PMA Improvements – Strategies Employed for Faster Mitigation, Increased Capacity, and Cost Savings

A Technical Paper prepared for SCTE by

Jonathan Leech
Principal Engineer II
Comcast
183 Inverness Dr West, Englewood, CO 80112
(720) 289-1561
jonathan_leech@cable.comcast.com

Andy Martushev
Principal Engineer II
Comcast
183 Inverness Dr West, Englewood, CO 80112
720-284-2277
andy_martushev@cable.comcast.com

Table of Contents

Title **Page Number**

List of Figures

Title **Page Number**

List of Tables

Title **Page Number**

1. Introduction

A Profile Management Application (PMA) is a critical component of DOCSIS downstream and upstream for both speed and reliability. This is especially true with increased bandwidth demands in recent years. As such, it is critical to react quickly to issues in order to provide the best customer experience. Faster mitigation of network issues reduces customer impact and call volumes. The profile recommendation interval was lowered from 6 hours in the previous system (Harb, 2020) to 5 minutes in the DOCSIS 3.0 (D3.0) upstream (US), and from 3.5 days to 1 day in the DOCSIS 3.1 (D3.1) US and downstream (DS), while reducing operational costs and improving capacity.

Ingesting and analyzing large amounts of data at a high rate creates high demand for both storage and CPU. The technology stack was refactored and costs were lowered by eliminating redundancy and leveraging streaming, batching, cloud computing, and parallel processing. Aligning the polling and PMA processing using Simple Storage Service (S3), Simple Notification Service (SNS), and Simple Queueing Service (SQS) allows for processing of a single batch of related data immediately after polling. Storage demand was reduced by moving components from a relational database to S3 with a large batch size. CPU demand was reduced by moving the analysis logic from a large Apache Spark cluster to a smaller Elastic Kubernetes Service (EKS) cluster. CPU demand was further reduced by refactoring the clustering algorithm to use Single Instruction Multiple Data (SIMD) parallel processing.

Making PMA recommendations more often improves capacity to an extent, but larger capacity gains were made by making changes to the profile selection and clustering algorithms. For D3.1, the Modulation Error Ratio (MER) data model was improved by using histograms and a time decay function. Better utilization and capacity estimates were created by using the model and the added time dimension. Optimal percentiles and corresponding weights are generated for each modem and used as inputs to the clustering algorithm. The result was capacity gains of greater than 8% and 5 Tb/s.

2. US D3.0 Profile Management

Capacity gains were increased primarily by decreasing the recommendation interval from 6 hours to 5 minutes. To do this, the data granularity was also changed, and the operating model changed from batch to streaming. Additionally, some changes were made to the profile recommendation business logic. Finally, costs were reduced by eliminating redundancy in the polling and data ingest pipeline.

2.1. Data Granularity

Per-modem SNR and Forward Error Correction (FEC) statistics were replaced with interface-level aggregates in the data ingest pipeline. Interface-level data is orders of magnitude smaller than the equivalent per modem data, which allows for more frequent collection and the incorporation of more historical data for making profile recommendations. Data ingested includes interface level FEC, MER, Profile, and Utilization metrics.

2.2. Collection and Recommendation Interval

Interface-level aggregate SNR and FEC is ingested in 5-minute intervals. The incoming data is retrieved from AWS S3 storage and an SNS notification informs of the availability of new data. The data is written to a time series database. There is minimal lag between the poll and the ingest, and a new set of profile recommendations is made immediately. This allows the system to respond quickly to a change in the plant conditions and results in a better customer experience. As a result, a 60% reduction in major alarms and a 50% reduction in minor alarms was achieved.

2.3. Recommendations

US profiles for D3.0 channels are comprised primarily of a modulation order from Quadrature Phase Shift Keying) (QPSK) to 64-Quadrature Amplitude Modulation (QAM), and data and parity lengths for short and long grants. A recommendation is then made and implemented for each upstream channel on a Cable Modem Termination System (CMTS). Recommendations are made by choosing the most appropriate profile for the channel from a fixed set of available profiles. The chosen profile minimizes FEC error rates while maximizing throughput. Additional consideration is given to interfaces with high utilization, e.g., a profile change won't be made if it would increase the utilization above 80%.

To reduce profile flapping (repeated profile changes downgrading and upgrading the same channel), the algorithm will lower the profile more readily than it will raise it. It also takes into consideration past profiles and their corresponding MER values.

2.4. Downgrade Profile

The decision to downgrade a profile is made when the channel uncorrectable FEC rate is impaired (> 1%) for 20 minutes.

2.5. Upgrade Profile

The profile is upgraded after 30 minutes of clean uncorrectable and correctable FEC rates (< 0.0001%) and MER > 36 dB, or 75 minutes of Uncorrectable FEC < 0.1%, and a +2 decibel (dB) improvement in MER and a 50% reduction in correctable FEC.

2.6. Capacity Gain

On average, the capacity gain is measured at 16.6 %. This is about a 1.5% capacity improvement vs. the previous version of US D3.0 PMA. Gains in various markets vary from 14% to 22%. US D3.0 PMA adds about 4.2 Tb/s in total capacity to the upstream, which represents an average gain of 3.1 Mb/s per channel. System performance and capacity gains are continuously monitored via dashboards and automated alerting.

Overall Profile Stats

cmts_count	gainpct	interface_count	rawgain	avgintgain
3756	16.6	1304932	4.18 Tb/s	3.05 Mb/s

Figure 1 – Overall US D3.0 PMA Stats

2.7. Dashboards

The dashboards provide insight into the performance and health of the system. Figure 2 shows a dashboard view of the performance of the US3.0 PMA system, showing statistics for the Mountain West region, including profile distribution, capacity gain, interface degradation, and the number of changes applied by the PMA System.

Figure 3 shows a dashboard view of the health of the US3.0 PMA system. Metrics shown include processing time and volume, message lag, and quantities of profile changes.

