MONITORING THE VARIABLE NETWORK:
IMPACT OF DOCSIS® 3.1 ON PLANT UTILIZATION METRICS

Niki Pantelias
Broadcom Corporation

Abstract

In order to make good decisions about traffic engineering and capital expenditures in the access network, an operator must have a clear understanding of the capacity and utilization of the plant. A heavily utilized plant segment may require a node split or modification of channel allocation to support customer demand. Accurate assessment of network utilization allows potentially expensive upgrades to be targeted where the need is greatest and return on investment is the highest.

DOCSIS® 3.1 introduces new sources of uncertainty into the metrics for capacity and utilization. The use of multiple profiles with variable data rates increases performance for modems with high SNR, but also means that the capacity of the access network is not fixed; instead, it may vary over time depending on traffic pattern, mix of active modems, system configuration, plant conditions, and other factors.

This paper offers operators a clear understanding of the impact of new DOCSIS 3.1 features on current methods of measuring downstream plant utilization and capacity. Topics addressed include: use cases for multiple profiles with variable data rates, MAC-layer tools and methods for determining what modems can use a given profile, detailed analysis of the accuracy operators can expect from the DOCSIS 3.1 utilization MIBs for various configurations and traffic mixes, explanation of factors that impact accuracy and how operators can configure the system to address them, and suggestions for incorporation of this information into network planning.

BACKGROUND

DOCSIS 3.1 introduces many new capabilities for cable data transmission systems. The Orthogonal Frequency-Division Multiplexing (OFDM) technology used by the DOCSIS 3.1 downstream offers a wide range of parameter settings to address specific plant conditions. One important new feature of the DOCSIS 3.1 downstream is the ability to use different modulation profiles to reach different groups of cable modems.

Profile Definition

A "profile" in the DOCSIS 3.1 downstream is a set of assignments of modulation orders to OFDM subcarriers on a given channel. Profiles are formally defined by DOCSIS MAC Management Messages carried on the channel. A single OFDM channel can have a bandwidth of up to 192 MHz, and may contain in excess of 7000 data-carrying subcarriers. In the DOCSIS 3.1 MAC protocol, an OFDM Channel Descriptor (OCD) message defines these and other parameters of the channel. A different message called a Downstream Profile Descriptor (DPD) is used to define each downstream profile available on the channel. A DPD message assigns a modulation order to each data subcarrier on the channel. Supported modulation orders for data subcarriers range from 16 QAM to 4096 QAM (with a provision for higher modulation orders in the future). A subcarrier may also be "zero bit loaded" in a particular profile. In principle, every subcarrier could have a different modulation order from the subcarriers adjacent to it. This high level of flexibility allows profiles to be configured to avoid very narrow bands of interference or impairment.

A DOCSIS 3.1 Cable Modem Termination System (CMTS) allows up to 16 different profiles to be configured on a given channel. The profiles are identified in their respective DPD messages by numerical values, but in verbal communication, including the DOCSIS specification language, they are often referred to by letters, e.g., “Profile A,” “Profile B,” and so forth. “Profile A” is configured so that it can be received by all modems and is used for initialization and for distribution of certain broadcast MAC messages, such as MAPs and UCDs. Other profiles have non-specific uses but are expected to use higher modulation orders than Profile A in at least part of the spectrum. These profiles would be used to carry unicast or multicast data traffic more efficiently than Profile A. A DOCSIS 3.1 CM has the ability to receive up to four different profiles – Profile A plus up to three additional profiles; in addition, it is capable of conducting tests (but not passing traffic) on a fifth profile.

Use Cases for Profiles

Use of multiple profiles can be helpful whenever a downstream channel impairment impacts different modems on the plant in different ways. Some example impairments are:

- Low SNR. This may occur, for example, when a modem is located deep in a home network, behind a number of splitters with high loss. The result is a lower-than-expected signal level at the modem, even though the signal level at the entry to the home meets operator specifications. A profile addressing this case would use lower modulation orders across the full spectrum. On today’s cable plants, SNR variations of 10 dB or more have been reported across different modems. This could motivate operators to provision multiple profiles with a range of modulation orders covering the range of SNRs on the channel.
- Tilt variations. Drop cables are one source of spectrum tilt due to their small diameter. Modems connected to drop cables of different lengths will see different tilt. A profile aimed at modems with higher tilt would use lower modulation orders at higher frequencies compared to one aimed at modems with lower tilt.
- Diplex filter rolloff. A long cascade of amplifiers, each with a high-order diplex filter at its input and output, will result in a wider rolloff region with higher loss compared to a short cascade. This case is similar to the case of tilt variation and is addressed by using lower modulation orders in the rolloff region, possibly with zero-bit-loaded subcarriers at the outermost frequencies.
- Geographically localized interferers. If a portion of a plant is located near a source of interference, such as a cellular tower, modems near that source may suffer from interference that does not affect modems farther from the source. A profile aimed at modems near the source might use zero-bit-loaded subcarriers at the relevant frequencies, whereas other profiles reaching modems far from the source could have nonzero modulation orders on these same subcarriers.

In all cases, profiles are a tool for avoiding the impairment in question, not eliminating it. Modems that are unaffected by any impairments will operate with the highest instantaneous throughput. Modems experiencing one or more impairments can use different profiles to share the same channel, albeit with reduced capacity due to the use of lower modulation orders or of zero bit loading for some subcarriers.

Assignment of Profile to Modems

When a new modem joins the network, it is necessary to determine what profiles will be the best fit for that particular device. Is this modem a "low SNR," "average SNR," or "high SNR" device, and is it being affected by interference, tilt, or some other impairment?

One way to determine this is to examine the modem's per-subcarrier RxMER (Receive Modulation Error Ratio). For each subcarrier, the value measured at the modem can be compared to the RxMER required to receive the modulation order used for that subcarrier by a particular profile. If sufficient margin is available on all subcarriers, it may reasonably be expected that the modem can receive that profile successfully.

However, this comparison may not tell the whole story. The use of FEC (Forward Error Correction) on the downstream channel offers some amount of error correction capability, meaning that a modem might be able to use a profile even if its RxMER is too low on a small number of subcarriers. Unfortunately, it is difficult to predict exactly what combinations of RxMER will give the desired post-FEC performance. Sometimes the easiest way to find out whether a modem will support a particular profile is simply to try it – ideally in a way that does not disrupt user traffic if the experiment fails.

The DOCSIS 3.1 specification provides new MAC Management Messages incorporating all of the above approaches.

To gather RxMER information from a modem, the CMTS can use the ODS (OFDM Downstream Spectrum) message exchange. In response to an ODS-REQ message from a CMTS, the modem sends an ODS-RSP containing its measured RxMER per subcarrier for all non-excluded subcarriers on the channel. The measurement is performed on pilot or preamble symbols, which will occur on every non-excluded subcarrier at various points in time, so valid measurements can be made even for subcarriers on which the modem has not actively performed demodulation of data.

