, we considered the variation in MER on a subcarrier level over a period of ~1 week during which MER is sampled at ~n hour intervals. The analysis covered a single CMTS containing 29 OFDM channels serving DOCSIS devices. The outcome of this analysis is presented in the form of the boxplots shown in Figure 10. Each boxplot corresponds to the distribution of the range (max-min) of MER variation across all combinations of devices and subcarriers in the corresponding channel. The boxplots show that the variation in MER is around 3 dB (median represented as vertical bold line in boxplots). Though there are outliers (data points in the boxplots) extending as far as >20 dB. These outliers indicate highly unstable subcarriers within the considered 1-week period. The impact of the variation of MER on modulation efficiency assignment was further investigated. Figure 11 shows that by the first day, the ratio of subcarriers that retain the modulation efficiency assignment, based on their initial MER point-in-time sample, dropped to ~0.83. After ~10 days, this value further dropped to ~0.75. The analysis suggests that fluctuations in MER values cause somewhat impactful changes early on, after the initial modulation efficiency assignment. Thus, the problem warrants some consideration of the underlying causes, as well as a strategy based on utilizing longitudinal samples rather than attempting to follow the changes in real time. Note that not operating PMA in real-time is an implementation choice we made; a choice that could be reconsidered once CMTS protocols achieve certain level of maturity that allows seamless switching between profiles. The dynamic nature of the channel and profile changes was one of the challenging stability challenges identified as part of the D3.1 hardening work.

The ~3 dB variation in MER represents a combination of inherent noise in the MER measurement (i.e. even without any material changes in quality of the spectrum, repeated measurements will show some variation) as well as slight fluctuations in the MER level. When investigating the causes for the larger variations, these were found to relate to impairments that appear intermittently, periodically, or that have an unstable character (e.g. they shift position in the spectrum or exhibit a change in their level and/or shape). An example of an impairment that causes large variation in the character of the MER curve is shown in Figure 12 for CM84, which is one of the 20 devices we've been following in this paper. Such varying MER presents a challenge to the generation and assignment of profiles. This is because each of the different MER samples presented in Figure 12, if considered separately, would suggest a different modulation profile for the device. Ideally, the fluctuation/instability in MER must be factored into the algorithm to maintain network stability, given knowledge of the MER history for devices in the network. Another example of the shortcomings of basing profile generation on a single-point-in-time MER is shown in Figure 13 For CM95. In this example, the device appears to be healthy overall. However, impairments do show up in 2 of the time samples.

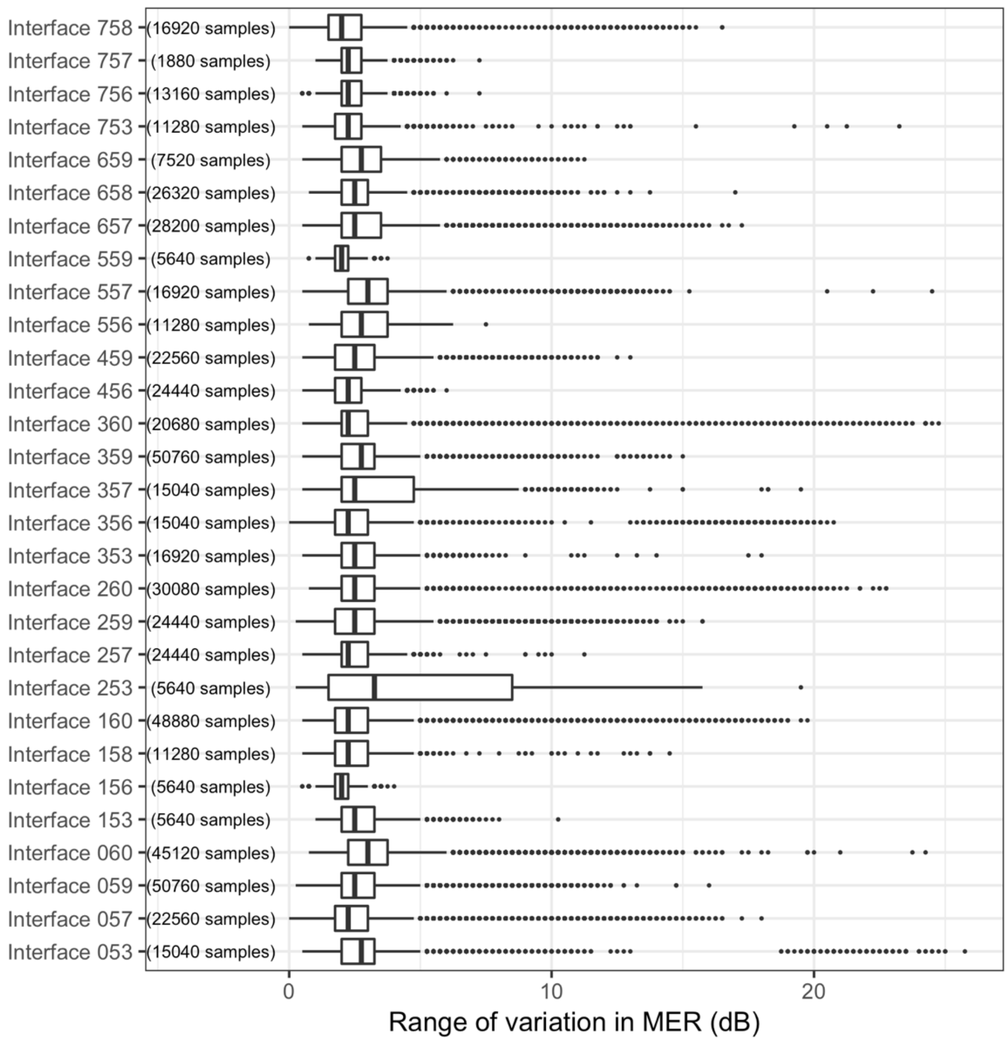


Figure 10 - Distributions of variation in MER over time by OFDM interface for one CMTS. Each boxplot represents the distribution of MER range (max – min) for the considered ~1-week time period for a population of device-subcarrier combinations. The number of samples represents the total number of devices multiplied by the number of frequency subcarriers in the OFDM spectrum. In each boxplot, the bold vertical line corresponds to the median, the rectangular box to the inter-quartile range (IQR), and the upper/lower whiskers to (Q3 + 1.5IQR)/(Q1 - 1.5IQR). Data points represent outliers.

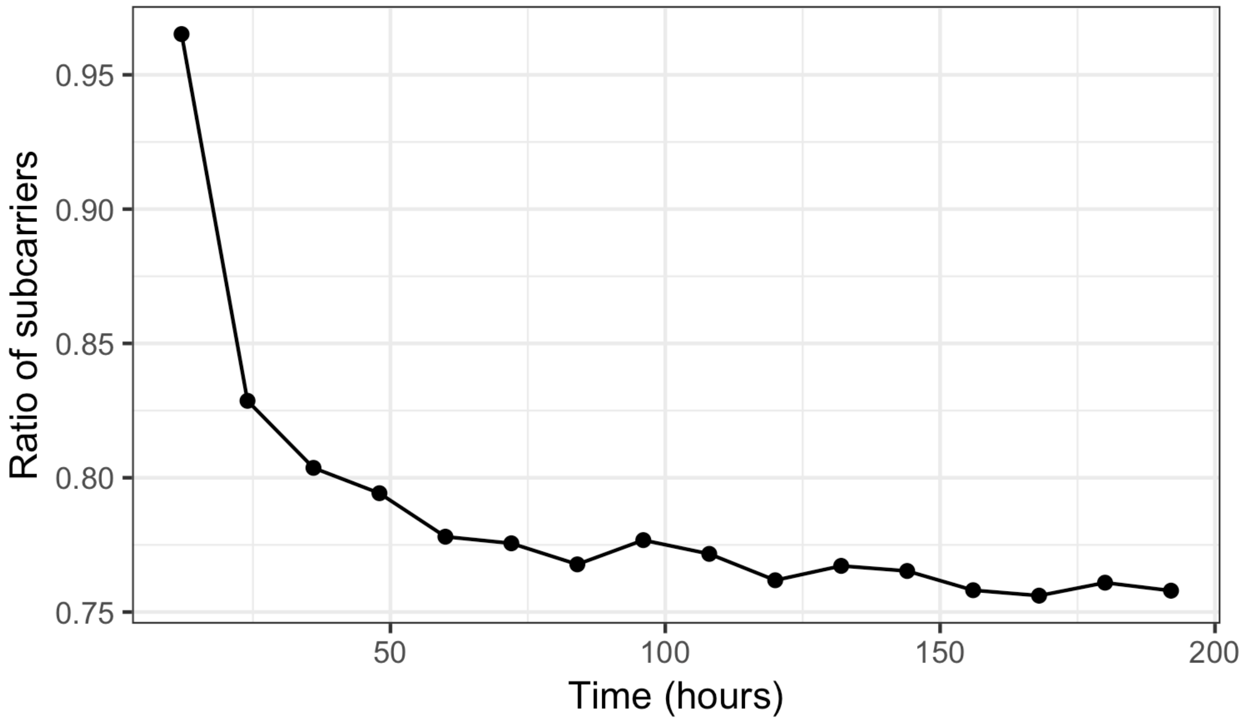


Figure 11 - Analysis showing the impact of MER variation on the stability of modulation efficiency assignments. The ratio of subcarriers that retain their original bitlaod assignment as their MER varies drops in time: to ~0.83 within one day and ~0.75 within 10 days.