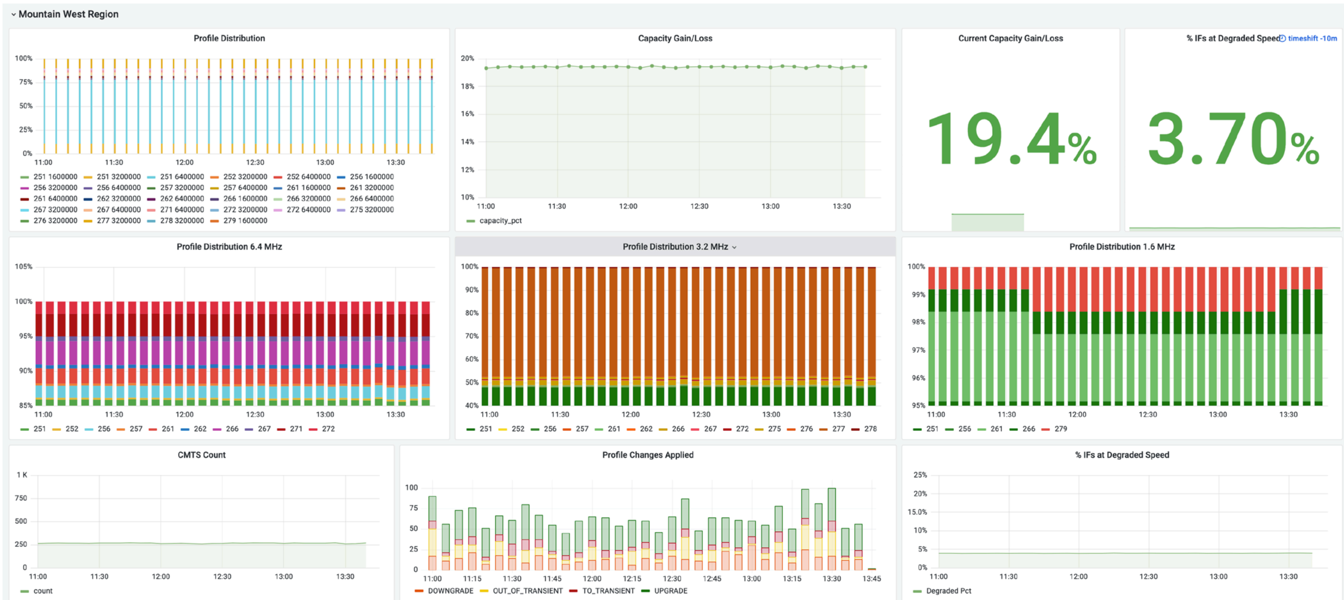


Figure 2 - Example US D3.0 PMA Stats for Mountain West Region

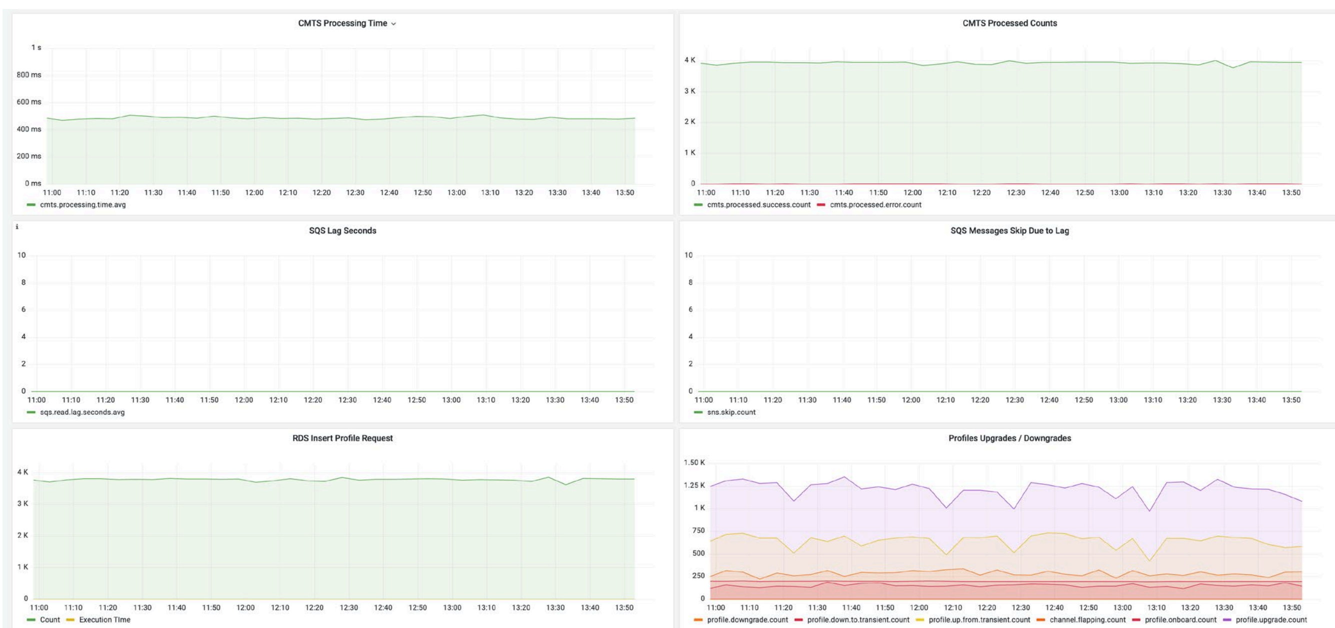


Figure 3 - Example US D3.0 PMA Monitoring Dashboard

3. DS D3.1 Profile Management

Capacity was increased while lowering cloud computing costs, by improving the data model, improving the clustering algorithm, and making performance improvements to the clustering algorithm. Additionally, the processing model was changed from batch-oriented to streaming. The performance improvements allowed the recommendations to be made more frequently as compared to the prior version, lowering the recommendation interval to 1 day from 3.5 days. As a result, approximately 8% more OFDM capacity was gained, more than 5 Tb/s in total.

3.1. Model

The MER data model was changed to use a two-dimensional histogram, or heatmap, incorporating an exponential decay function. Figures 7 and 8 below show representative histogram rendered using a color scale. This allows for the quick computation of any percentile value between 1 and 100 without sorting. The purpose of the decay function is to age out older data over time and introduce a recency bias. A half-life of 4 days was chosen for the decay function. An entire CMTS worth of compressed JSON data is stored in a single S3 file, organized by interface. On average, the data for a modem (with a single 96 MHz wide Orthogonal Frequency Division Modulation (OFDM) channel, 25 kHz subcarrier spacing) takes up only 75 kilobytes of space, or about 20 bytes per subcarrier. This space savings is the key to cost savings in cloud storage, as the full dataset is both read and written hourly.