An operator or an external application may also collect RxMER information directly from the modem through the DOCSIS 3.1 OSSI interface by reading the SNMP MIB object `CmDsOfdmRxMer`.

The OPT (OFDM Downstream Profile Test) message exchange goes further by asking a modem to provide information that is specific to a given profile. In the OPT-REQ message, a CMTS may ask a modem to provide either or both of two types of information:

- RxMER comparison. The CM performs the per-subcarrier comparison described above, with the CMTS providing the target RxMER and required margin for each modulation order.
- FEC codeword testing results. The CM attempts to receive actual traffic on the profile – either in the form of "test codewords" artificially generated by the CMTS, or "live" traffic destined for other modems on the plant. The modem keeps counts of the total number of codewords received and codeword errors (if any), but does not pass any received data to its MAC layer.

The modem reports the results of these operations to the CMTS in the OPT-RSP message.

Results of the ODS and/or OPT exchanges may be evaluated by the CMTS itself or by third-party applications that interface with the CMTS. The specification does not prescribe exact algorithms for use of these results, since it is expected that profile evaluation will be an active area of vendor innovation.

Once a decision is made to assign a particular profile to a modem, the CMTS initiates an exchange of DOCSIS MAC

Management Messages to implement this assignment. For initializing modems, Registration messages may be used. Once a modem is operational, a DBC (Dynamic Bonding Change) exchange is used.

Profile Use in Packet Transmission

Once a profile has been assigned to a modem, the CMTS transmitter may use that profile to send traffic to the modem. As previously described, a modem may have up to four profiles enabled for reception of data. To avoid any issues with packet ordering or duplication, the CMTS is required to transmit all packets for a given Service Flow on a single downstream profile. If the Service Flow uses channel bonding, a single downstream profile must be used on each of the channels being bonded. If multiple Service Flows share the same resequencing DSID, all of these flows will have the same set of assignments of profiles on channels.

Typically, unicast Service Flows will use the highest-throughput profile currently assigned to the destination CM. For Group Service Flows (multicast flows), the selected profile will be one which is assigned to all CMs receiving the flow.

At the CMTS, Quality of Service (QoS) decisions are made to determine the order in which packets are to be transmitted. The basis for these decisions is unchanged in DOCSIS 3.1 – i.e., packets are classified to Service Flows and must pass through per-Flow rate-shaping before being presented to the downstream scheduling engine, which delivers packets for transmission based on relative priority and other QoS parameters. Importantly, profile assignment is not a QoS parameter and has no impact on QoS decisions. The rate, priority, and other QoS-related properties of a Service Flow are not altered by the fact that the Flow may be assigned to a lower- or higher-throughput profile.

Once QoS decisions have been made and a packet is ready for transmission, the CMTS modulator assigns subcarriers using the modulation orders specified by the DPD message describing the profile on which the packet will be sent.

Thus, the transmitter's use of profiles, and hence the mix of modulation orders used on the plant, is based entirely on the current mix of active traffic as managed by the downstream QoS and scheduling engines.

ESTIMATION OF CAPACITY AND UTILIZATION

With the optimizations possible through using multiple profiles come many challenges. Among these challenges is the question: What is the "speed," or capacity (maximum throughput) of the channel? If this value is approximately given by:

$$\begin{aligned}&\text{speed in bits per second} =\\ &\quad(\text{modulation order per subcarrier}\\ &\qquad\qquad\times \text{subcarriers per symbol}\\ &\qquad\qquad\times \text{symbols per second})\\ &\qquad\qquad- \text{overhead}\end{aligned}$$

then it can clearly be seen that using multiple profiles with different modulation orders gives multiple results for capacity. The impact of this can be substantial. The difference between 16 QAM and 4096 QAM is a factor of three, and different profiles may have different numbers of data-carrying subcarriers per symbol due to the use of zero-bit-loaded subcarriers to account for different interferers.

Similarly, how can the utilization of such a channel be measured? Typically this would be given by an equation such as:

$$\text{Utilization} =$$

$$\frac{\text{bits carried over a given time interval}}{\text{speed in bits per second } \times \text{ interval duration}}$$

but since this relation includes the speed value from above, it suffers from the same dependency on modulation order.

Thus, a non-trivial question, both for designers and users of DOCSIS 3.1, is how to estimate downstream channel capacity and utilization in a meaningful way in the presence of multiple profiles. Because these metrics feed operator decisions about capital expenditures, it is vital to understand where the numbers come from and how they can be interpreted.

Normalization and the Reference Algorithm

The DOCSIS 3.1 OSSI addresses the first question in a simple, if limited, way: the "speed" of the downstream channel (as reported in the `ifSpeed` MIB object) is taken to be the throughput that could be achieved if all traffic on the plant used the profile with the highest throughput. This choice was made so that the behavior of `ifSpeed` for DOCSIS 3.1 would be consistent with previous DOCSIS versions, as well as with other interfaces such as Ethernet network ports, in the following ways:

- Reported `ifSpeed` is a constant (not time-varying) for a given channel configuration.
- Reported `ifSpeed` represents a maximum achievable rate; it is understood that actual throughput might approach but will never reach this value due to various factors (e.g., per-packet overhead on Ethernet interfaces).

One difference from other interfaces' treatment of `ifSpeed` is that DOCSIS 3.1 accounts for overhead due to FEC and other sources (discussed in more detail below).

Another important difference is that the actual speeds achieved on the channel may be much lower relative to the reported value when compared with interfaces such as DOCSIS 3.0 or wired Ethernet. In this respect, a DOCSIS 3.1 channel is more like a WiFi home networking channel, where the quoted speed can be achieved for devices close to the wireless access point, but distant devices may see significantly lower throughput.

This brings us back to the question of how the operator can extract meaningful metrics for channel utilization, so as to understand how heavily loaded the channel is and how loading might change under various projected scenarios.

To address this, the specification uses the concept of normalization and introduces a reference algorithm for making use of it.

To see how this concept is applied, it may be useful to imagine an example system using Profile A and Profile D. If Profile A is configured for 8 bits per symbol per subcarrier, and Profile D is configured for 12 bits per symbol per subcarrier, then all other things being equal (and discounting overhead for purposes of this example), the maximum possible throughput using Profile A would be two-thirds of that possible using Profile D. An inverse statement could also be made that a certain number of bytes sent using Profile A will consume 50% more of the channel than the same number of bytes sent using Profile D.