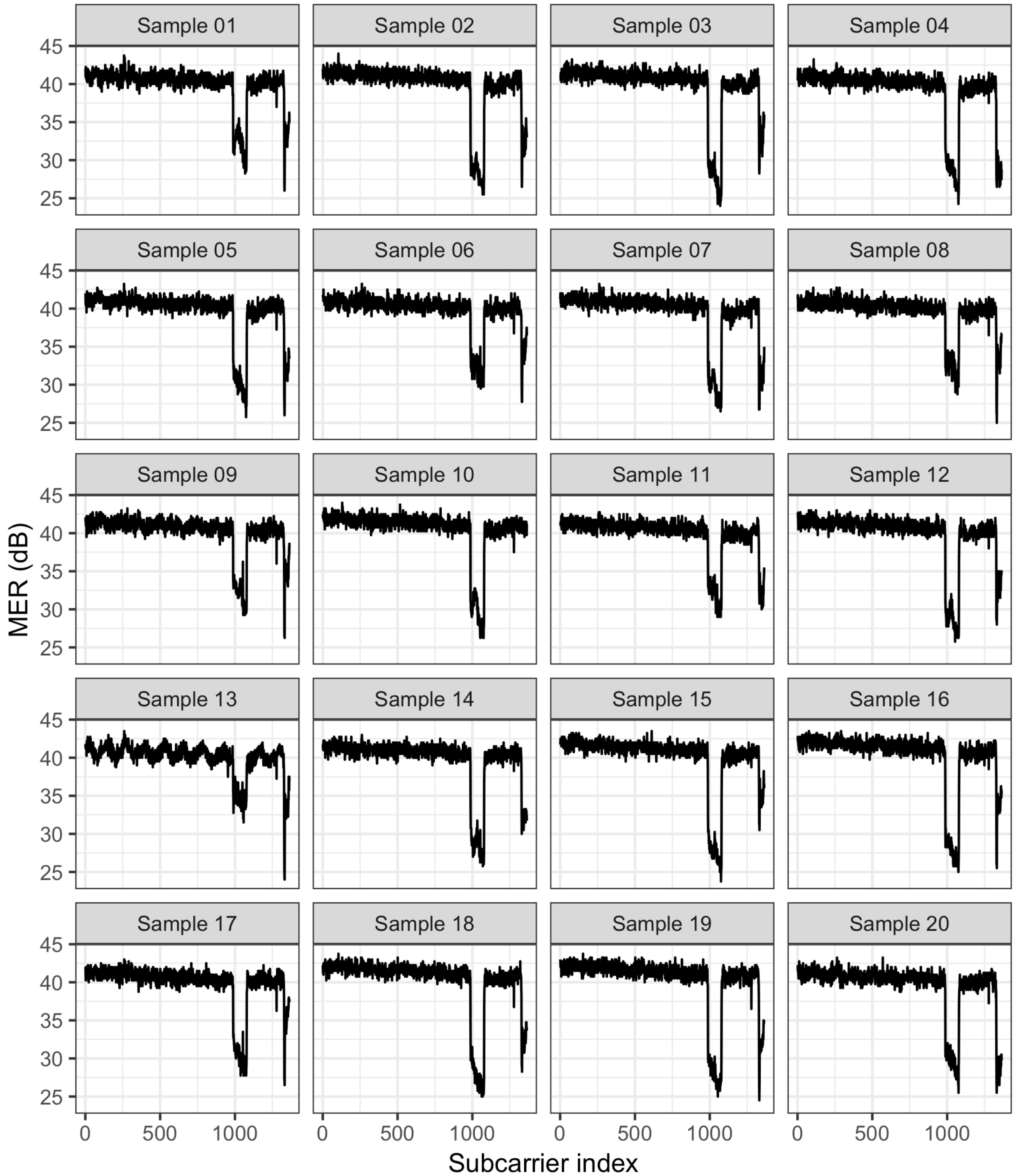


Figure 12 - MER time samples for CM84. The sampling period was ~n hours. CM84 exhibits instability in its MER curve due to impairment, with the level of MER in the impaired region changing over time. Certain time samples also show oscillatory shape in the MER curve outside the impaired region (e.g. sample #13).

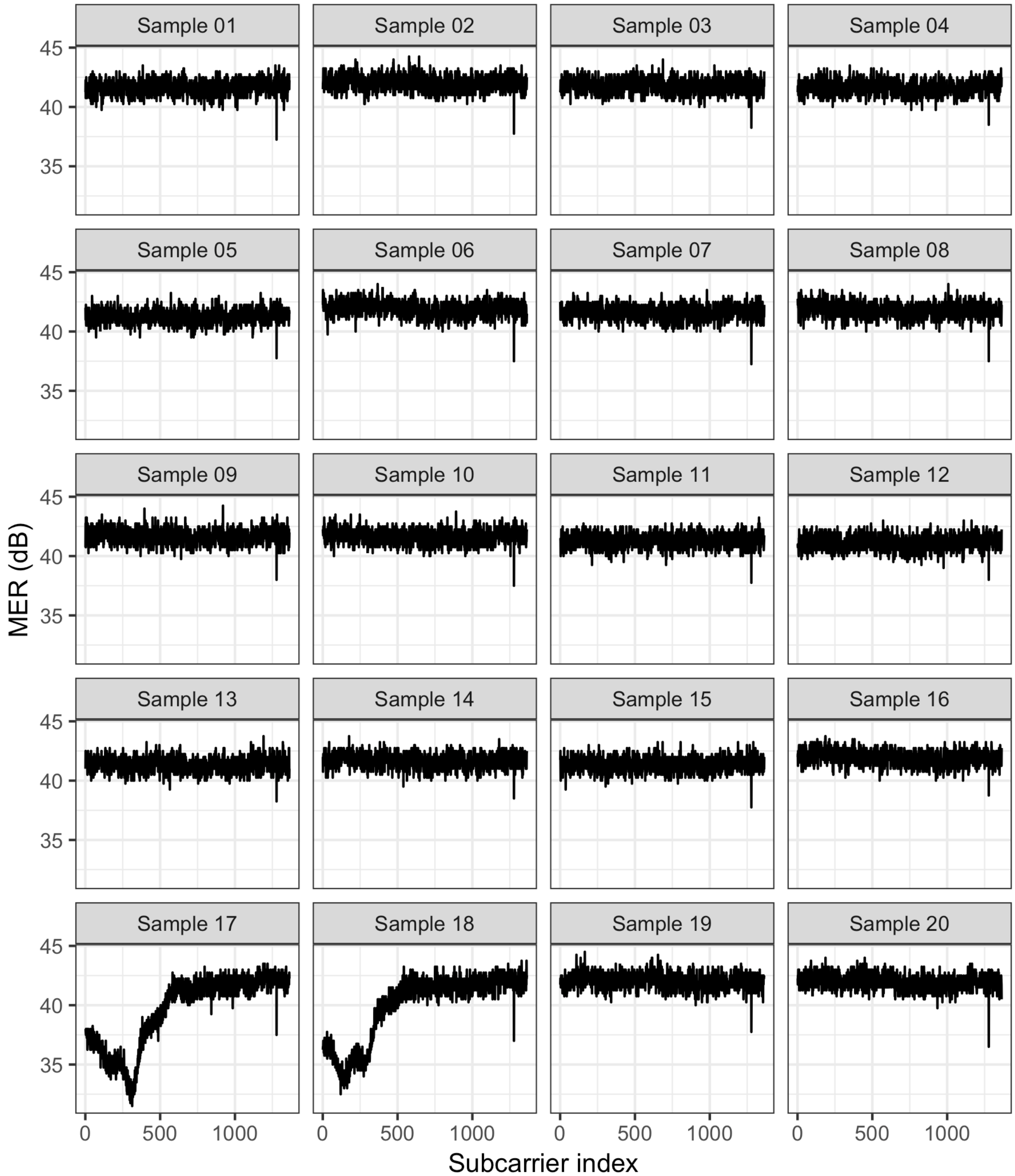


Figure 13 - MER time samples for CM95. The sampling period was ~n hours. CM95 shows healthy MER level overall, but an intermittent impairment appears in 2 of the time samples (#17 & #18). If profile generation was based on a single point-in-time, this impairment may be missed.

The MER instability was addressed by selecting the appropriate MER level for each device's subcarriers based on the collected time samples from history. The basic idea is to consider the measured MER curves for a device over a period of several days. With a ~n hour sampling interval, this should result in ~tens of MER samples collected per device. Thus, for each combination of device and subcarrier frequency there will be a distribution of MER values, from which the *n*th percentile is selected to represent the MER level for the device-subcarrier. The adopted method allows 3 tuning parameters that are specified through the pipeline global policy:

- The historical period considered for collecting MER time samples (e.g. last 10 days)
- The maximum number of samples to be considered per device (e.g. 100 samples)
- The value of *n*th percentile used for MER selection from the distribution (e.g. 10th percentile)

Controlling these parameters allows the selection of MER curves to vary on a spectrum ranging from the very conservative to the very aggressive. The conservative end corresponds to using small values for the *n*th percentile. At one extreme, picking the 0th percentile corresponds to the minimum value for selecting MER from each subcarrier distribution. With 0th percentile, we're effectively preparing for the worst observed state, given historical data. On the other end, large values for the *n*th percentile have the opposite effect, with the 100th percentile corresponding to the maximum value for selecting MER from each subcarrier distribution. Increasing the considered period also has an effect of adding some conservatism to the approach. This is because the likelihood of capturing irregularity with the device increases with increasing number of considered samples. The approach also allows the use of a single-point-in-time MER by setting the maximum number of samples to 1 (in this case, the *n*th percentile value becomes irrelevant). The maximum number of samples is currently set to a high enough value so not to influence the selection. Note that the default values are viewed as initial trial policy. For instance, the 10th percentile, while conservative enough, still allows for elimination of outliers (extremely low MER levels encountered). The 10-day period is viewed as an onboarding period, during which new devices and devices with recently fixed impairments are expected to demonstrate stable behavior before their "bad" history is erased/refreshed. The Analytics Engine pipeline allows experimenting with changing these tuning parameters and measuring the effect on the key performance metrics. These results are presented in later section, along with additional sensitivity analyses.

To visualize the processing of historical MER data, let's once again consider the same 20 devices introduced earlier. Figure 14 shows MER data collected for 10 days (thick green band) for the devices, along with the 10th percentile selection curve (solid gray line). Notice that an immediate benefit to using this approach is the reduced variance in the aggregated MER curve. This effect is expected, since the *n*th percentile selection has a similar effect to averaging, in terms of increasing the S/N of the measurement. The effect of increased

MER is also manifested in the less fluctuating levels in the profiles generated based on the aggregated MER values (Figure 15).