Modem level per-subcarrier MER data is ingested on an hourly basis. The polling system produces a single S3 file per CMTS and sends an SNS notification when the file is ready. The large batch size minimizes costs for reading the incoming data. Due to the streaming nature of the system, there is a minimal delay between the poll completion and the processing.

The previous system (Harb, 2020) used a static MER percentile and a weight of 1 for each modem. Using a static 10^{th} percentile of MER of each modem as an input to the clustering algorithm was insufficient to achieve optimal gains. This is based on the observation that more than 90% of the time the modem MER is higher than the 10^{th} percentile MER and could thus be using a higher profile. On the other side, up to 10% of the time the MER is lower than the 10^{th} percentile MER, and the modem might be using profile 0 (see below). As a result, the system under-estimated capacity gain.

The changes to the model allow for a more accurate estimation of capacity gain, which in turn allow for greater capacity gain. A fixed profile 0 at 64 or 256-QAM is used, and PMA generates 3 additional dynamic profiles. Depending on the MER distribution, a modem may spend a percentage of time on a single profile or a subset of all the profiles. Using the time percentages as weights, the estimated capacity for a modem is calculated as the weighted average of the profile bit-loading. The estimated capacity for an OFDM interface is then calculated as the average of the per modem estimates. Further increases of estimation accuracy at the OFDM interface level could be obtained by weighting the average by per-modem utilization. This would result in larger capacity gains and would be useful for comparing the predicted capacity against actual capacity but could unfairly punish lower utilization users. This is because modem utilization is not equal. For example, a modem with a high utilization on a low profile would skew the actual capacity downwards as compared to the estimate.

A single, optimal profile for a modem can be computed by iterating over the percentiles 1 – 100, estimating the resulting capacity as above, and choosing the percentile that maximizes capacity in conjunction with profile 0. Similarly, a set of N optimal profiles can be computed for a modem by choosing N percentiles of increasing value such that the chosen percentiles maximize capacity.

3.2. Clustering Algorithm

The clustering algorithm as described in (Harb, 2020) already minimizes capacity loss. The inputs were a profile per-modem (computed from the 10th percentile of MER values), with a weight of 1 for each profile. The clustering algorithm reduces the input to the desired number of profiles, e.g., 3, by merging the two profiles together that result in the lowest capacity loss. The improvements to capacity gain were achieved primarily by improvements to the inputs. Versus the previous inputs of a single profile for each modem, 2 optimal profiles are instead computed for each modem (as described above), as well as an additional input representing profile 0. Rather than a weight of 1.0 for each input, varying weights are used. The weight used for each input represents the estimate of the percentage of time spent on the profile, as mentioned above. The weight for profile 0 is the sum of the percentages of time each modem is estimated to use profile 0. Including its contribution to the profile 0 weight, the sum of the weights for each modem's inputs equals 1.

Rather than excluding modems with low MER prior to the clustering, the clustering algorithm merges those inputs as appropriate into the profile 0 cluster. As such, depending on the distribution of MER values, some modems may be ultimately assigned 0, 1 or 2 profiles.

Table 1 – OFDM Profile MER Thresholds, 5 dB more aggressive than DOCSIS 3.1 spec

Bits	Modulation	MER Threshold (dB)
2	QPSK	6.0
4	16-QAM	12.0
6	64-QAM	18.0
7	128-QAM	21.0
8	256-QAM	24.0
9	512-QAM	27.5
10	1024-QAM	31.0
11	2048-QAM	34.0
12	4096-QAM	38.0

3.3. Performance

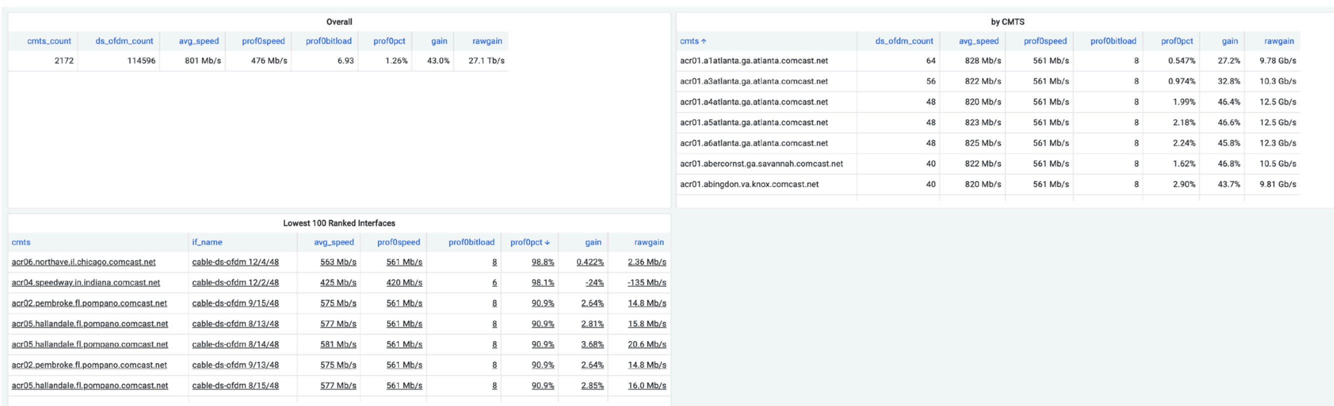
The core clustering algorithm was made 25 times faster. The biggest speed increases were due to rewriting the distance function using SIMD parallel processing via the Advanced Vector Extensions 512 (AVX-512) instructions on Intel CPUs and in Java via the jdk.incubator.vector package. This allows for 16 integer operations to run in parallel on a single core. The next largest boosts came from switching the internal representation from double to int, and from immutable Scala Array types to primitive arrays, which was also necessary to take advantage of SIMD. Finally, upgrades to versions of the Java Virtual Machine (JVM), Scala, and Spark software were made.