The above relationship can be used to estimate channel utilization in the presence of packets from both Profiles A and D. Bytes sent on Profile A "count" towards utilization 50 percent more than bytes sent on Profile D. If the `ifSpeed` of the interface is the throughput of Profile D in units of bits per second, then the fraction of the channel which is utilized over some time interval T can be expressed as

$$\text{Utilization} = \frac{(\text{bytes sent on Profile D during } T) + 1.5(\text{bytes sent on Profile A during } T)}{\left(\frac{\text{ifSpeed}}{8}\right) \times (\text{duration of } T)}$$

In a real system, more than two profiles will probably be in use, and the relationship between their rates is probably not a simple fraction. Nonetheless, it is possible to generalize the above equation by defining a "normalization factor" for each profile:

$$\text{NF}(\text{Profile } P) = \frac{\text{ifSpeed}}{\text{throughput of } P}$$

By definition, the normalization factor of the highest-throughput profile on the channel is 1; other profiles will have normalization factors larger than 1.

For a set of Profiles P, the fraction of the channel which is utilized during some time interval T is then expressed as

$$\text{Utilization} = \sum_{P} \frac{(\text{bytes sent on } P \text{ during } T) \times \text{NF}(P)}{\left(\frac{\text{ifSpeed}}{8}\right) \times (\text{duration of } T)}$$

This equation is reflected in the reference algorithm given in the DOCSIS 3.1 OSSI specification for calculating utilization. It behaves as we would expect of such a metric. It reflects the fraction of the channel that is occupied, even if the resulting throughput in terms of bytes or bits is reduced due to the use of lower-throughput profiles. For instance, if traffic from a lower-throughput profile completely fills the channel, the actual rate will be lower than the rate reported in ifSpeed, and the resulting channel byte counts will reflect this, but the utilization value of 100 percent will correctly indicate that the channel is full and cannot carry any additional bytes.

How accurate is this formula? It turns out that it gives excellent results for most practical cases, but there are potential sources of inaccuracy due to non-ideal aspects of the system. Some can be accounted for in calculation, while others are dynamic and not directly visible.

To understand these items, it is first useful to study the sources of overhead on a DOCSIS 3.1 channel, particularly those related to FEC.

FEC Encoding on Multi-Profile Downstreams

Like previous versions of DOCSIS, the DOCSIS 3.1 downstream makes use of FEC to provide error correction capability. The cost of this capability is a loss in channel capacity due to the overhead of the parity bits that FEC adds to the data stream. In DOCSIS 3.1, a combination of block codes is used. An LDPC code contributes most of the error correction capability (and overhead), while a BCH outer code surrounds each LDPC block. A total of 1,968 parity bits are provided for each block of up to 14,232 information bits. The combination of information and parity bits is termed a "codeword." Codewords may be "shortened" by providing fewer information bits, but the number of parity bits per codeword is constant for these codes, so the efficiency of shortened codewords is lower than that of full-length ("full") codewords. Specifically, the efficiency of a full codeword is about 87.8 percent, while the efficiency of a shortened codeword can be virtually any number lower than this (as codewords down to one information bit are allowed).

Data packet sizes are, of course, variable, and in any given queue of packets to be transmitted, packet boundaries will almost never align with full codeword boundaries. To avoid constant codeword shortening, "streaming" of packets across codeword boundaries is typically performed. As much MAC-layer data as will fit is placed into a codeword, and if this results in only a partial packet being sent, the remainder of the packet is carried over to the next codeword. Two bytes at the beginning of each codeword are used as a "header" which provides a pointer for locating the first packet boundary within the codeword.

If only a single profile were in use, packets could be streamed continuously into codewords and codeword shortening would never be needed. However, the use of multiple profiles introduces new complexity. In order to properly decode an LDPC/BCH codeword, a receiver must be able to correctly receive all bits of the codeword. But it is the nature of a multi-profile system that not all receivers can decode all profiles. If a codeword were allowed to contain bits from packets being sent on different profiles, some receivers would not be able to decode it, and these receivers would lose the entire codeword, including the portion sent on profiles they can receive.

This problem is solved by assigning codewords to profiles. When the transmitter wants to send a particular packet, it creates a codeword and assigns that codeword to use the profile of the packet to be transmitted. Other packet data using the same profile can also be included in the codeword, but if no such data is available, the codeword must either be filled with MAC-layer "stuff bytes" or shortened. It cannot be filled with packet data using other profiles.

This ensures that each receiver can fully decode all codewords it needs, but how does the receiver know which codewords use which profiles, and where in the data stream they are located? DOCSIS 3.1 solves this by devoting a few subcarriers out of each OFDM symbol to communicating "Next Codeword Pointers," or "NCPs." NCPs indicate the start location of each codeword by means of a pointer to a subcarrier number, and also state which profile the codeword uses. NCPs use a very low modulation order (configurable from QPSK through 64 QAM) and are placed on subcarriers which can be received by all modems. When the transmitter creates each codeword, it also creates a corresponding NCP on the symbol during which the codeword begins.

NCPs consume subcarriers that might otherwise be used for data, so they contribute to the total overhead lost from the channel. The number of NCPs may vary from one OFDM symbol to the next, depending on how many codewords are included in that symbol. This means that the amount of NCP overhead may also vary. On most channels, the amount of overhead is 3 percent or less.