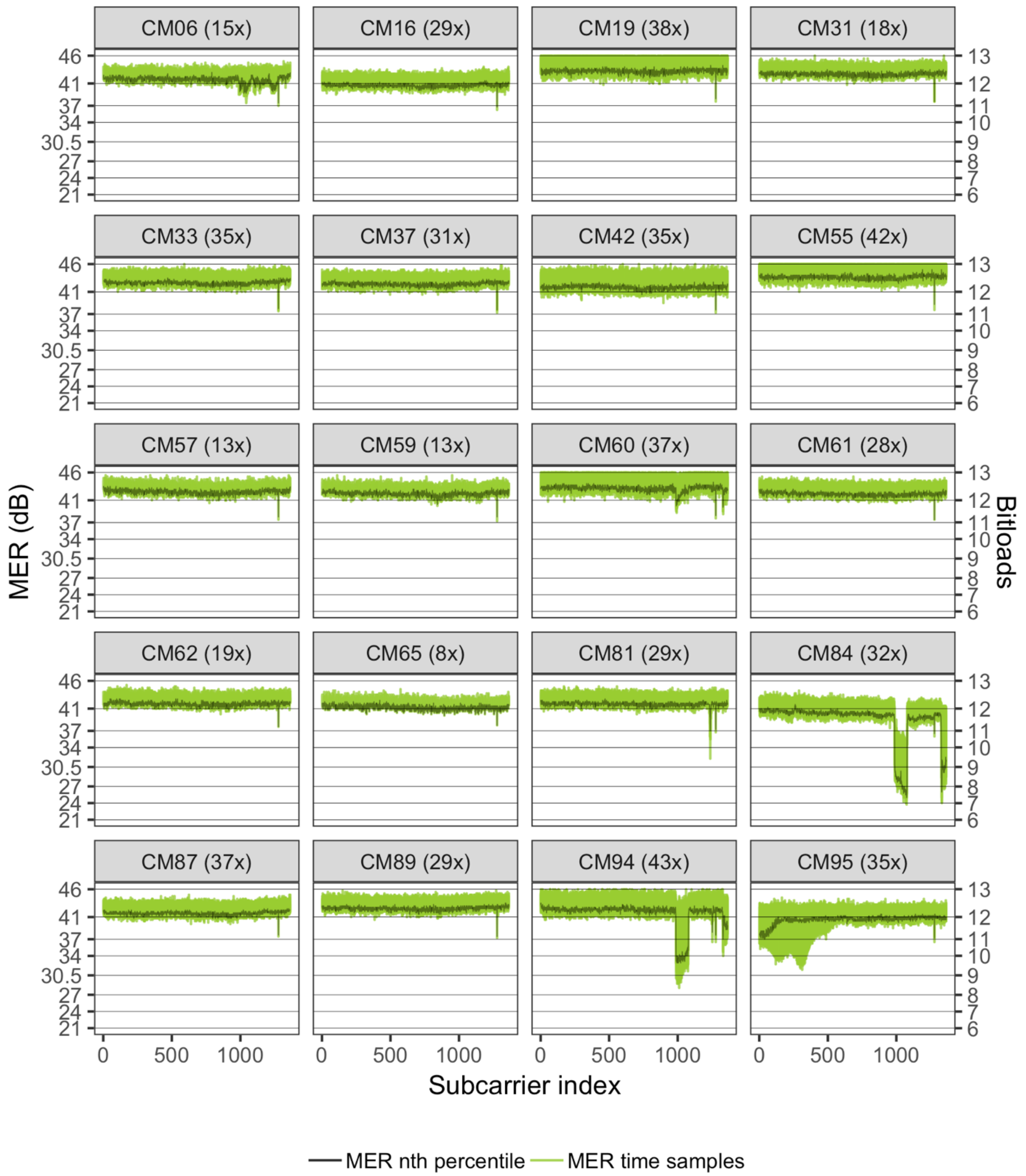


Figure 14 - MER time samples for the same 20 devices collected over a period of 10 days. The thick green band represents all MER time samples. The number of time samples varies by device and is indicated in each panel's header. The solid gray line represents an MER curve constructed by selecting the 10th percentile value for the distribution of time samples at each frequency subcarrier. Hence, the aggregate MER curve tracks the lower end of the green band.

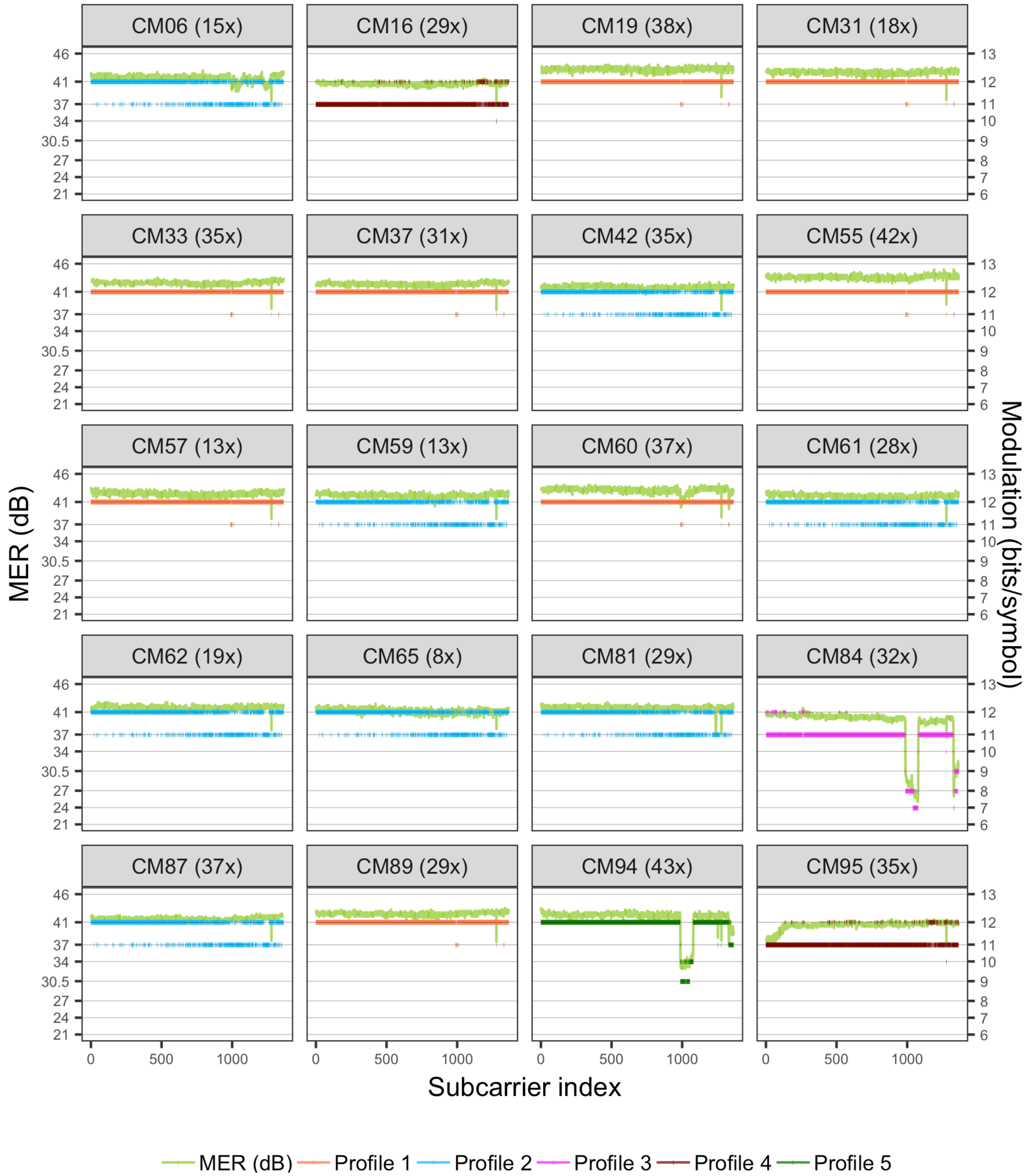


Figure 15 - The same 20 devices assigned profiles that are based on the aggregated MER values rather than a single point-in-time MER. These profiles are expected to be more stable because they consider the variability in MER measured over a period of ~days.

Core Algorithm

We adopted a *greedy* approach to the generation of profiles that optimizes OFDM channel capacity and satisfies given vendor and policy constraints. The main steps involved are the following:

1. Clustering of modems within an OFDM interface based on some objective function (e.g. the similarity of their MER curves)
2. Construction of profiles for the clustered modems within the interface
3. Segmentation (reshaping) of the profiles to satisfy the segment-related constraints
4. Pruning of the number of profiles to satisfy the maximum profiles per CMTS constraint

The ordering above represents the natural way to attack the problem: we start with the core profile generation, then reduce the number of segments, then (optionally) reduce the global number of profiles. The approach is greedy because it does not guarantee that the outcome yields the absolute optimum solution. Instead, the approach presented includes a combination of several steps, involving various computational techniques, ranging from machine learning algorithms to rule-based approaches. Choices made are in large part informed by the prior exploratory data analyses, and lab experiments. It is expected that the algorithm will continue to evolve and develop through rollout of future versions. Next is a description of each of those steps.

6. Clustering of Modems

The core profile generation algorithm uses hierarchical clustering [3] to group modems within an OFDM interface that share the same MER characteristics into the requested number of clusters. Deciding on which modems to group together proceeds iteratively: in each iteration, the two modems/clusters that have the most similar MER curves are merged together. Similarity is measured using several alternative objective functions. These include the Euclidian distance between the MER curves of two clusters and the Capacity Loss (i.e. the amount by which capacity of the OFDM interface would be reduced following merger of the two clusters). Once a merger occurs, the MER curve representative of the newly formed cluster is calculated on a subcarrier basis by taking the minimum of the MER levels of the two merged clusters along that subcarrier. That is, if cluster p was merged with cluster q to form cluster r, then the MER vector of the newly formed cluster is:

$$CM^{(r)} = \left[min\left(x_1^{(p)}, x_1^{(q)}\right) \quad min\left(x_2^{(p)}, x_2^{(q)}\right) \quad \cdots \quad min\left(x_i^{(p)}, x_i^{(q)}\right) \quad \cdots \quad min\left(x_L^{(p)}, x_L^{(q)}\right) \right]$$

The requested number of clusters corresponds to the number of profiles per OFDM channel allowed by the CMTS vendor minus one (as one profile is reserved for Profile 0). Once the requested number of clusters is reached, the iterative process terminates.

7. Modulation Efficiency Assignment

The MER curve for each cluster is mapped into modulation efficiency values based on the mapping in Table 1. No consideration for segment-related constraints is made at this point. Profiles are thus generated, but they are most likely "invalid" from the perspective of the CMTS.

8. Segmentation

The two nontrivial segment-related constraints are the segment width and the maximum number of segments. An illustrative example for the segmentation process is shown in Figure 16. The first constraint is addressed by binning the modulation efficiency vector such that every *n* subcarriers get assigned the same modulation efficiency value—the median of the *n* modulation efficiencies, with *n* being a parameter that is determined by the vendor constraints data (steps a-c). Next, the number of segments is reduced in accordance with the vendor-related constraint on maximum number of segments. This is done by running an aggregation (smoothing) function on the modulation efficiency vector within an initial window size of 3. Each value at the center of the window is replaced by the median value for the window. The window size is increased iteratively until the segments are reduced to the set limit (step d). Once the required number of segments is reached, the modulation efficiency values are expanded by replicating each value *n* times (step e).

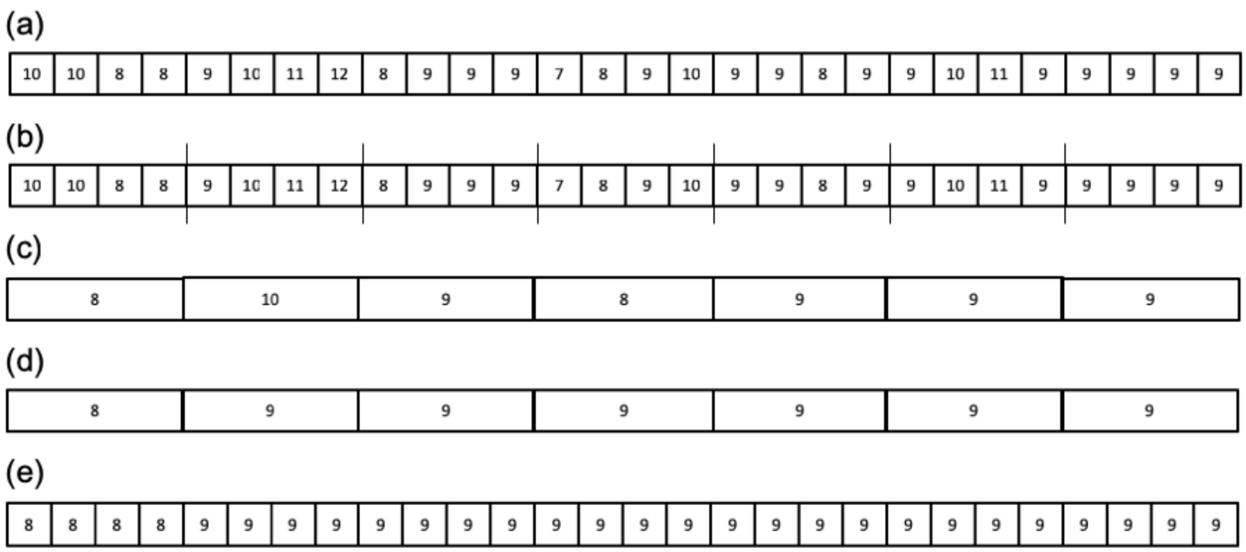


Figure 16 - Illustration of the segmentation process. (a) The starting profile has 18 segments. The (fictitious) constraints require a maximum of 3 segments and a segment width that is a multiple of 4 subcarriers. (b) Subcarriers are divided into blocks of 4 subcarriers. (c) Each block is assigned the median value. When there's a tie, the lower modulation efficiency value is used. (d) A moving median window of width 3 is applied, resulting in a reduction of the number of segments to 2. (e) The binned modulation efficiencies are exploded to retain the original length of the profile.