Some of the performance gains were offset due to doubling the number of inputs. As the algorithm has an $O(N^2)$ complexity, doubling the inputs results in 4 times slower performance. The net speed increase of more than 6 times faster directly results in lower cloud computing costs. Rather than a single large batch, the work is spread throughout the day. This lowers cloud computing costs by keeping a smaller number of cores continuously busy. The cycle time was reduced from twice a week to daily, and is expected to be reduced to hourly on an as-needed basis, matching the MER ingest rate. As it was also observed in the

D3.0 US, faster response to changes in the plant result in greater capacity gain / and or less capacity loss. However, and an important difference: because the model and clustering account for MER changes over time, and the CMTS assigns profiles periodically throughout the day, a daily cycle is sufficient in most cases.

3.4. Capacity Gain

Greater gains are possible on OFDM than on D3.0 channels, due to the higher modulation orders available. The average capacity gain on OFDM is 43%. Actual capacity gain is measured vs a static 8-bit (256-QAM) modulation profile. As the maximum modulation is 12-bit (4096-QAM), the maximum gain possible on a perfect network would be 50%. This is about an 8% improvement to capacity vs the previous version. This represents an average bits / symbol across all OFDM channels of 11.47, compared to 10.83 on the prior version. In terms of raw capacity, the new DS D3.1 PMA is adding 27.1 Tb/s of total DS capacity, about 5 Tb/s more than the previous version.



Overall

cmts_count	ds_ofdm_count	avg_speed	prof0speed	prof0bitload	prof0pct	gain	rawgain
2172	114596	801 Mb/s	476 Mb/s	6.93	1.26%	43.0%	27.1 Tb/s

by CMTS

cmts ↑	ds_ofdm_count	avg_speed	prof0speed	prof0bitload	prof0pct	gain	rawgain
acr01.a1atlanta.ga.atlanta.comcast.net	64	828 Mb/s	561 Mb/s	8	0.547%	27.2%	9.78 Gb/s
acr01.a3atlanta.ga.atlanta.comcast.net	56	822 Mb/s	561 Mb/s	8	0.974%	32.8%	10.3 Gb/s
acr01.a4atlanta.ga.atlanta.comcast.net	48	820 Mb/s	561 Mb/s	8	1.99%	46.4%	12.5 Gb/s
acr01.a5atlanta.ga.atlanta.comcast.net	48	823 Mb/s	561 Mb/s	8	2.18%	46.6%	12.5 Gb/s
acr01.a6atlanta.ga.atlanta.comcast.net	48	825 Mb/s	561 Mb/s	8	2.24%	45.8%	12.3 Gb/s
acr01.abercornst.ga.savannah.comcast.net	40	822 Mb/s	561 Mb/s	8	1.62%	46.8%	10.5 Gb/s
acr01.abingdon.va.knox.comcast.net	40	820 Mb/s	561 Mb/s	8	2.90%	43.7%	9.81 Gb/s

Lowest 100 Ranked Interfaces

cmts	if_name	avg_speed	prof0speed	prof0bitload	prof0pct ↓	gain	rawgain
acr06.northave.il.chicago.comcast.net	cable-ds-ofdm 12/4/48	563 Mb/s	561 Mb/s	8	98.8%	0.422%	2.36 Mb/s
acr04.speedway.in.indiana.comcast.net	cable-ds-ofdm 12/2/48	425 Mb/s	420 Mb/s	6	98.1%	-24%	-135 Mb/s
acr02.pembroke.fl.pompano.comcast.net	cable-ds-ofdm 9/15/48	575 Mb/s	561 Mb/s	8	90.9%	2.64%	14.8 Mb/s
acr05.hallandale.fl.pompano.comcast.net	cable-ds-ofdm 8/13/48	577 Mb/s	561 Mb/s	8	90.9%	2.81%	15.8 Mb/s
acr05.hallandale.fl.pompano.comcast.net	cable-ds-ofdm 8/14/48	581 Mb/s	561 Mb/s	8	90.9%	3.68%	20.6 Mb/s
acr02.pembroke.fl.pompano.comcast.net	cable-ds-ofdm 9/13/48	575 Mb/s	561 Mb/s	8	90.9%	2.64%	14.8 Mb/s
acr05.hallandale.fl.pompano.comcast.net	cable-ds-ofdm 8/15/48	577 Mb/s	561 Mb/s	8	90.9%	2.85%	16.0 Mb/s

Figure 4 – DS D3.1 PMA Profile Stats

3.5. Dashboards

The dashboards for DS D3.1 PMA show the system performance and health. Figure 4 shows the overall performance of the system, including the capacity gain, as well as a break-down by CMTS, and a view of the poorest performing interfaces.

Figure 5 shows a dashboard view of the performance of a specific OFDM interface. Metrics shown include traffic distribution by profile, channel speed by profile, interface statistics, etc.

Figure 6 shows a dashboard view of the health of the DS D3.1 PMA system, focused on AWS resources. Metrics shown include CPU utilization of the components, and metrics for the various notification topics and queues used by the system.

Figures 7 and 8 show example modem MER histograms, the optimal percentiles chosen as inputs to the clustering algorithm, and the resulting segmented profiles.