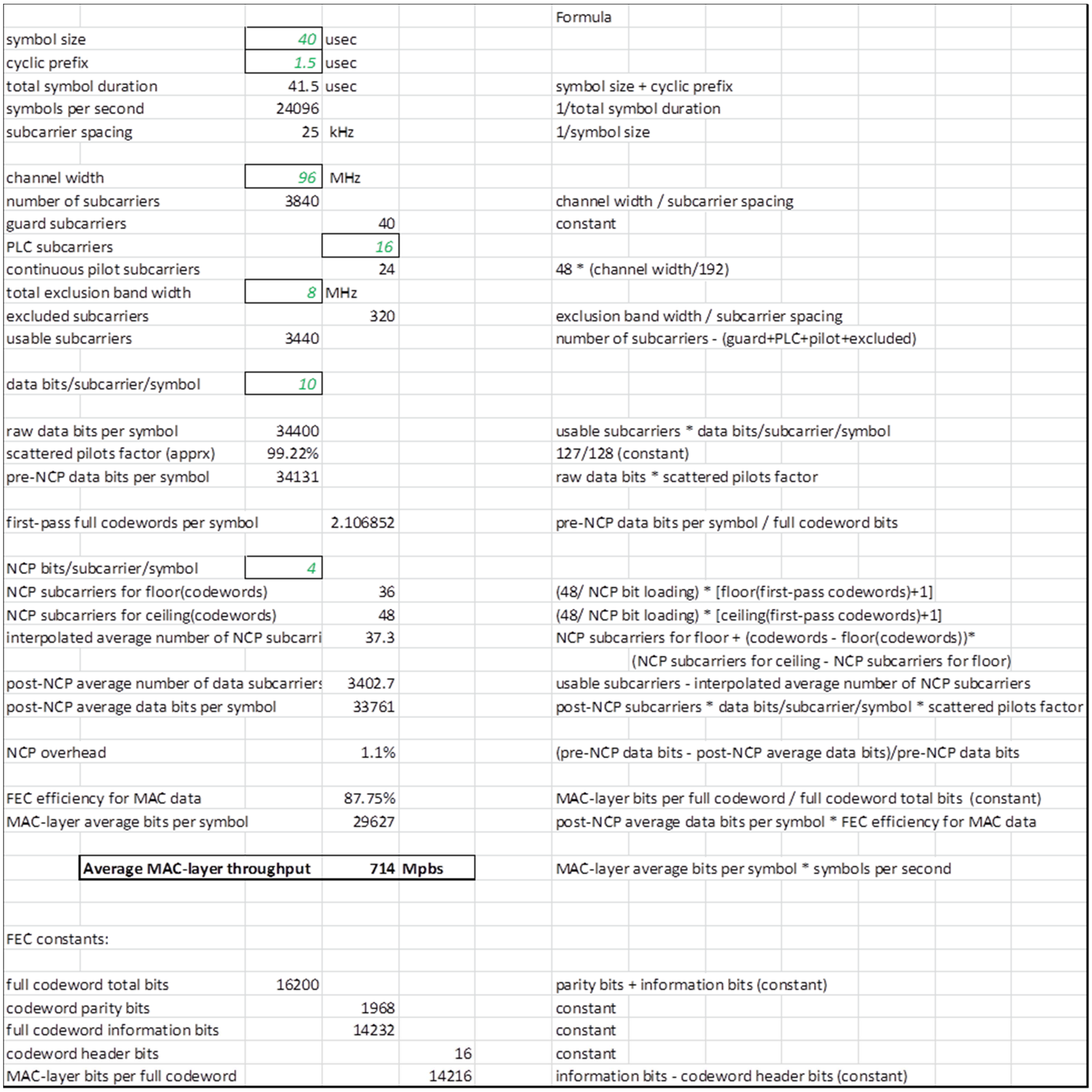
Throughput Calculation for a Single Profile

Besides FEC and NCP overhead, there are several other items that must be included if overhead is to be accounted for in calculating throughput.

On every OFDM channel, some subcarriers are dedicated to functions other than carrying data, and the transmitter must avoid using these subcarriers. PLC (PHY Link Channel) and continuous pilot subcarriers are defined in the OCD message and remain in a fixed location once defined. These contribute to channel overhead, but do not vary from symbol to symbol.

In contrast, scattered pilots occur on a different set of subcarriers every symbol, in a pattern that repeats every 128 symbols. Every potentially data-carrying subcarrier on the channel will contain a scattered pilot at some point. When that happens, the subcarrier is not used for data and whatever number of bits might normally be carried on it is lost from the total available on the symbol. The exact amount of loss this represents depends on the modulation order assigned to that subcarrier on whatever profile might have been used on that subcarrier, had it been used. Of course, this is impractical to predict, and so this overhead is difficult to quantify on an instantaneous basis. However, it can be accounted for in a calculation that assumes that same profile is used for an extended time period (at least 128 symbols).

Based on all of the above items, and with careful study of the PHY specification and references, it is possible to calculate the post-overhead maximum throughput using a single profile to a very high degree of accuracy. A few assumptions must be made:



				Formula
symbol size	40	usec		
cyclic prefix	1.5	usec		
total symbol duration	41.5	usec		symbol size + cyclic prefix
symbols per second	24096			1/total symbol duration
subcarrier spacing	25	kHz		1/symbol size
channel width	96	MHz		
number of subcarriers	3840			channel width / subcarrier spacing
guard subcarriers		40		constant
PLC subcarriers		16		
continuous pilot subcarriers		24		48 * (channel width/192)
total exclusion band width	8	MHz		
excluded subcarriers		320		exclusion band width / subcarrier spacing
usable subcarriers	3440			number of subcarriers - (guard+PLC+pilot+excluded)
data bits/subcarrier/symbol	10			
raw data bits per symbol	34400			usable subcarriers * data bits/subcarrier/symbol
scattered pilots factor (apprx)	99.22%			127/128 (constant)
pre-NCP data bits per symbol	34131			raw data bits * scattered pilots factor
first-pass full codewords per symbol		2.106852		pre-NCP data bits per symbol / full codeword bits
NCP bits/subcarrier/symbol	4			
NCP subcarriers for floor(codewords)		36		(48/ NCP bit loading) * [floor(first-pass codewords)+1]
NCP subcarriers for ceiling(codewords)		48		(48/ NCP bit loading) * [ceiling(first-pass codewords)+1]
interpolated average number of NCP subcarri		37.3		NCP subcarriers for floor + (codewords - floor(codewords))* (NCP subcarriers for ceiling - NCP subcarriers for floor)
post-NCP average number of data subcarrier		3402.7		usable subcarriers - interpolated average number of NCP subcarriers
post-NCP average data bits per symbol		33761		post-NCP subcarriers * data bits/subcarrier/symbol * scattered pilots factor
NCP overhead		1.1%		(pre-NCP data bits - post-NCP average data bits)/pre-NCP data bits
FEC efficiency for MAC data		87.75%		MAC-layer bits per full codeword / full codeword total bits (constant)
MAC-layer average bits per symbol		29627		post-NCP average data bits per symbol * FEC efficiency for MAC data
Average MAC-layer throughput		**714**	**Mpbs**	MAC-layer average bits per symbol * symbols per second
FEC constants:				
full codeword total bits	16200			parity bits + information bits (constant)
codeword parity bits		1968		constant
full codeword information bits		14232		constant
codeword header bits			16	constant
MAC-layer bits per full codeword			14216	information bits - codeword header bits (constant)

Table 1: Simplified Calculation of MAC-Layer Throughput for a Single Profile

- The system will be observed over a relatively long interval (say, seconds or more), guaranteeing that symbol-to-symbol variations are averaged out.
- The system is fully loaded with packets for a single profile so that the transmit queue never underruns and the transmitter never needs to switch profiles.
- Full codewords are always used (although the specification does not require it, a quality transmitter implementation is expected to use full codewords under the conditions described).

The resulting calculated throughput describes the amount of DOCSIS MAC-layer data that can be carried on a single profile, accounting for PHY-layer overhead, but not subtracting out MAC-layer overhead such as DOCSIS® MAC headers. This is the value that would be used in the formulas given above for the normalization factor and channel utilization.

The complete details of such a calculation are beyond the scope of this paper. However, a simplified example is shown in Table 1. The example uses a 96 MHz wide downstream channel with an exclusion band of 8 MHz, which might occur if it were necessary to avoid some other service using a 6 MHz carrier (allowing a 1 MHz guard band on each side). For simplification, the example assumes a constant bit loading of 1024 QAM (10 bits per symbol) on all subcarriers. It also uses approximations for NCP and scattered pilot overhead, which have high validity if the system is observed over many symbols. The interpolation approach used for NCP overhead gives excellent results if the number of codewords per symbol is not too close to an integer, as in this example.