Figure 17 shows the 5 profiles generated for the sample 20 devices, both before and after segmentation. Note that the segmentation procedure outlined here may result in devices operating above/below the recommended modulation efficiency value, according to the mapping in Table 1 for parts of the spectrum. We permitted this violation of the recommended mapping to occur for a couple of reasons. First, the choice of 10th percentile and large MER consideration period already includes conservatism in choice of MER values from time samples. Second, we believe that this effect is minimal because MER levels between neighboring subcarriers are correlated (e.g. impairments tend to affect a chunk of contiguous subcarriers, not randomly positioned subcarriers). We tested this proposition by tracking the ratio of profile subcarriers operating below, at, and above their recommended level. The result in Figure 18 shows that ~90% of subcarriers across the whole network are assigned their appropriate modulation level after segmentation.

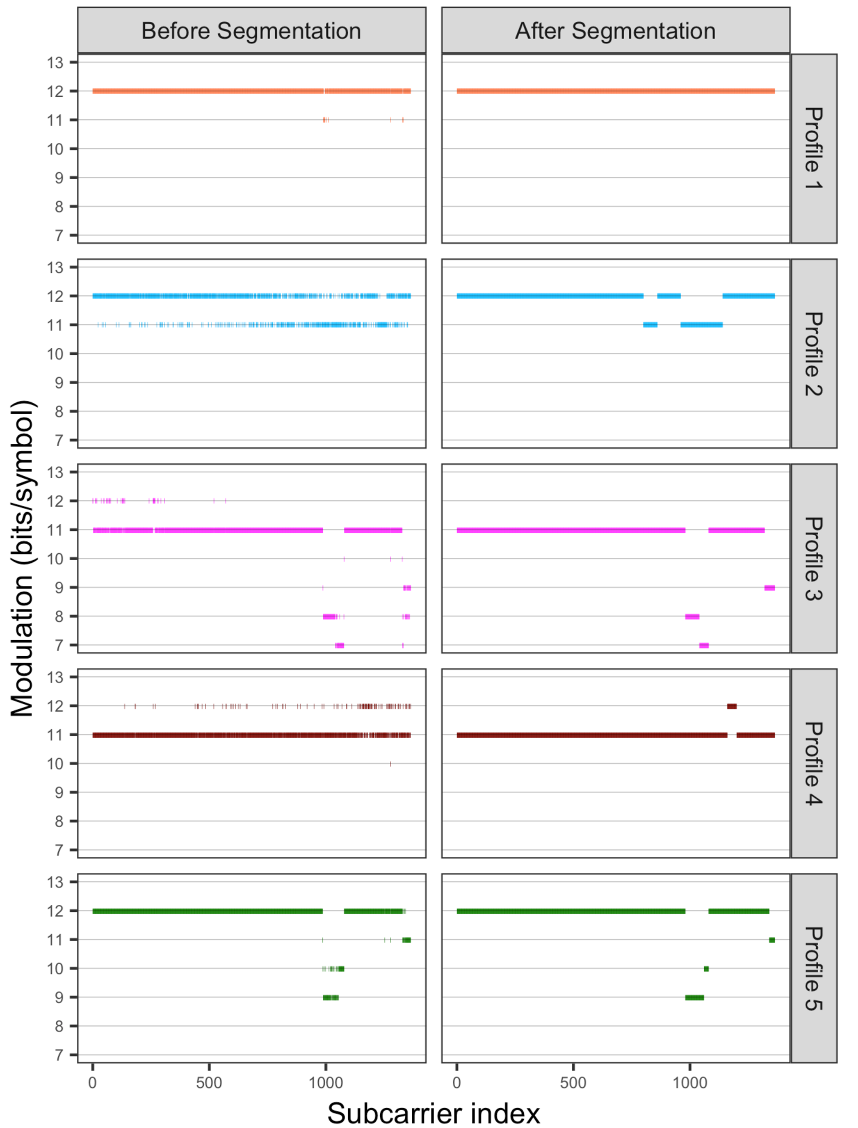


Figure 17 - Example illustrating segmenting profiles to satisfy vendor constraints. Here, the profiles for the 20 example devices were segmented such that the maximum number of segments is 5 and each segment width is a multiple of 20 subcarriers.

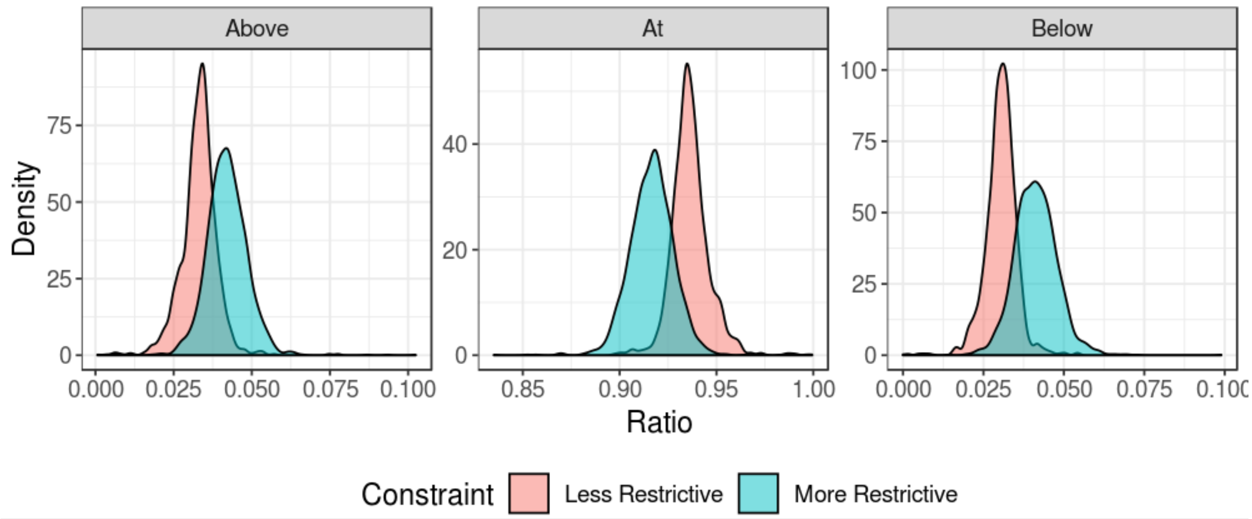


Figure 18 - Impact of segmentation on reassignment of bitloads as measured across Comcast's full D3.1 network. The analysis shows the impact of segmentation for two hypothetical CMTS vendors: one with more restrictive constraint and one with less restrictive constraint. Overall 90% of the subcarriers keep the modulation efficiency value assigned according to the mapping in Table 1. Less than 5% operate at level higher (lower) than the level recommended for their MER.

We also confirmed that the segmentation procedure does not have a negative or positive impact on interface capacity. This is because the aggregation algorithm described above is, in effect, a smoothing operation: On balance, the total sum of bitloads across the spectrum does not change when comparing the values before and after segmentation.

9. Profile Consolidation (Pruning)

Profiles are consolidated to satisfy the global limit on the number of profiles within the CMTS. This step is invoked only if the CMTS vendor limits the total number of profiles within the CMTS. It uses the exact same algorithm for clustering of modems. The end result is the pruning of profiles to the limited number, such that certain profiles are reused across multiple OFDM channels.

We examined the effect of pruning on the capacity gain for one CMTS. The results are shown in Figure 19. The study shows that pruning slightly reduces capacity gain. More interestingly, this analysis reveals the greedy nature of the algorithm: it does not pay to over-optimize in step 1 (clustering), given that profiles will have to be pruned at a later stage. For the CMTS used in this study, working with 8 to 10 profiles seems to be ideal without other limitations, although most CMTSs are limited to fewer than 6 profiles in the current implementation.

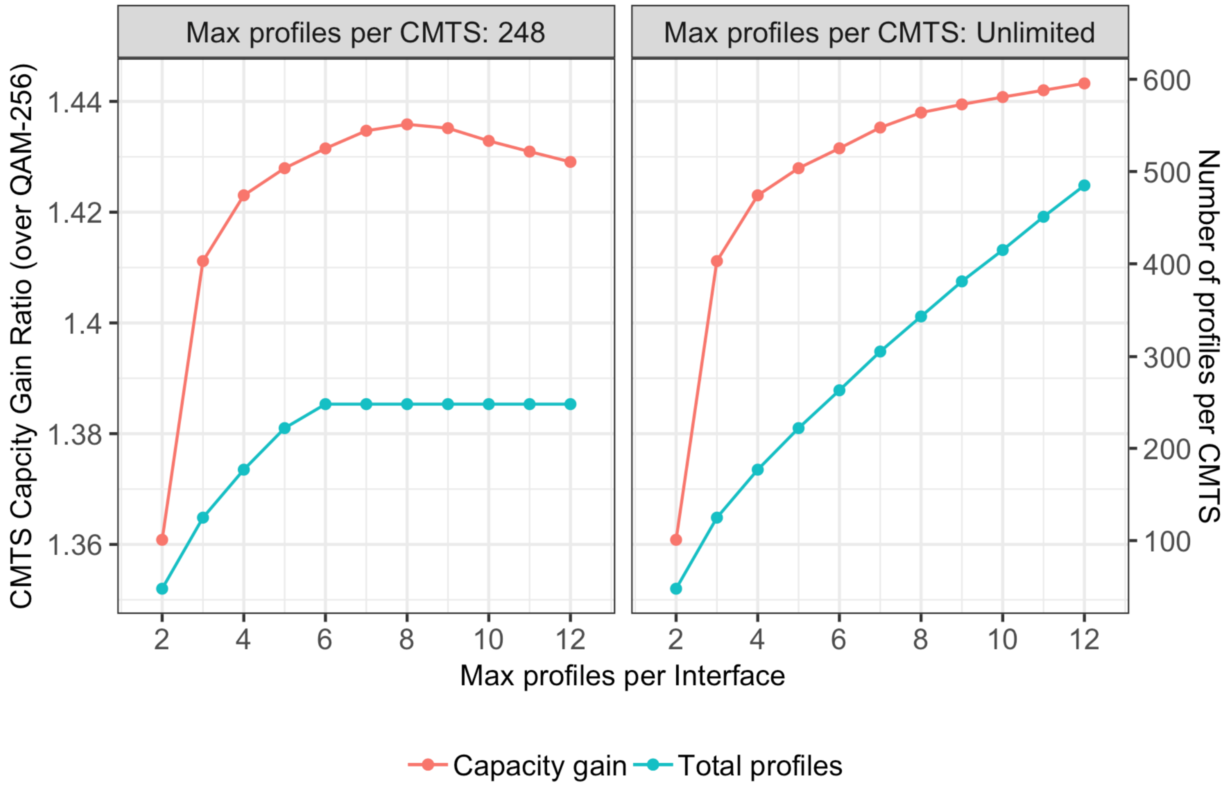


Figure 19 - Exploring capacity gain as the local (interface-level) and global (CMTS-level) constraints on the number of profiles are varied. The panel on the left shows capacity gain (left y-axis) and number of profiles (right y-axis) for the scenario where the constraint on global profiles is 248. The panel on the right shows the same metrics for the scenario where the global profiles are not constrained.