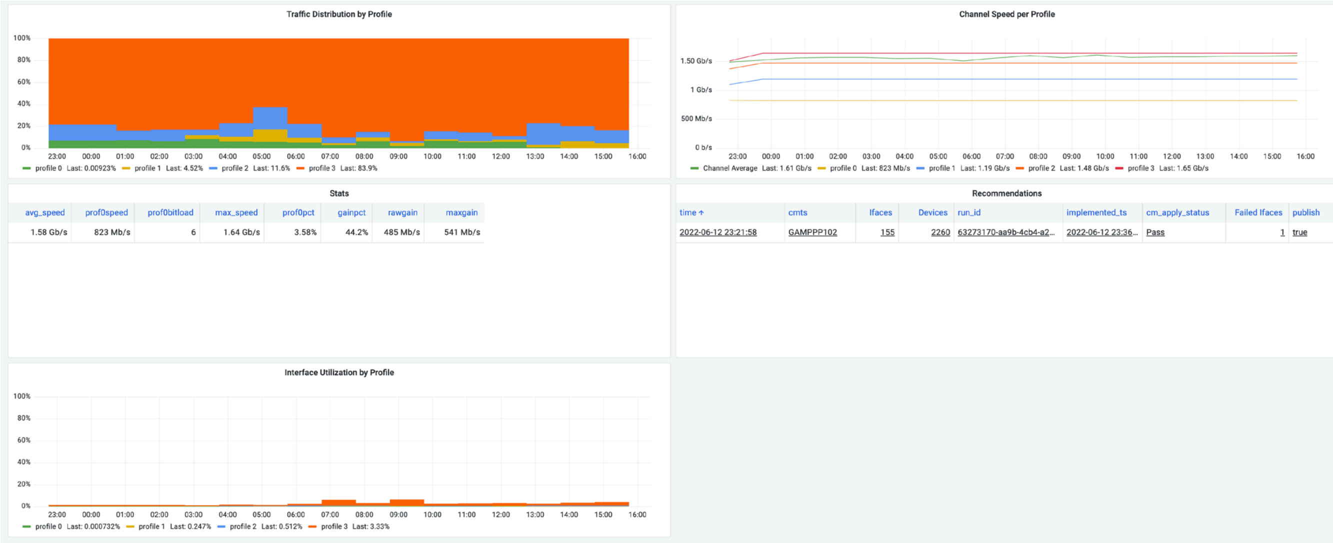


Figure 5 – Example DS D3.1 OFDM Utilization Dashboard

Figure 6 – Example DS D3.1 AWS Dashboard

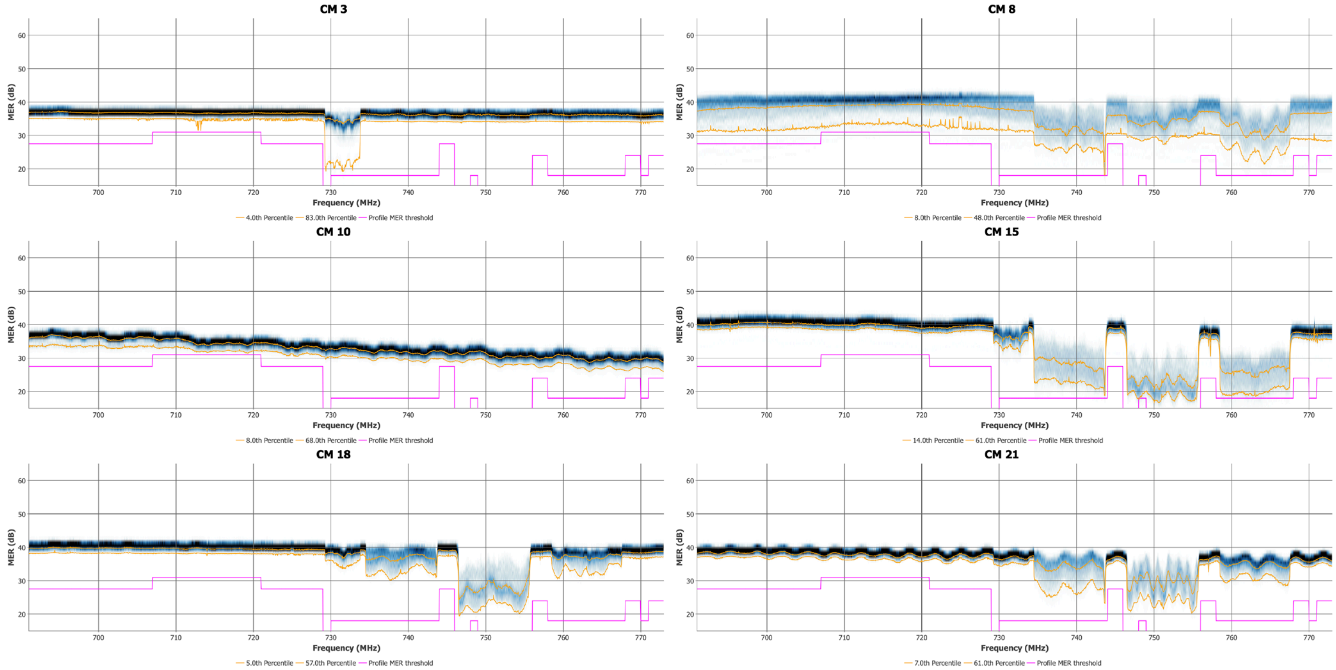


Figure 7 – Example profile 1 modem MER histograms, percentiles, and MER threshold