IMPACT OF MULTIPLE PROFILES

Multiple Profiles and Convergence Layer Functionality

In a real system, more than one profile will be used at a time. In DOCSIS 3.1, the downstream convergence layer is primarily responsible for implementing multi-profile transmission.

As previously discussed, each FEC codeword is assigned to use a particular profile, but packet boundaries rarely align with codeword boundaries. What should a transmitter do if it does not have enough packet data available on a particular profile to fill a codeword? This could happen either because the packet(s) in the queue are too short, or because, after filling some number of codewords in the streaming fashion previously described, there is a partial packet remaining that does not fill a codeword.

In this situation, a transmitter has two choices: it can shorten the codeword or it can wait for more packets to arrive for the profile in question. Either option has drawbacks. Codeword shortening has a high cost in efficiency: a 64-byte packet sent in a codeword by itself is about 20 percent efficient. If even a small fraction of packets is sent in this way, the loss for the channel as a whole may be considered significant. However, waiting for more data adds latency to the packets which are already available. Waiting might also violate the QoS principles being enforced at higher layers of the CMTS. While the transmitter is holding a packet on one profile, packets that may have lower QoS priority are being sent ahead of it on other profiles. The fact that a user happens to be on a profile for which little other traffic is available should not be allowed to alter the QoS guarantee to that user.

The solution adopted by DOCSIS 3.1 is a hybrid of these two choices. The transmitter is allowed to hold data for a certain small amount of time in hopes of receiving more data to fill a codeword. Within that window of time, packets from different profiles may be transmitted in a different order than they were originally queued (although ordering must be maintained within a single profile). Once the maximum wait time is reached, the transmitter is expected to shorten codewords, giving up efficiency in exchange for meeting the desired latency bounds. A typical latency bound is 200 microseconds, although this value can vary, as will be discussed in more detail below. The intent is that the latency bounds are small enough so as not to interfere with QoS guarantees, although they may be large enough to impact latency budgeting for some services.

Figure 1, taken from the DOCSIS 3.1 PHY specification, shows a sample system architecture used for describing and specifying (though not necessarily implementing) this functionality. Packet forwarding and QoS, including shaping to the instantaneous channel rate, are performed in the leftmost blocks of the diagram. Packets are queued "single file" for transmission and passed to the convergence layer, which contains per-profile buffers that are used to hold data while waiting to fill a codeword. The wait time is expected to be constrained to match the configurable latency targets. The "codeword builder," shown within the PHY block, is the primary decision-maker. It makes a "best guess" tradeoff to decide which profile to use for each codeword, attempting to maximize efficiency without exceeding latency targets.

Importantly, there are no specification requirements stating when a codeword builder sends shortened codewords rather than full codewords. An implementation is free to send shortened codewords at any time, for any reason. As an example, when a channel is lightly loaded, an implementation may intentionally send shortened codewords to increase channel robustness (shorter codewords are more robust since there are more parity bits relative to the number of information bits). As another example, an implementation may choose to prioritize latency over efficiency by sending packets

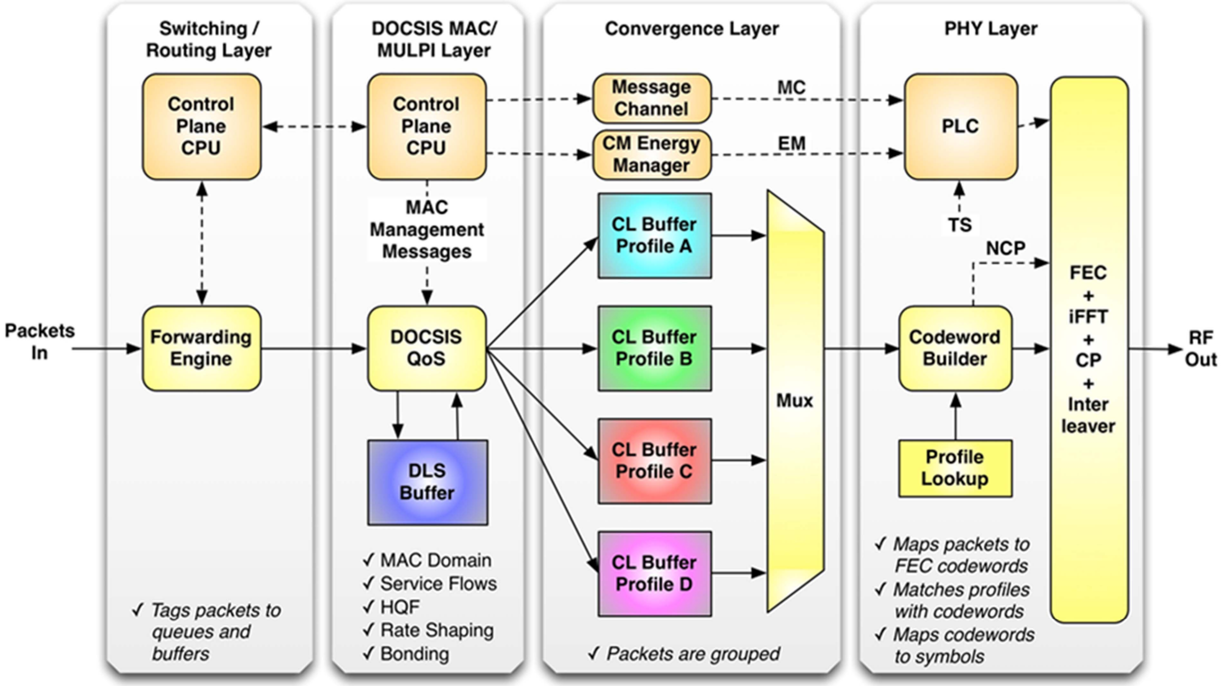


Figure 1: Sample System Architecture for Multi-Profile CMTS Downstream Transmitter (from DOCSIS PHY specification)

with less wait time when the channel is lightly loaded. In both cases, since the channel is not full, codeword shortening may be seen as a good use of otherwise unutilized capacity on the channel. If the load increases, the codeword builder can decide not to perform these types of codeword shortening. In other words, the bandwidth used by such cases of codeword shortening can be recovered if it is needed for actual data traffic.

Thus, it is expected (though not stated in the specification) that under conditions of heavy load, a codeword builder will use mostly full codewords. This assumption is built into the utilization reference algorithm, which contains no allowance for overhead due to codeword shortening. If the assumption is incorrect, then the reference algorithm will be less accurate than we would like. Thus, it is important to study conditions under which this assumption might not hold.