The algorithm as presented contains several tuning parameters that allow further optimization of the output and can be set through global policy. These are listed in Table 2.

Table 2 - List of Algorithm tuning parameters

Step	Hyperparameter
Processing MER Time Samples	Collection period; default value: n days
Processing MER Time Samples	Percentile selection value; default value: n%
Processing MER Time Samples	Maximum number of samples default value: n samples
Clustering	MER similarity function; default value: Capacity Loss
Clustering	Option to threshold the device MER before clustering; default value: Yes
Modulation efficiency	Global offset applied to threshold in MER mapping table; default value: n dB
Segmentation	Initial size of smoothing window; n

The effect of changing the hyper-parameters on the capacity was explored through various sensitivity analyses. One analysis is shown in Figure 20, where the MER selection percentile and the global decision boundary offset was changed. The analysis was conducted on the entire D3.1 footprint. Qualitatively, the observed trend in capacity gain is expected. A 3 dB offset (increased/decreased thresholds) can change the capacity gain by about 10% (decreased/increased capacity). However, note that the main tradeoff at play is the decrease/increase in uncorrectable error rates. We are currently working on measuring the actual response of the system as manifested in the FEC data.

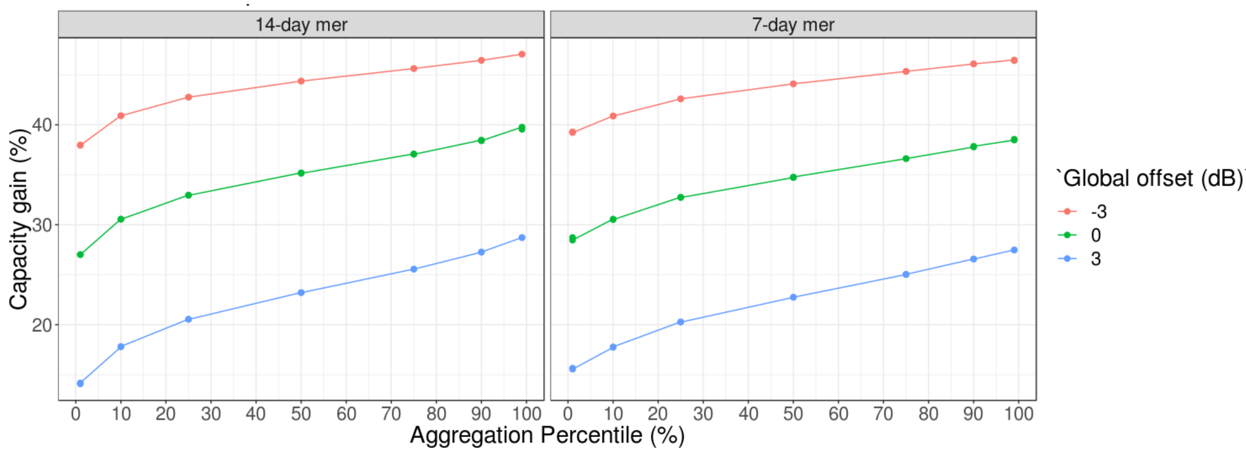


Figure 20 - Full footprint sensitivity analysis exploring change in capacity gain as the MER consideration period and MER selection percentile are varied. A 2-week consideration period (left) is mostly similar to a 1-week consideration period (right). The main difference is the extended y-axis range of the curves for the 2-week period as more extreme samples are encountered. In both panels, the global offset has the effect of shifting the curves up (reduced thresholds) or down (increased thresholds).

Based on the analysis above, we believe that a conservative estimate for implementing PMA will be >30% increase in capacity across Comcast's network. Future opportunities to increase by an incremental ~10% exist. The estimate corresponds to 2-week MER consideration period, 10^{th} percentile selection value, and 0 dB offset (i.e. keeping the mapping in Table 1 as is).

Lab Environment

We recognize the need for lab environments that can support CMTS feature testing and qualification, pre-deployment testing of the system, algorithm development, and automated testing for our planned Continuous Integration & Continuous Deployment (CI/CD) pipeline. To ensure good test coverage, our labs are configured to represent the most common combinations of CMTS and cable modem populations present in our deployed systems, rather than exhaustive testing of all possible combinations of CMTS and cable modems in use. From the perspective of feature and hardware support required by profile management, CMTS variants are further defined by particular downstream line card features and CMTS firmware versions.

At Comcast, we support two primary integrated CMTS solutions and our vCMTS (virtual-CMTS) solution. Our D3.1 modem population is a mix of Comcast provided gateways and customer-owned retail modems. Two primary cable modem chipsets are in use in the modem population. Here again, the firmware version further defines and constrains support for profile management type applications. Our Labs provide large groups of test modems, selected to achieve good test coverage of the numerous combinations of chipset, firmware, and retail brands.

Early in the development process, it was apparent that support for profile management related features in all of our CMTS platforms was quite different, and highly dependent on the actual hardware and software configuration of the devices. One of the first use cases for lab activity was focused on documenting profile management related hardware and software support for each CMTS variant, testing how the features worked, and verifying that the two cable modem SoCs were properly supporting the profile management features. For example, early in our D3.1 deployments, we experienced modem instability due to profile flapping, described earlier, which was traced, based on careful lab testing and analysis, to hold-off timers implemented differently in each CMTS and the CMs.

Additionally, we established two primary test labs, one of which is used for development related activities, and the other for pre-deployment verification of proper operation of our solution. To fully test the solution, each lab provides groups of modems with different RF channel impairments, in order to verify proper operation of the PMA solution. As of this writing, the development lab is configured with three groups of modems, two of which experience different impaired downstream channels, and the other being a control group with an unimpaired channel. Each of the labs has all CMTSs, test populations of modems, RF signal sources used to produce impairments, traffic generation, and the necessary RF plumbing that would ordinary be present in the HFC RF network.

To test any DOCSIS profile management solution, some means of generating realistic HFC channel impairments must be provided. From the data we collected, it was clear that the most common impairments our solution would encounter involved some sort of signal interference. As such, our first test beds focused primarily on duplicating interference sources and presenting them to the test network in a repeatable, controllable way. It was also clear that given the numerous types of interference scenarios we needed to test against, some basic automation of CMTS and test equipment would be required to efficiently configure the channel environment, run tests, and produce reports. Our automation solution is an ongoing work-in-progress, but the initial automation solution was based on simple Python scripts that manipulate the CMTS via the CLI interface. Automation of commercial test gear follows a similar design pattern: Python scripts with appropriate interface libraries to access the test gear (for example, using the VXI-11 standard). Software Defined Radio (SDR) solutions, which appear particularly promising, have been developed to be highly configurable through software control.

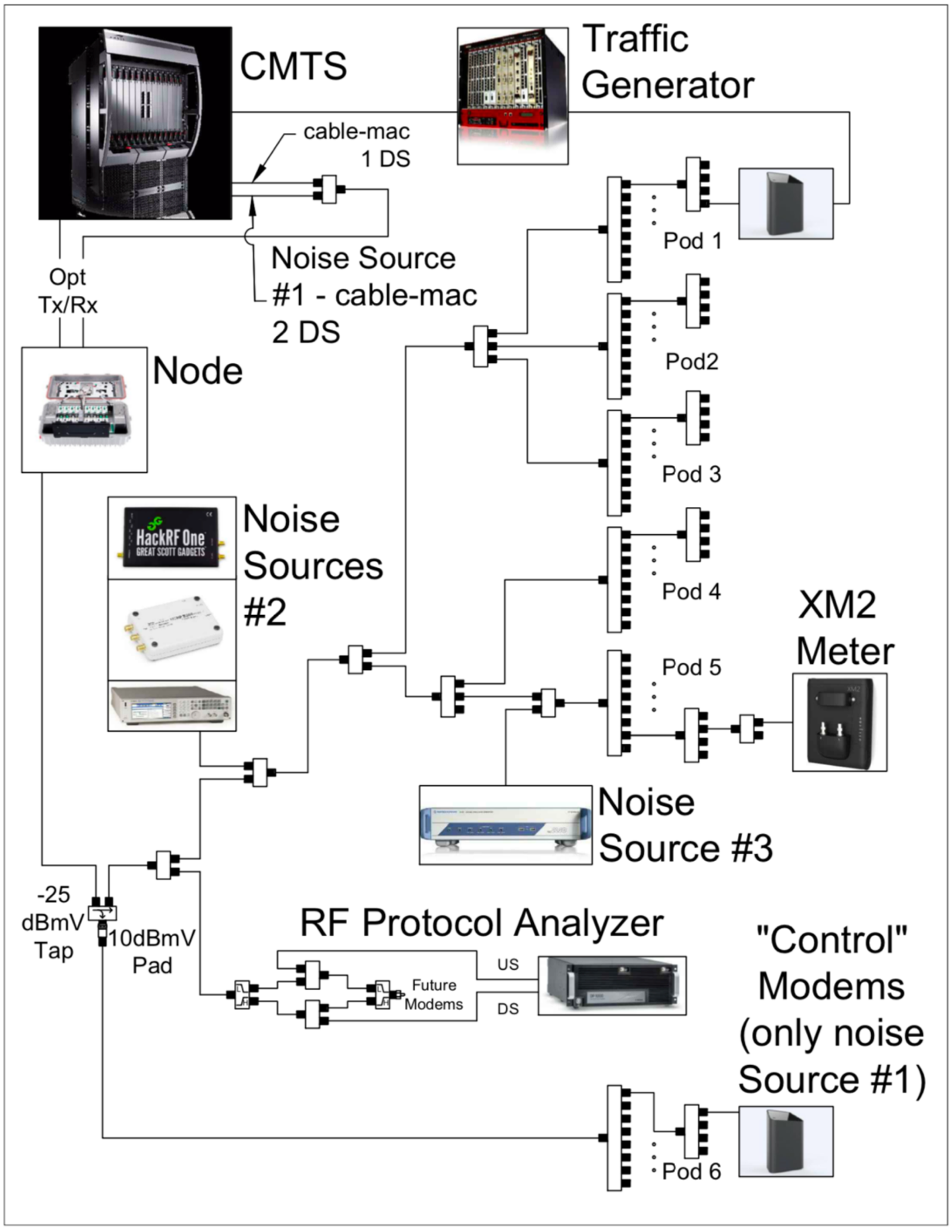


Figure 21 - Development lab RF connectivity showing our current development lab signal flow. Traffic flows are generated by an Ixia TG traffic generator; the three test populations of modems are shown on the right side of the diagram; the CMTS and optical node are shown on the left side of the diagram. Noise generation sources are connected as needed during the test processes, and an Averna DP-1000 is included for DOCSIS packet capture. The combining networks are designed in a way that isolates the impairments sources, to ensure the two impairment groups experience unique RF channel environments.