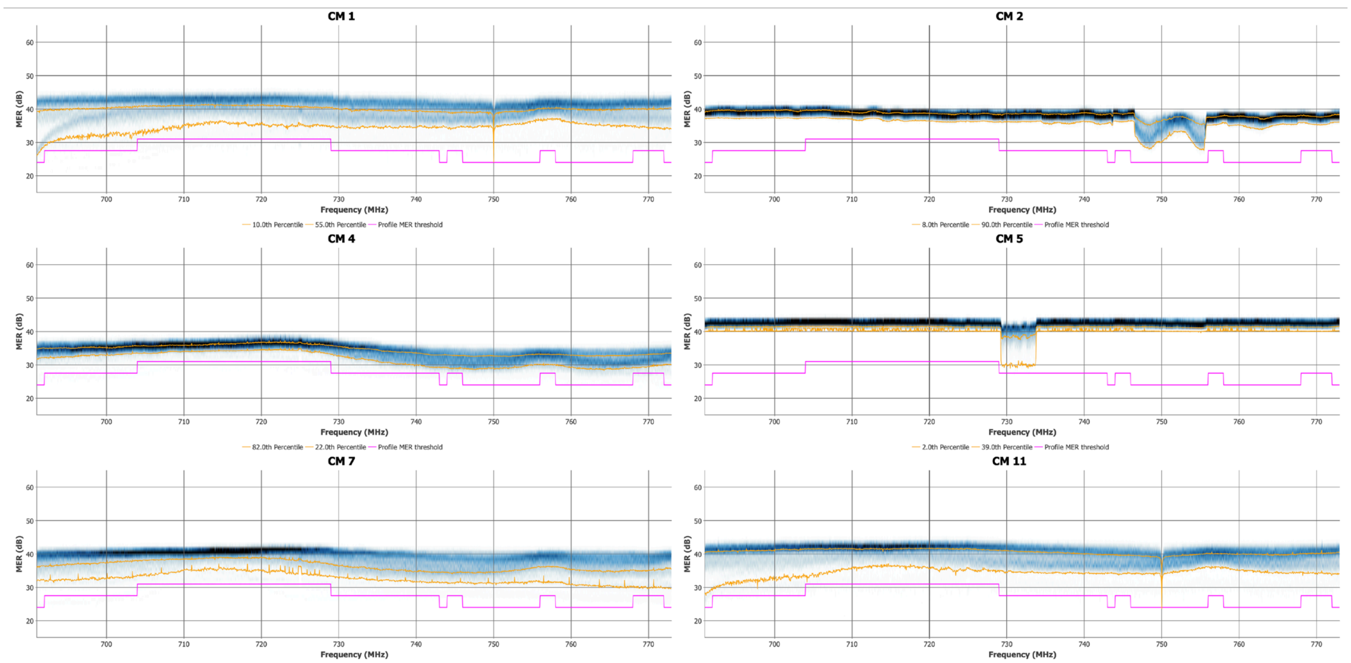


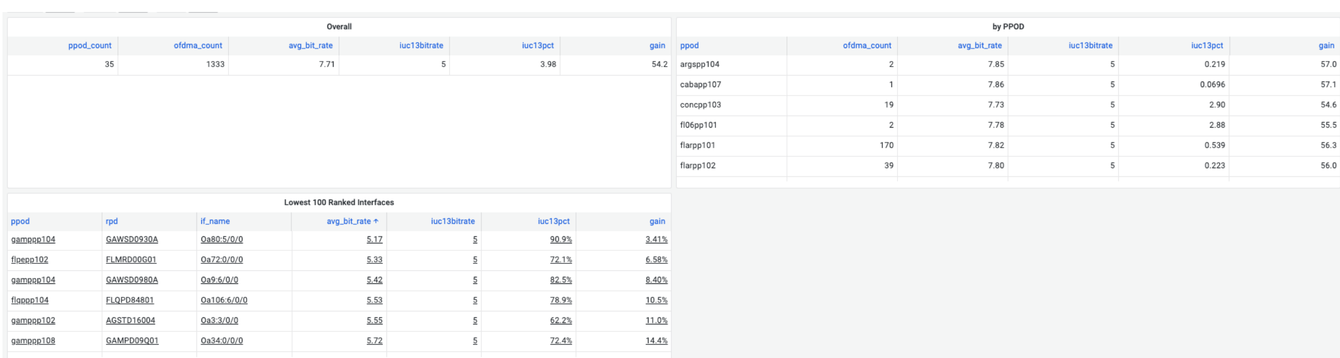
Figure 8 – Example profile 2 modem MER histograms, percentiles, and MER threshold

4. US D3.1 Profile Management

Profile Management for Orthogonal Frequency Division Multiple Access (OFDMA) on US D3.1 is similar to OFDM with a few notable differences. The MER values are per-minislot (400 kHz) instead of per subcarrier. The data is ingested at a 5-minute interval instead of hourly. The increased rate of collection helps offset the loss in granularity. The upstream uses IUCs instead of profiles, and more of them (7 vs 4). A single, static 10th percentile of MER per modem is used as the input to the clustering algorithm. This appears to be sufficient as there is less volatility in the MER values within a single modem as well as across the population of modems on an OFDMA interface. Having less volatility is a factor of the averaging of subcarrier MER into minislots, the upstream funnel effect, the ability of modems to adjust transmit power levels to achieve a desired receive power, and the effectiveness of the pre-equalization process. As of the time this writing, OFDMA PMA is not in full production, but based on trials, capacity gains similar to OFDM PMA are expected.

Table 2 – OFDMA Profile MER Thresolds from DOCSIS3.1 spec

Bits	Modulation	MER Threshold (dB)
2	QPSK	11.0
3	8-QAM	14.0
4	16-QAM	17.0
5	32-QAM	20.0
6	64-QAM	23.0
7	128-QAM	26.0
8	256-QAM	29.0
9	512-QAM	32.5
10	1024-QAM	33.5
11	2048-QAM	39.0
12	4096-QAM	43.0



Overall

ppod_count	ofdma_count	avg_bit_rate	iuc13bitrate	iuc13pct	gain
35	1333	7.71	5	3.98	54.2

by PPOD

ppod	ofdma_count	avg_bit_rate	iuc13bitrate	iuc13pct	gain
argspp104	2	7.85	5	0.219	57.0
cabapp107	1	7.86	5	0.0696	57.1
concpp103	19	7.73	5	2.90	54.6
fl06pp101	2	7.78	5	2.88	55.5
flarpp101	170	7.82	5	0.539	56.3
flarpp102	39	7.80	5	0.223	56.0

Lowest 100 Ranked Interfaces

ppod	rpd	if_name	avg_bit_rate ↑	iuc13bitrate	iuc13pct	gain
gamppp104	GAWSD0930A	Oa80:5/0/0	5.17	5	90.9%	3.41%
flpepp102	FLMRD00G01	Oa72:0/0/0	5.33	5	72.1%	6.58%
gamppp104	GAWSD0980A	Oa9:6/0/0	5.42	5	82.5%	8.40%
flqppp104	FLQPD84801	Oa106:6/0/0	5.53	5	78.9%	10.5%
gamppp102	AGSTD16004	Oa3:3/0/0	5.55	5	62.2%	11.0%
gamppp108	GAMPD09Q01	Oa34:0/0/0	5.72	5	72.4%	14.4%