Codeword Builder Behavior Under Worst-Case Conditions

First consider a case in which the system is 100 percent loaded with traffic, all for a single profile. As already discussed, in this case it is expected that the codeword builder will virtually always use full codewords. In a steady-state condition, the buffer for the profile is always full, so full-size codewords can be created with no wait time.

Next, consider a case in which the system is 100 percent loaded with traffic, but this time for two different profiles. In this case, the codeword builder behavior is affected by the very-short-term traffic mix at the input to the convergence layer, since this will determine the fullness of the two convergence layer profile buffers. At one extreme, packets for the two profiles could arrive in alternating fashion – for instance, one packet for Profile A, immediately followed by a same-sized packet for Profile B, and so forth. In this case, the steady-state condition for the codeword builder would be that both buffers are equally full, and so full codewords can always be created. Each profile has to wait up to the duration of one codeword while the other profile occupies the channel, but this time is short compared to the latency targets.

At the other extreme, packet distribution for the two profiles could be extremely unbalanced at the codeword builder input; one profile – say Profile A – could dominate, with only short, infrequent, widely-spaced packets for Profile B. This situation is actually quite plausible. If the only traffic active on Profile B is, say, a Voice-over-IP flow, the packets on Profile B will be short and the QoS engine will ensure that they are spaced according to the packetization interval of the service. This interval (10 or 20 milliseconds) is much longer than the typical sub-millisecond latency target of the codeword builder. Even for services such as best-effort downloads and web browsing, the traffic shaping function of the DOCSIS QoS will ensure that packets for the flow are not clumped together, but are more-or-less evenly spaced.

Thus, any time a profile carries traffic for only a small number of Service Flows, a "widely spaced packets" scenario is likely to occur.

In this scenario, the codeword builder will primarily create full codewords for the dominant profile – say, Profile A. When a packet for Profile B arrives, it might attempt to wait for more data on Profile B, but that data will not arrive in time due to the packet spacing enforced by the QoS engine. The codeword builder will be forced to send only shortened codewords for Profile B.

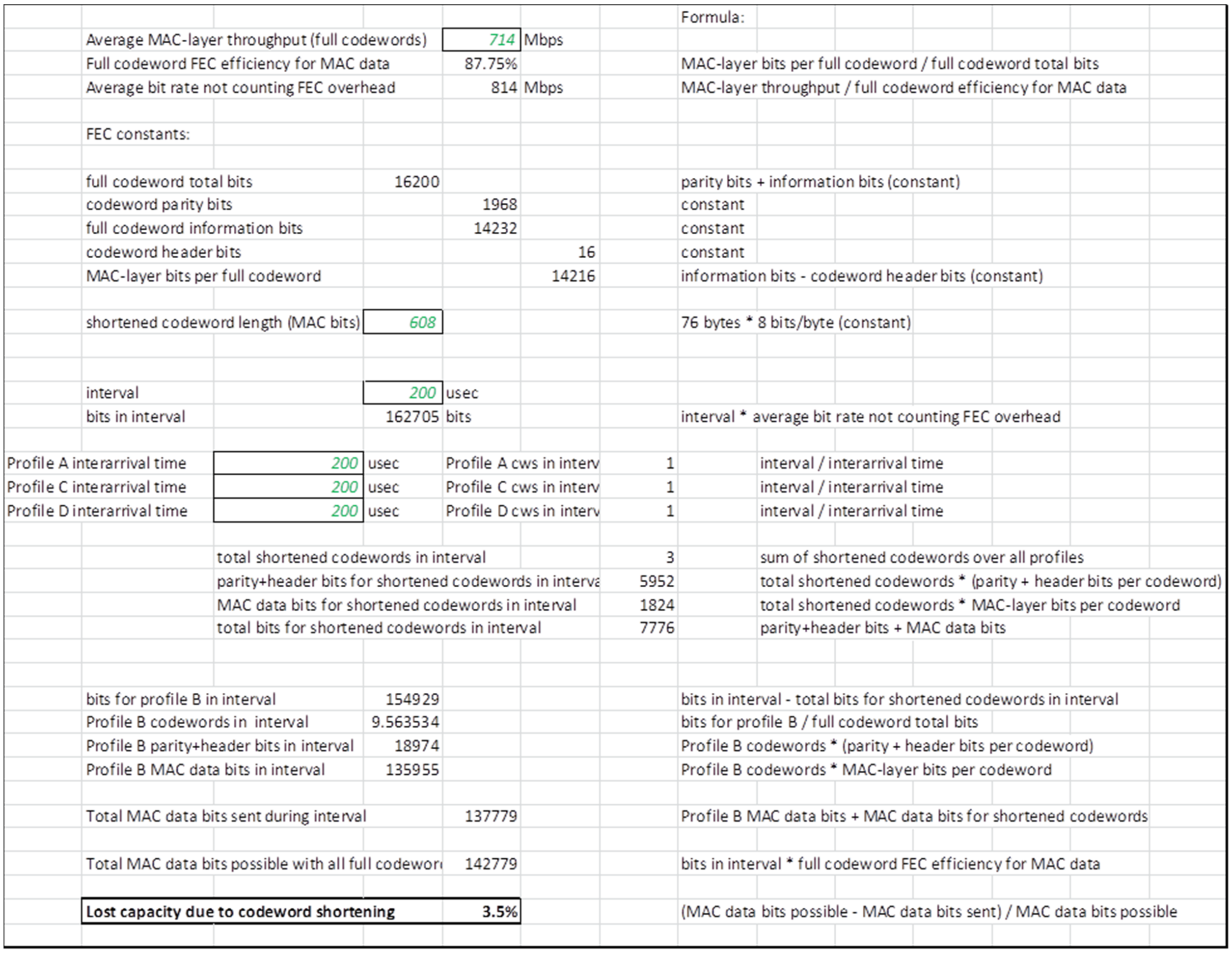
The shortening of codewords in this case represents an actual loss of capacity for the system due to the use of multiple profiles. Had the traffic been sent on Profile A instead, it could have been included as part of a full codeword. However, with the traffic occurring as isolated packets on a different profile, there is no option but to use lower-efficiency shortened codewords. In other words, the bandwidth consumed by codeword shortening cannot be recovered if the shortening is dictated by the traffic pattern and latency targets.

The worst-case loss for this scenario occurs when the spacing of packets on Profile B is infinitesimally larger than the latency target for Profile B. In this case, the codeword builder waits up to the latency target, then releases a codeword containing a single packet – then immediately thereafter, a new packet arrives, which in turn is delayed up to the latency target.

Now suppose that additional profiles are configured on the system, up to the maximum of 16. The worst case here occurs when one of the profiles – say Profile A – dominates the traffic, and all of the other 15 profiles exhibit the worst-case behavior described in the example above for Profile B. In this case, during every time interval equal to the duration of the latency target, 15 codewords must be shortened.