In the development lab (Figure 21), impairments are generated by commercial RF signal generators, unused RF output ports of the CMTS, and SDR devices. Unused CMTS RF ports were a first and convenient way to introduce easily-controlled impairments into the development system RF networks. One desirable feature of this method was to be able to use the same CMTS software drivers we developed for profile management to also turn on, turn off, and adjust levels of the simulated impairment source. Our lab signal generators are used for a multitude of projects, and tend to move between project teams, so using the CMTS RF ports for impairments virtually guaranteed access to a signal source when a particular test flow needed to be run, while use of the signal generators required scheduling of their use. The downside of using the CMTS RF ports for impairment signal generation is that the possible impairment waveforms is constrained to what the CMTS can produce -- either SC-QAM carriers of 6 MHz, or an OFDM carrier of varying width. For LTE-like pulse interference or noise, it was still necessary to use a commercial RF signal generator. For automation purposes, commercial RF signal generators vary in their support for remote control applications. Older generators simply have no means of programmatic control; others use older GPIB interfaces; and the more modern versions often rely on vendor-provided libraries, or require sometimes difficult-to-use frameworks or libraries. Usually, the add-on libraries require expensive software licenses to enable remote control of the device. One final concern with commercial RF generators is sometimes limited waveform memory, which in turn limits the time duration and/or bandwidth of the impairment signal that can be produced by the device.

One alternative to RF impairment generation that was developed is based on commercially available SDR technology. As the name implies, SDR generates waveforms using software, and streams the digital representation of the waveforms over USB or Ethernet to the SDR device. The SDR device converts the digital waveform back into an RF signal. The advantage of this technology is the intrinsically software-based approach to signal generation, which, in turn, meshes very well with our objective of full automation of test flow, and support of our CI/CD pipeline as it matures. In the sections below, we will discuss one novel method we are developing that allows us to reproduce, using SDR, actual impairment scenarios extracted from modem MER data.

Early on, as our data collection systems became operational and began accumulating MER data from our deployed D3.1 modems, it was clear that our modems were operating in channels impaired by locally generated (in-home) and plant-side LTE interference, misconfigured sweep generators, HFC suck-out, roll-off, old pilot carriers mistakenly left on, and a variety of other unusual scenarios, as described in the D3.1 Hardening section. Our profile assignment algorithm is designed to reduce modulation around such impairments, but one question arose: with such a diverse set of channel impairments present in our data, how do we exhaustively test and verify the PMA system will react properly in each of these impairment scenarios?

To generate such a diverse set of RF impairments, we have started down the path of using SDR techniques to generate a library of waveforms used to test our PMA. Initially, we have settled on a relatively low-cost unit named “HackRF”, and a slightly more expensive device from Ettus Research, the USRP B210. Both devices are capable of generating RF signals up to 6 GHz, but support different sampling rates in their analog-digital conversion (HackRF up to 20 mega-samples/sec, B210 up to 56 mega-samples/sec). The useful width of the generated impairments is constrained by the sample rate of the SDR and the data rate the associated Linux system is able to generate. In our current system, the B210 is operating at 38.4 mega-samples/sec and provides roughly 34 MHz of spectrum that is used to generate impairments in the downstream channel. The SDR hardware is driven from software around the popular open-source GNURadio software system. GNURadio offers an easy to work with Python/C++ programming environment and also a tool named GNURadioCompanion that allows construction of simple SDR applications built up from a library of common Digital Signal Processing blocks.

Figures 22 & 23 show a promising approach to developing realistic impairments, using MER data collected from cable modems operating in an impaired channel. The basic concept is to select a subset of subcarriers experiencing the impaired channel, performing an inverse Fast Fourier Transform (iFFT) on them, then streaming an impairment out of the SDR. The RF signal from the SDR is then combined with the pristine downstream signal from the CMTS, thereby producing the impairment observed on the original cable channel.

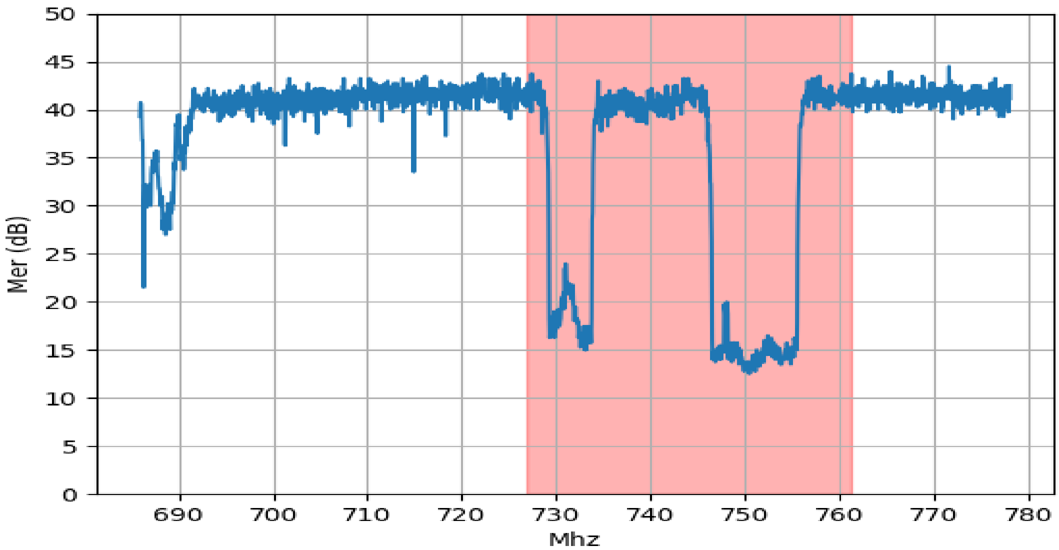


Figure 22 - MER data received from a D3.1 modem operating on our HFC plant. This plot shows two significant interference regions, likely from cellular LTE transmission in the 700 MHz band. The shared red region is the segment of spectrum the SDR impairment generator will duplicate and inject into the lab test network.

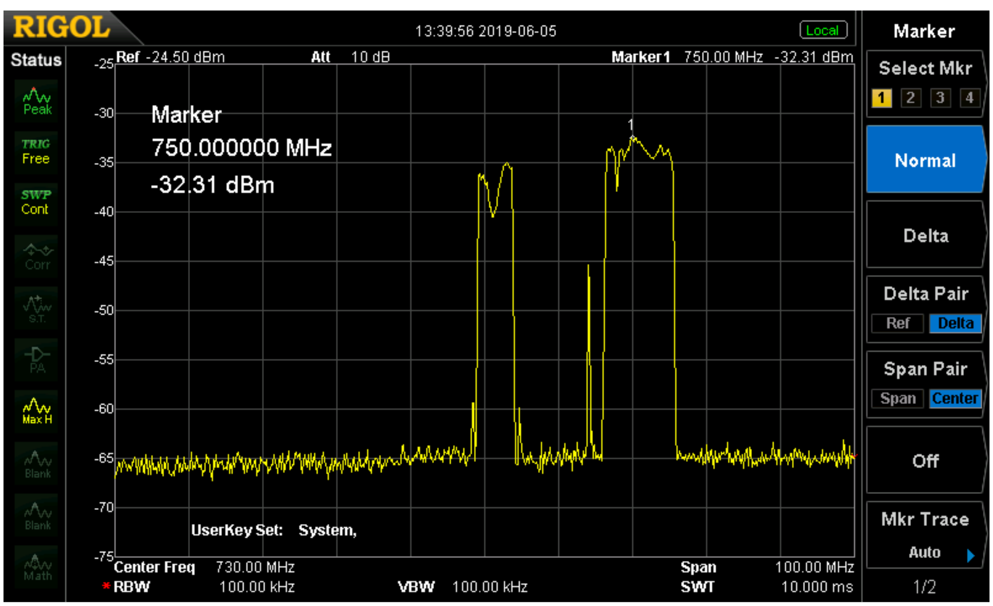


Figure 23 - Spectrum analyzer capture of the SDR generated spectrum based on MER data shown in Figure 22. The tone artifact at about 740 MHz is due to I-Q imbalance in the SDR and isn't considered an issue for this particular test. The SDR-generated signal would be added to the original, clean lab downstream OFDM carrier to produce the original MER response measured in the field.

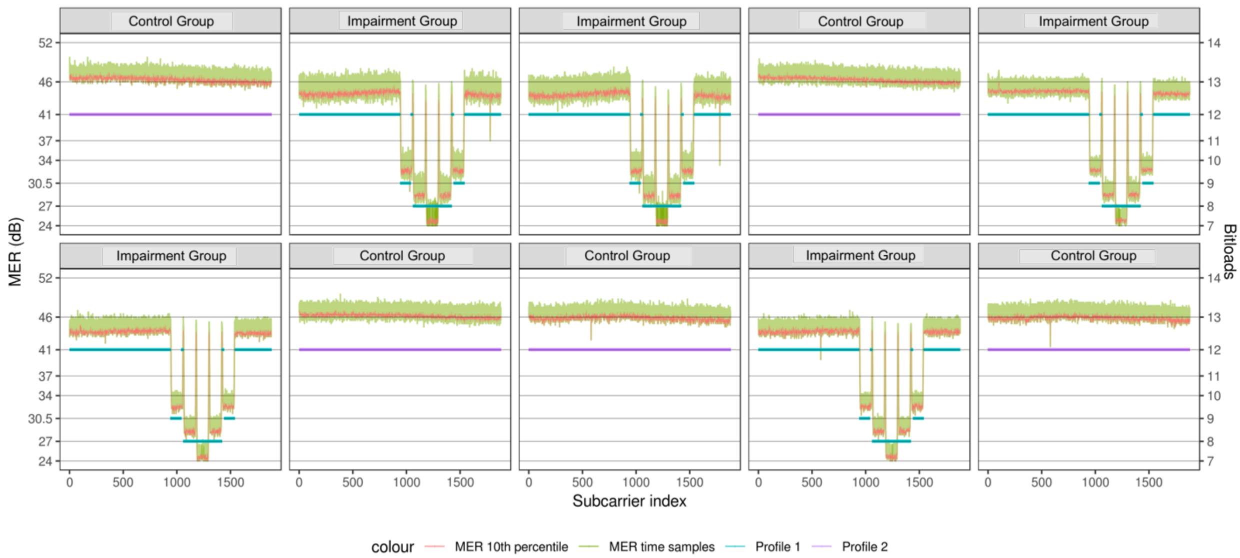


Figure 24 - Example of an impairment generated via SC-QAM carriers from an otherwise unused port on the CMTS. In this plot, the control group (unimpaired) modems are shown against modems operating in an impaired channel. The colored lines depict how the PMA solution is bit-loading around the impaired regions. Actual impairments observed on the operating HFC plant are not as cleanly defined in either time or frequency as this lab generated impairment is, but for functional testing, SC-QAM-generated impairments were an easy, first test case in our lab work. Time-based behavior of the impairment is simulated by turning on/off the SC-QAM carriers from automated scripts.