Figure 9 – US D3.1 PMA IUC Stats

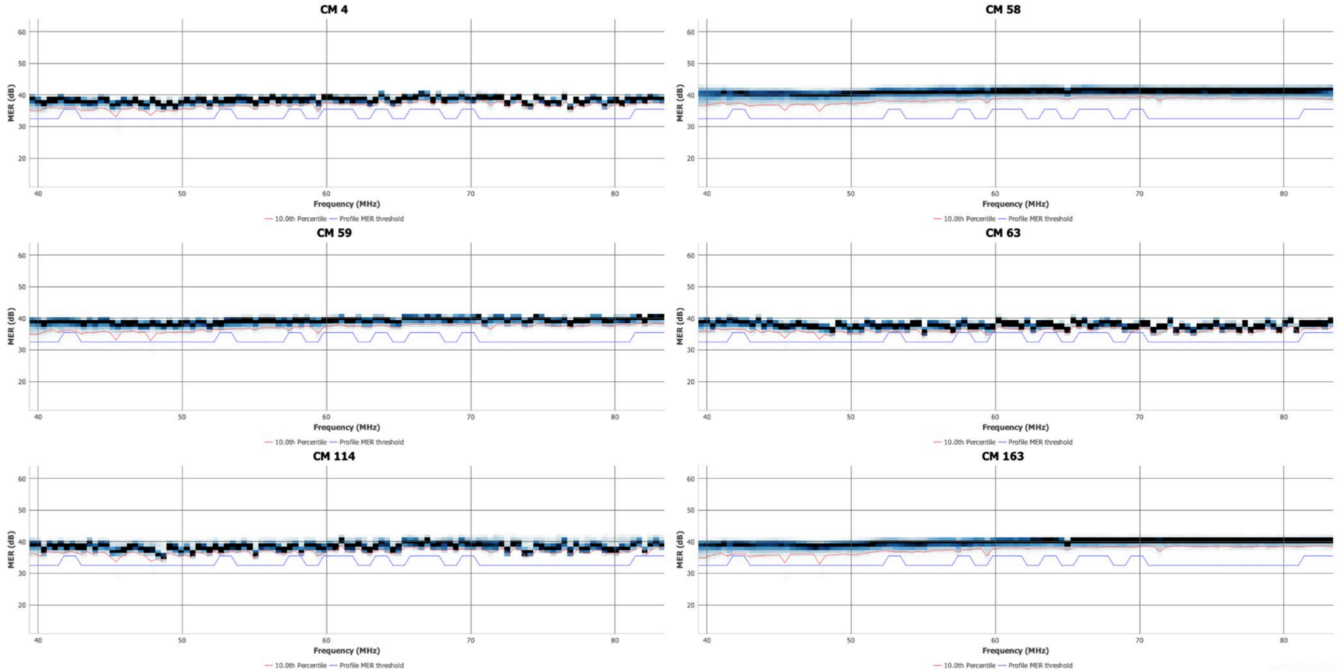


Figure 10 – Example IUC 5 modem MER percentiles, histograms, and threshold

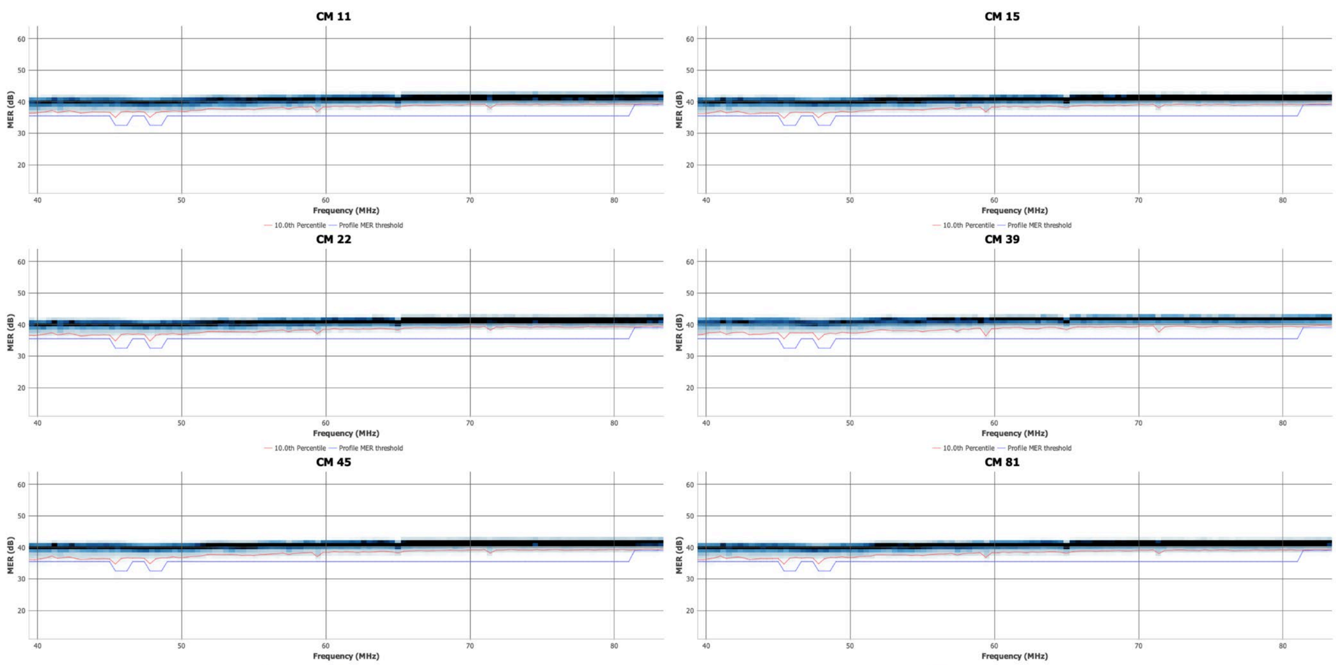


Figure 11 – Example IUC 6 modem MER percentiles, histograms, and threshold

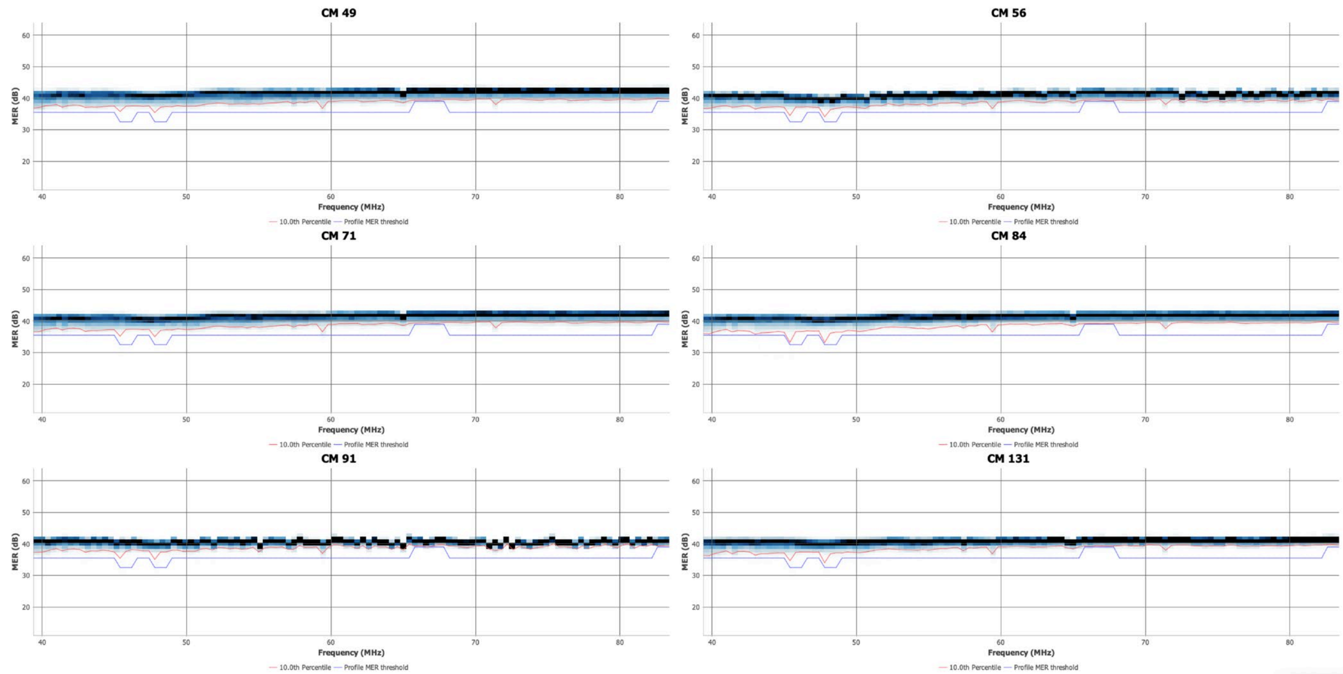


Figure 12 - Example IUC 9 modem MER percentiles, histograms, and threshold

5. Future Work

Increase the profile recommendation rate for Downstream and Upstream DOCSIS 3.1 from daily to hourly on as-needed basis. Faster reaction to changing conditions would result in even greater capacity gains, or less capacity loss.

Make further performance optimizations and improvements to the clustering algorithms. A better result can be achieved by adjusting the weights accordingly as the clusters are merged, and optimizations can be made to reduce the algorithmic complexity.

Investigate generating more profiles and using profile 0 to a greater effect by modifying it from a flat 64-QAM modulation to a segmented profile. In many cases profile 0 could be made to be more robust, have higher throughput, and automatically handle cases otherwise handled by exclusion zones.

The MER thresholds used in the clustering algorithms to determine modulation are estimates, and appropriate values for the thresholds have been chosen, albeit generally conservative ones. There are cases where more aggressive thresholds can be used without driving FEC rates and using them would result in greater capacity gains. There may also be cases where more conservative thresholds are necessary in order to lower uncorrectable FEC rates. In these cases, the capacity estimates that the PMA algorithm makes would be high, and modems would be using profile 0 at a greater percentage than expected. Lowering the thresholds would lower estimated capacity but increase actual capacity, by shifting traffic to higher profiles. By comparing estimated profile utilization with and actual profile utilization, the MER thresholds may be refined, creating realistic thresholds per CMTS, interface, or modem.

6. Conclusion

By making strategic changes to the US D3.0, DS D3.1, and US D3.1 PMA software, the system is able to mitigate issues faster, increase network capacity, and lower cloud computing costs. This was done by eliminating redundancy and leveraging a combination of streaming and batching, cloud computing, and. vector processing. Improvements were also made to the underlying data model, business logic, and clustering algorithms. The net result was a reduction in the US D3.0 recommendation interval from 6 hours to 5 minutes, and from 3.5 days to 1 day for DS 3.1 and US 3.1. Capacity was increased by 1.5% for US D3.0 and by 8% for DS D3.1.