With this worst-case pattern in mind, it is possible to calculate the capacity lost due to required codeword shortening for a variety of parameter sets.

Table 2 illustrates the complete calculation for an example case. In this example, to



					Formula:
Average MAC-layer throughput (full codewords)	714	Mbps			
Full codeword FEC efficiency for MAC data	87.75%				MAC-layer bits per full codeword / full codeword total bits
Average bit rate not counting FEC overhead	814	Mbps			MAC-layer throughput / full codeword efficiency for MAC data
FEC constants:					
full codeword total bits	16200				parity bits + information bits (constant)
codeword parity bits	1968				constant
full codeword information bits	14232				constant
codeword header bits	16				constant
MAC-layer bits per full codeword	14216				information bits - codeword header bits (constant)
shortened codeword length (MAC bits)	608				76 bytes * 8 bits/byte (constant)
interval	200	usec			
bits in interval	162705	bits			interval * average bit rate not counting FEC overhead
Profile A interarrival time	200	usec	Profile A cws in interv	1	interval / interarrival time
Profile C interarrival time	200	usec	Profile C cws in interv	1	interval / interarrival time
Profile D interarrival time	200	usec	Profile D cws in interv	1	interval / interarrival time
			total shortened codewords in interval	3	sum of shortened codewords over all profiles
			parity+header bits for shortened codewords in interva	5952	total shortened codewords * (parity + header bits per codeword)
			MAC data bits for shortened codewords in interval	1824	total shortened codewords * MAC-layer bits per codeword
			total bits for shortened codewords in interval	7776	parity+header bits + MAC data bits
bits for profile B in interval	154929				bits in interval - total bits for shortened codewords in interval
Profile B codewords in interval	9.563534				bits for profile B / full codeword total bits
Profile B parity+header bits in interval	18974				Profile B codewords * (parity + header bits per codeword)
Profile B MAC data bits in interval	135955				Profile B codewords * MAC-layer bits per codeword
Total MAC data bits sent during interval	137779				Profile B MAC data bits + MAC data bits for shortened codewords
Total MAC data bits possible with all full codewor	142779				bits in interval * full codeword FEC efficiency for MAC data
Lost capacity due to codeword shortening	**3.5%**				(MAC data bits possible - MAC data bits sent) / MAC data bits possible

Table 2: Sample Calculation of Codeword Shortening Capacity Loss Using Worst-Case Traffic

simplify the calculation, it is assumed that all profiles have the same bit rate. Although unlikely, this could occur on a real channel if the profiles targeted similar SNR but different exclusion bands. Four profiles are assumed to be in use, each with the same parameters as in Table 1. All four profiles have a latency target of 200 microseconds; thus, 200 microseconds is the interval over which the analysis is performed. It is assumed that Profile B dominates the traffic and uses only full codewords. The other profiles carry only a single 64-byte packet plus DOCSIS headers (76 bytes total) every 200 microseconds, requiring the use of a single shortened codeword once per interval.

Starting from the pre-FEC throughput of the profiles, the calculation determines the total number of shortened codewords and full codewords which are sent on a fully loaded channel during the analysis interval. This, in turn, defines the total number of FEC parity and codeword header bits. Subtracting these bits from the total available during the interval gives the number of bits which carry MAC-layer data. The difference between this value and the amount of MAC-layer data that could be carried using only full codewords represents the lost capacity on the channel as a result of using multiple profiles with a worst-case traffic pattern. In this example, the lost capacity amounts to about 3.5 percentof the total channel bandwidth.

In interpreting this result, recall that it is based on a worst-case traffic pattern in which traffic across profiles is extremely unbalanced, and the interarrival time of packets on a lightly-loaded profile is essentially equal to the profile's latency target. Although a pattern of widely-spaced packets is probably typical for lightly-loaded profiles, it is unlikely that the interarrival time will be so consistent. Sometimes it will be larger, in which case codewords will be shortened less often. Sometimes it will be smaller, in which case some aggregation of data packets can take place, resulting in shortened codewords that are longer and hence more efficient than those shown in the analysis.

Thus, the result of a calculation such as that shown in Table 2 should be taken as an upper bound on lost capacity which is very unlikely to be reached in a real system observed over a typical time period (seconds or more).

So how can the operator know what actual capacity loss might be taking place in a real system? Ideally, it could be reflected in reported utilization statistics. However, as previously mentioned, this loss of capacity is not accounted for in the reference algorithm for calculating utilization.

Importantly, a real-world implementation of a codeword builder may have little or no ability to account for this loss. The reasons may vary with implementation, but some considerations are:

- The codeword builder algorithm may consist of a large number of decision-making rules, and thus it may lack the ability to determine a "reason" why a particular rule set resulting in certain codeword sizes was invoked at any given instant.
- The codeword builder algorithm may need to use predictive techniques in order to meet latency targets; for example, it cannot wait until the last possible instant to decide to shorten codewords for all 15 different profiles, since 15 codewords will not fit in one OFDM symbol. Determining whether a different prediction would have given a "better" result is beyond the scope of an implementation.

Thus, it is entirely possible that capacity could be lost without any way of making this visible to the operator.

The DOCSIS OSSI specification allows a CMTS to deviate from the reference algorithm when reporting utilization by incorporating any vendor-specific information it may have that points to actual capacity loss in the system due to codeword shortening. Vendor innovation in this area might lead to CMTS reporting that improves upon the reference algorithm.

Regardless, the best way for an operator to cope with this potential capacity loss is probably to avoid it in the first place. This is most easily done by configuring profiles so that traffic is relatively balanced among them. Although this does not guarantee that codeword shortening will never be needed (since even when the long-term traffic mix is balanced, the very-short-term packet mix presented to the input of the convergence layer could be virtually anything), it does prevent a pattern of repeated codeword shortening such as that studied in the above analysis. Over time, occasional intervals requiring codeword shortening are likely to be balanced out by those in which traffic aggregation allows full codewords to be used so that the net capacity loss becomes negligible.

If it not possible to avoid potentially problematic configurations, the next best thing is for the operator to understand the contributing factors and their potential impact. It has already been seen that traffic imbalance across profiles is a major "red flag." Other contributors include the number of profiles, latency targets, and channel bandwidth. The methods of the previous analysis can be used to explore the effects of these factors.

Analysis of Multiple Configurations

The DOCSIS 3.1 specification recognizes the relationships described above and offers recommendations for latency targets based on channel bandwidth and number of profiles. These recommendations are shown in Table 3, which is taken from the DOCSIS 3.1 MULPI document (table 7-12 in that document).