Another complication we encountered was attributable to the nature of how the profile management solution was architected. At Comcast, our D3.1 system is composed of over three thousand CMTS units, outputting more than one hundred thousand OFDM channels, received by millions of D3.1 cable modems. From the outset, we know our profile management solution needed to deal with a very large, and quickly growing cable modem delivery system. The system architecture embraced cloud-based techniques from the beginning, with the core concept that cloud-based architecture, with its inherent elasticity, would deliver the necessary performance, as our D3.1 delivery system grew. Our solution is structured around a closed loop concept where cable modem telemetry is continuously collected at scale, stored in cloud-based technology, and analyzed continuously to design appropriate profiles for each OFDM channel in our delivery system. However, for the purposes of testing, the architecture presented some challenges to our test workflow. We couldn't just configure the test bed, and invoke the profile management solution on demand, in a controlled way. We needed some way of scheduling test workflows around the normal operation of the pipeline.

The profile management solution can be thought of as a conveyor belt, with measurements from the cable modems entering on the left, data analyzed as the belt moves left to right, profiles emitted, and finally CMTS channels configured, once the profiles are available and the system determines that the latest profile materially improves the observed channel for the modems. The virtual conveyor belt turns continuously, asynchronous to test work flow. Additionally, the profiles are based (as discussed earlier in this paper) on data collected over a week or more of calendar time. So as we considered how to structure our test workflow, it was apparent that if we wanted to conduct the tests on the unaltered profile management pipeline, we would somehow have to compress a week or more of data collection into a much shorter period of time, such that multiple test cases could be managed each day. One concept we developed is called "spot-beaming". In the actual solution, the system collects MER data from the modems four times a day, and SNMP-based measurements are gathered every twenty minutes. With the spot-beaming solution, the collection infrastructure is configured for the test CMTS to sample at an increased rate of 1-2 minutes per data point, essentially compressing a calendar week worth of measurements into several hours. With the spot-beaming telemetry available, it was then possible to run tests using simple mechanisms such as Linux "cron" to manage individual test cases. From the perspective of the profile management system, the test workflow is simply modifying the channel environment, as would be encountered in the actual deployment of the system. Figure 24 shows example MER charts and profile assignment recommended by the Analytics Engine for a population of devices with lab-generated impairments. With the spot-beaming capability, lab efforts can cycle through many impairment scenarios on a daily basis to validate the implementation of PMA.

Pattern Detection

A PMA solution has proven to be very effective at managing both the network stability and network capacity. Comcast has also developed methods to reduce the impact on customer experience of network impairments very effectively [4]. The result can be temporary or less effective than desired, if the operational mitigation of network issues is not also considered. In turn, the full benefit and savings on operations metrics, like truck rolls and the customer experience, were not fully realized until the tools and metrics were used, to dispatch technician fix agents to implement network changes that resolved the problems. Networks tend to degrade over time, from an impairment perspective, if not maintained. When both network remediation and automated mitigation were applied, the customer experience improved, operations metrics reduced, network performance improved. The network maintenance was targeted and efficiently scheduled, and repeat truck rolls were avoided, providing significant value.

This PMA solution is following with the same dynamics as the aforementioned case study [4]. To learn from past lessons, pattern detection algorithms and other analytics are provided as a core part of the architecture. Events are delivered across a messaging bus to other Comcast OSS tools, to ensure that technicians are dispatched to the right hubs, network segments and homes, to remediate issues such as those identified within this paper. In addition to calculating the profiles to optimize customer experience, a pattern detection pipeline is required to ensure that care/fix agents are efficiently dispatched to remediate network challenges. As such, adaptive systems as PMA can easily mask the degradation of the network. The types of pattern detection already developed and under development include:

- Mobile wireless interference enriched with mobile carrier information
- PLC-related ingress in the spectrum, enriched with mobile carrier information
- Non-mobile wireless ingress, such as:
    - Incorrectly-configured sweep insertion points in the OFDM channel
    - Incorrect placement of sweep QAMs incorrectly within the OFDM spectrum
    - Other wireless ingress
- Analog Television non-linear distortion interference falling into the OFDM channel
- Linear impairments due to micro-reflections
- Profile recommendations with X% of subcarriers below Y bps/Hz modulation efficiency
- Recommendation for new PLC locations

Many of these impairments can be diagnosed with high resolution MER per subcarrier and spectrum analysis information, available through D3.1 PNM. The RxMER, for example, can be used to identify ingress that is at a lower power than the signal, and as such not visible on typical spectrum analysis tools. For example, the PLC location relative to mobile spectrum is now leveraged, along with MER, to provide a preferred PLC location, should new spectrum be licensed and deployed, or other ingress challenges emerge. With each of these detection events, the following bullets describe types of data are made available to other OSS tools that do the triangulation, event prioritization, and triage functions; these OSS tools can correlate with Geographic Information System (GIS) and address information to isolate the common network point of the issue:

- Count and list of impacted cable modems
- Total MHz of impairment and spectrum location
- API reference to access the results of the pattern detection data analysis (such as the data informing Figure 25)
- API reference for historical data stored in the data lake related to the event
- API reference for real-time collection, on-demand, to enable fix agents to confirm the issue is still present, and to assist in both isolation and mitigation confirmation
- Other metadata to enrich the event, such as the mobile wireless carriers, or the more optimal PLC location

Two examples from our pattern detection framework are described below. Each of these has a specific pattern detection algorithm associated with the event forwarded through the notification service. To illustrate the pattern detection process, the following rule-based approach is used:

- **Sweep Generator Detection** (Figure 24)
  1. MER for all cable modems for the past 3 days (~n Polls) collected.
  2. Because the sweep insertion points come and go, we aggregate the minimum MER for all subcarriers to maximize the possibility of detecting sweep.
  3. The algorithm checks individual cable modems to see if there are subcarriers exactly 3 MHz or 6 MHz apart that have significantly lower MER than the overall MER.
  4. The algorithm excludes cable modems where many subcarriers around the potential sweep points also have significantly lower MER, as this may be a sign of another type of interference.
  5. If more than 30% of cable modems on a given OFDM channel meet the above criteria, we classify the OFDM channel as being affected by sweep generator activities, and send notifications to Comcast OSS tools.

- **Mobile Wireless Ingress Detection** (Figure 25)
  1. The following datasets are used for detecting Mobile Wireless Ingress:
     - MER for all cable modems for the past 3 days (~n Polls) collected
     - Spectrum allocation by state and county
     - Account/cable Modem – ZIP code mapping
     - ZIP Code – state mapping
  2. The algorithm identifies portions of the OFDM channel that overlap with other spectrum owners for individual cable modems by joining the above datasets
  3. For any overlapping spectrum and poll, the algorithm checks for the presence of the following 2 conditions; if both these conditions are met, the cable modem is classified as being impacted by Mobile Wireless Ingress:
     - The average device MER is lower than the overall average MER by more than 3 dB
     - Drops exist in MER between consecutive subcarriers of more than 3 dB on either side of the overlapping spectrum

In addition, we are currently developing alternative approaches for Pattern Detection by working with expert field techs to label MER curves with the various impairment categories. With that information, we can train classification models by applying supervised machine learning (ML) approaches to the problem.

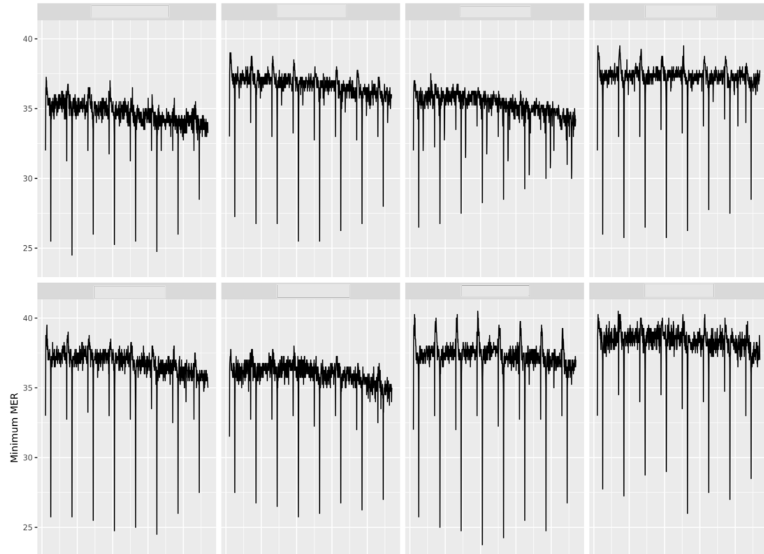


Figure 25 – Example MER charts for sweep generator insertion points in OFDM channel. The sweep insertion points are 3 or 6 MHz apart.

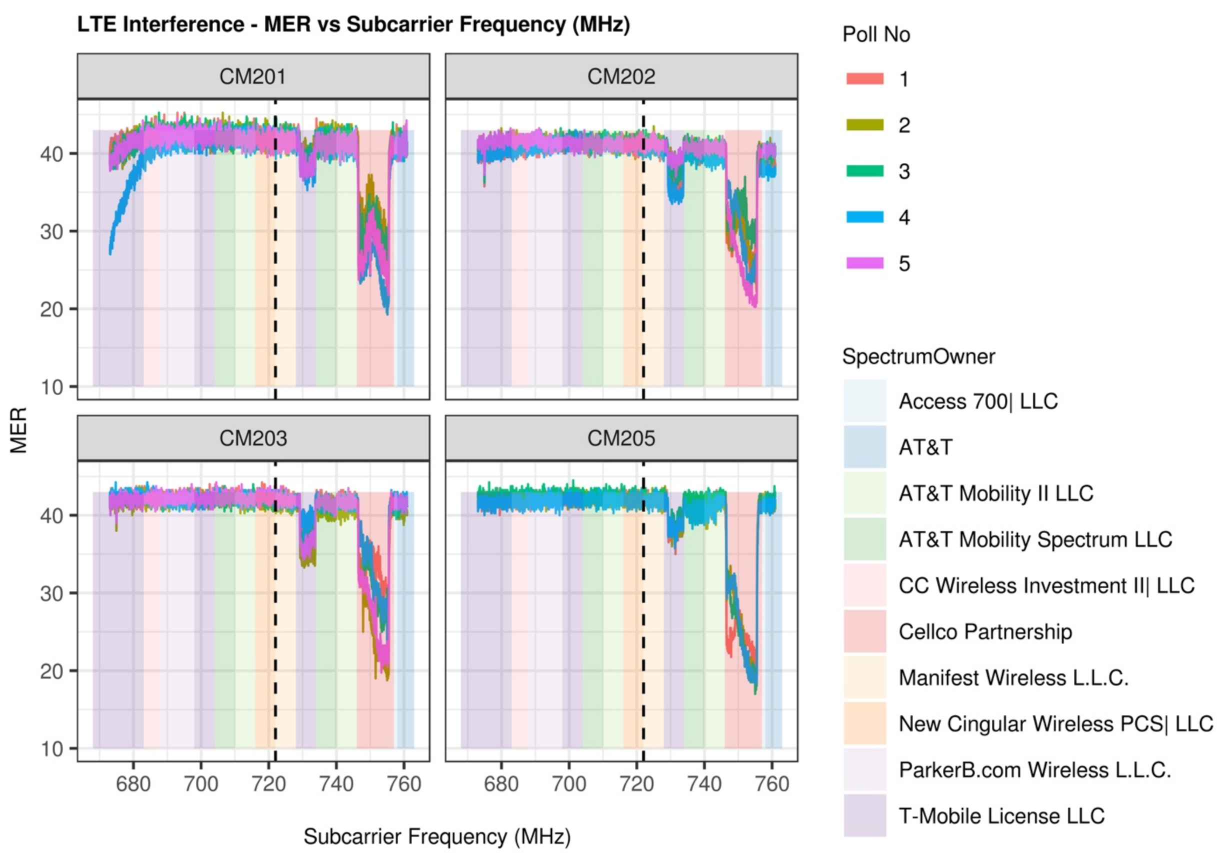


Figure 26 - Example of detected LTE Interference patterns, enriched with mobile carrier license data. The devices shown in this example experience two ingress patterns that perfectly overlap with the spectrums used by T-Mobile (~35-dB level) and Cellco Partnership (~20-dB level).