Abbreviations

AVX-512	advanced vector extensions, 512-bit SIMD instruction set on Intel
AWS	Amazon web services, a cloud computing platform
CMTS	cable modem termination system
CPU	central processing unit
D3.0	DOCSIS 3.0
D3.1	DOCSIS 3.1
dB	decibel
DOCSIS	data over cable service interface specification
DS	downstream
EKS	elastic Kubernetes service
FEC	forward error correction
Hz	hertz
IUC	interval usage code
JSON	JavaScript object notation
JVM	Java virtual machine
MER	modulation error ratio
OFDM	orthogonal frequency division multiplexing
OFDMA	orthogonal frequency division multiple access
PMA	profile management application
QAM	quadrature amplitude modulation
QPSK	quadrature phase shift keying, a 2-bit modulation form
S3	simple storage service, object storage on AWS
SIMD	single instruction multiple data, a type of parallel processing
SNR	signal to noise ratio
SNS	simple notification service, message delivery on AWS
SQS	simple queue service, message queuing on AWS
US	upstream

Bibliography & References

1. *"Full Scale Deployment of PMA", SCTE Cable-Tec Expo 2020, https://www.scte.org/documents/3132/1781_Harb_3030_paper.pdf, M. Harb, B. Santengelo, D. Rice, J. Ferreira.*
2. *"A Machine Learning Pipeline for D3.1 Profile Management", SCTE Cable-Tec Expo 2019, https://www.scte.org/documents/3816/799_Harb_A_Machine_Learning_Pipeline_for_D3.1_Profile_Management_paper.pdf, M. Harb, J. Ferreira, D. Rice, B. Santangelo, R. Spanbauer.*
3. *"Practical Implementation of Profile Management Application (PMA) to Improve Data Throughput in the Presence of Impairments", SCTE Cable-Tec Expo 2021, https://www.scte.org/documents/4700/2121_Volpe_3479_paper.pdf, B. Volpe.*
4. *"Practical Lessons from D3.1 Deployments and a Profile Management Application (PMA)", SCTE Cable-Tec Expo 2019, https://www.scte.org/documents/3867/806_Sundaresan_Practical_Lessons_D3.1_Deployments_Profile_Management_Application_PMA_p.pdf, K. Sundaresan, J. Zhu, M. Mishra, J. Lin*
5. *Cablelabs, Data-Over-Cable Service Interface Specifications DOCSIS® 3.1, Physical Layer Specification, CM-SP-PHYv3.1-I02-EC, 2014.*
6. *"AVX-512", Wikipedia, https://en.wikipedia.org/wiki/AVX-512*