Future Work

The future work efforts are focused on measuring the system response to the application of profiles through the FEC data, supporting Orthogonal Frequency Division Multiple Access (OFDMA) and D3.0 SC-QAM for the upstream, and enhancing our telemetry collection infrastructure.

10. FEC

Noise in a communication channel introduces the possibility that a symbol will be received erroneously, thus degrading the effective capacity of the channel. For the same level of S/N, increasing the modulation orders increases the likelihood of a symbol being errored. Given the clear goal of maximizing the channel capacity, we must also make special consideration for the increased likelihood of receiving failed symbols.

Modern communication systems mitigate this by employing advanced FEC techniques. Correspondingly, FEC can also serve as a primary feedback signal -- assuming we can measure the *amount* of FEC used for each recommended and applied profile on the network. Future work will include the consideration of levels of FEC, as part of developing a fully closed- or feed-forward loop system, so as to be able to measure the response from the network from the changes in profiles applied, as outlined in this paper. With such a system, the new stated objective would be to maximize channel capacity, while simultaneously minimizing the probability of symbol errors. Such a closed- or feed-forward loop system can consider both traditional control theory solutions and advanced machine learning / reinforcement learning techniques.

All aspects of the algorithm discussed thus far can be modeled based on historical data, producing a reasonable view of the generated profiles, modem assignment, and capacity gains achieved. However, when it comes to FEC, simulating error rates (both corrected and uncorrected) is non-trivial. Lacking a proper estimate of error rates, any analysis assessing the merits of a certain approach against another will be biased towards the approach that results in higher overall modulation orders (i.e. we would be simply ignoring the resiliency dimension in the optimization game).

In contrast with the complexity of modeling FEC, deciding on how FEC will be used within the program is somewhat straightforward: Error rates will be monitored continuously, and upper end thresholds on those rates will be set. If a rate is crossed, a corrective action will be taken (either at a modem level or at a CMTS level). For example, increased error rates across a CMTS or across the entire footprint may indicate too aggressive modulation efficiency thresholds, while increased error rates for a specific modem may indicate a newly formed impairment. We need to be able to separate the two and respond with the appropriate action. An example of an appropriate action would be to adjust the global thresholding offset and/or the *n*th percentile hyper-parameters, in response to increased or reduced symbol error rates (in response to overly conservative/aggressive parameter values). To target an action at a cable modem level we will need to modify the existing data model to allow the *n*th percentile, and the thresholding offset, to be defined at the device level (rather than globally). We have recently started experimenting with measuring the system response in the lab and correlating the behavior of the system to the generated noise and applied profiles. These are the first steps towards implementing a closed-loop solution.

11. Up-Stream Signal Path Implications

Orthogonal Frequency Division Multiple Access (OFDMA, a version of OFDM) can be applied for shared usages scenarios, such as upstream channels -- where multiple access is effectively achieved by assigning subsets of the channel subcarriers to different users, as illustrated in Figure 27.

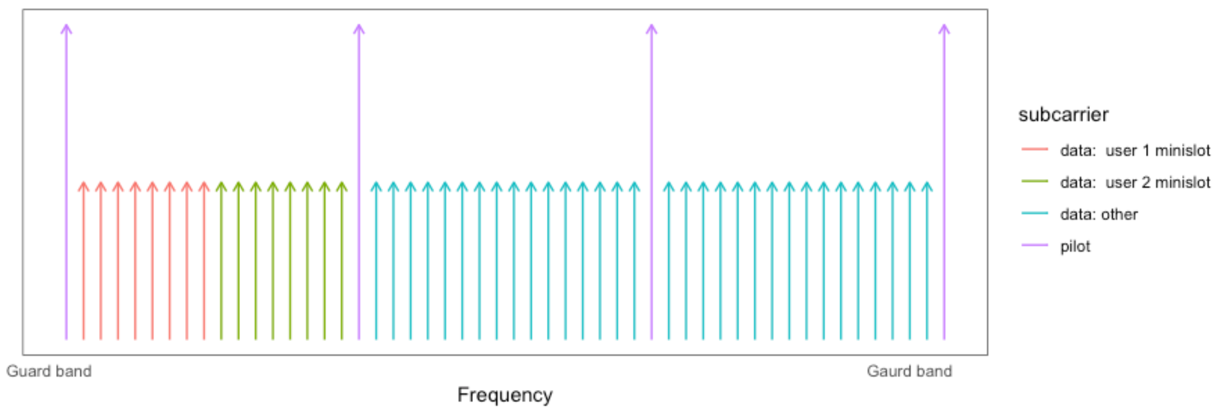


Figure 27. Illustration of subcarrier configuration in OFDMA. In OFDMA, contiguous subsets of subcarriers are assigned to different users.

Given the nature of a combined upstream plant and the nature of OFDMA, additional work will be necessary to adjust the analytic and control elements. We expect to find similar benefits to upstream capacity, including robustness, through the ability to manage and adapt upstream channel modulation profiles. We will update our findings accordingly.

D3.0 upstream channels already carry substantial flexibility, evidenced by variable modulation and error correction. While the network metrics are not yet as granular as those in D3.1, opportunities exist to optimize the performance -- as described in the case study, referenced earlier, and in which different FEC and modulation approaches have been successfully applied to mitigate the ingress impacts of conducted-switching power supplies. PMA for D3.0 US could apply one of a set of pre-defined modulation profiles to a channel. This solution would avoid applying the most robust configurations ubiquitously across the network, by applying them only when and where needed. Initial modeling, based on actual network data and currently conservative modulation profiles, indicates that a 10 to 15% increase in data bandwidth may be achievable. When it comes to the upstream signal path, even a modest improvement can provide significant benefits in deferred capital for node segmentation. These capabilities may also enable additional channels in the upstream spectrum, not currently in use because of challenging group delay or ingress characteristics.

12. Near-Real-Time Operation

Noise and ingress are often sporadic, periodic, and/or transient in nature, where the ability to detect and respond to the noise becomes highly dependent on the telemetry capture and sampling frequency. As we push further spectral efficiency and closer to Shannon limits, we'll need to fully understand what optimal sampling and the corresponding frequency of adaptive adjustments that are required to maintain maximum capacity with optimal robustness. Within these efforts, we're currently experimenting with a special on-demand telemetry collection that runs at much higher frequency compared to our standard solution. The new telemetry collection system referred to as "spot-beaming" targets a very small subset of OFDM interfaces in the network and runs on a sub-minute cycle. The current use case for this system is to collect MER data for specific interfaces of interest where gaining deeper knowledge of the MER dynamics is required. Managing the targeted interfaces is done through an API, such that spotbeaming interfaces can be created or deleted on a needs basis. We still have much work to do in terms of understanding how large of coverage can the spotbeaming system target before running into performance-economic issues. These investigations will also help us understand the possibilities in terms of bridging the gap between our standard collection scheme and the spotbeaming system, with the objective of collecting telemetry in near-real-time across the entire network.

Conclusion

Implementing PMA across Comcast's D3.1 network holds the opportunity to increase downstream bandwidth by ~30% while enhancing the resiliency of the network. The path to implementing PMA involved addressing a multitude of challenges including the hardening of D3.1, establishing a data architecture that supports capturing telemetry data for ~millions of cable modems at a sufficiently high sampling frequency, and establishing a lab for development & testing of the end-to-end PMA solution. We presented a core algorithm that recommends profiles that optimize the capacity of an OFDM Interface while taking into consideration the recommended mappings between MER and modulation orders, vendor constraints, and changes in MER over time. We are currently in the process of testing the PMA solution in the field in specific locations. In parallel to these efforts, we presented a host of pattern detection algorithms that complement PMA by ensuring that network issues are not masked by PMA, but rather reported to the proper entity within Comcast for taking corrective actions. Lastly, the development of the core algorithm continues with the anticipation of releasing a second version of PMA that operates in a closed-loop configuration—with the measured uncorrectable codeword error rate constituting a feedback signal to the algorithm.

Abbreviations

CI/CD	continuous integration / continuous development
CM	cable modem
CMTS	cable modem termination system
D3.0	DOCSIS 3.0
D3.1	DOCSIS 3.1
dB	decibel
DOCSIS	data over cable service interface specification
FEC	forward error correction
GIS	Geographic Information System
HFC	hybrid fibre-coaxial
IQR	inter-quartile range
LDPC	low-density parity-check
LTE	long term evolution
MAC	media access control
MER	modulation error rate
N+X	node plus X number of amplifiers in cascade
OFDM	orthogonal frequency division multiplexing
OFDMA	orthogonal frequency division multiple access
PLC	PHY link channel
PMA	profile management application
PNM	proactive network maintenance
QAM	quadrature amplitude modulation

SC-QAM	single carrier quadrature amplitude modulation
SDR	Software-defined radio
SoC	system on chip
SW	software

Bibliography & References

1. White and Sundaresan, *DOCSIS 3.1 Profile Management Application and Algorithms*. SCTE IBSE, NCTA, CableLabs, Spring Technical Forum Proceedings, 2016.
2. Cablelabs, *Data-Over-Cable Service Interface Specifications DOCSIS® 3.1, Physical Layer Specification, CM-SP-PHYv3.1-I02-EC*, 2014.
3. Rokach, Lior, and Oded Maimon. *Clustering methods*. Data mining and knowledge discovery handbook. Springer US, 2005.
4. Mutalik, Rice, Subramanya, Wang, *What Gets Measured Gets Done / What Gets Analyzed Gets Transformed, Analytics for a Wider/Deeper Network View*. SCTE IBSE TEC Expo, 2018.