It is recognized that when channel bandwidth is narrower, fewer full codewords can be transmitted in a given amount of time, making it harder for the codeword builder to achieve a certain latency without codeword shortening. Thus, latency targets are higher for narrow channels. Higher targets are also suggested for larger numbers of profiles.

In order to make it possible for an operator to deliver latency-sensitive services using configurations with higher targets, the specification suggests that some profiles have lower latency targets than others. Modems receiving a low-latency service are assigned profiles from among the limited subset of low-latency profiles, while other modems can be assigned profiles selected from the full set.

Total CMTS Profiles	**Minimum Profiles per Latency Target**	**Max Latency (us) based upon OFDM Channel Bandwidth (MHz)**			
		24	48	96	192
4	1	600	400	200	200
	3	800	400	200	200
8	2	800	400	200	200
	6	2400	1600	800	400
16	4	1600	1200	800	400
	12	3200	2400	1600	800

Table 3: Codeword Builder Latency (from DOCSIS MULPI specification)

The values in Table 3 are only recommendations, and vendors or operators may choose to configure different values. In particular, for some combinations, the latencies given are quite large. Values greater than a few hundreds of microseconds present challenges for low-latency services and may also impact DOCSIS performance in general. Operators may choose to observe these latency targets, configure more aggressive targets at a cost in efficiency, or avoid combinations for which the table gives high recommended values.

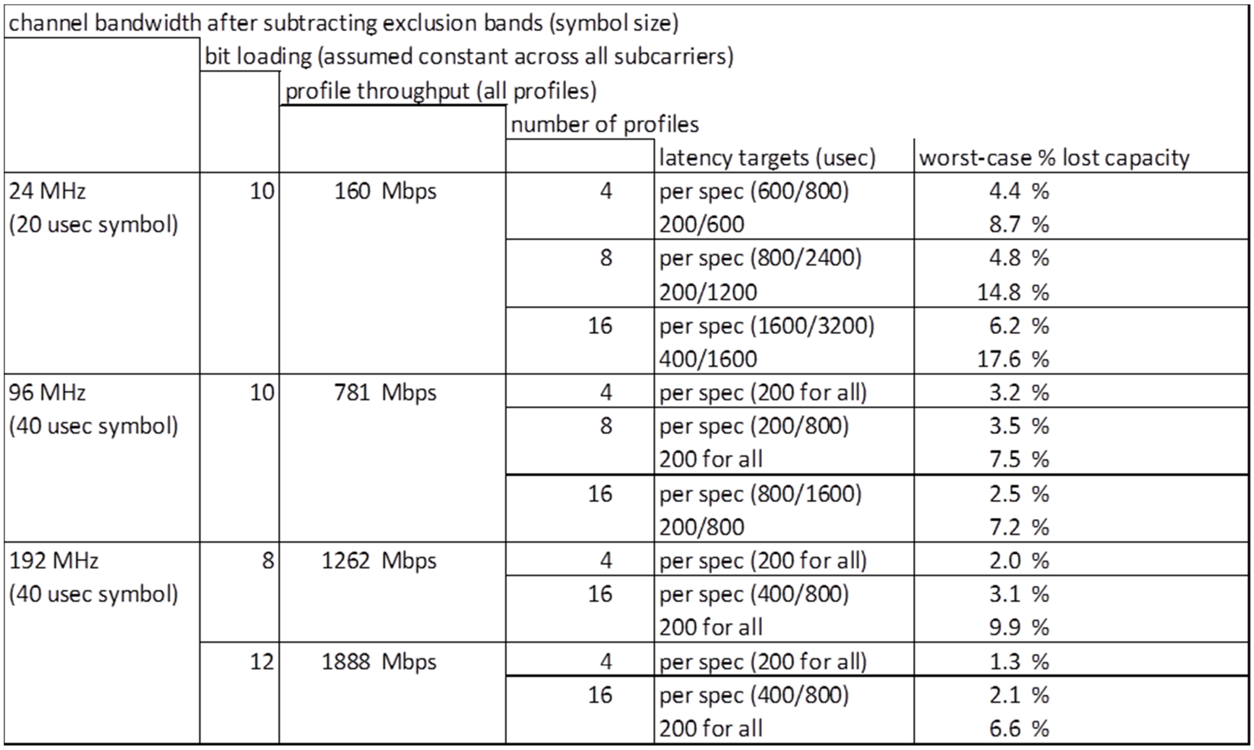
To analyze cases in which different profiles have different latency targets, the calculation of Table 2 can be done using an analysis interval equal to the least common multiple of the latency targets. Profiles with targets shorter than the interval will have multiple codewords shortened during the interval.

It is also possible to incorporate different rates per profile by incorporating a normalization factor. Such calculations are beyond the scope of the current paper.

Table 4 summarizes the results of calculations performed for various combinations of number of profiles, channel bandwidth, profile throughput (assumed to be the same across all profiles), and latency targets. Where two different targets are given, the dominant profile is assumed to use the larger value. The worst-case traffic pattern described above is assumed for these calculations. This is only a small number of possible options, but should give an idea of the type of results that can be expected.

Several points can be gleaned from the data in Table 4:

- Increasing latency targets beyond those recommended in MULPI has serious consequences under worst-case traffic conditions, often tripling the loss due to codeword shortening.
- Low-rate channels present the greatest challenge for a latency vs. efficiency tradeoff. Performance is improved if the channel bandwidth is increased, and also if the average bit loading on the channel is increased – i.e., if



channel bandwidth after subtracting exclusion bands (symbol size)	bit loading (assumed constant across all subcarriers)	profile throughput (all profiles)	number of profiles	latency targets (usec)	worst-case % lost capacity
24 MHz (20 usec symbol)	10	160 Mbps	4	per spec (600/800)	4.4 %
				200/600	8.7 %
			8	per spec (800/2400)	4.8 %
				200/1200	14.8 %
			16	per spec (1600/3200)	6.2 %
				400/1600	17.6 %
96 MHz (40 usec symbol)	10	781 Mbps	4	per spec (200 for all)	3.2 %
			8	per spec (200/800)	3.5 %
				200 for all	7.5 %
			16	per spec (800/1600)	2.5 %
				200/800	7.2 %
192 MHz (40 usec symbol)	8	1262 Mbps	4	per spec (200 for all)	2.0 %
			16	per spec (400/800)	3.1 %
				200 for all	9.9 %
	12	1888 Mbps	4	per spec (200 for all)	1.3 %
			16	per spec (400/800)	2.1 %
				200 for all	6.6 %

Table 4: Summary of Results of Capacity Loss Calculation for Selected Parameter Sets

the channel is dominated by modems using a higher modulation order.

- The best performance in both latency and efficiency is achieved with smaller numbers of profiles. Not only do these cases have the lowest potential capacity loss in the presence of worst-case traffic, but in a real system, it is easier to balance traffic across profiles when there are only a few of them, making a worst-case traffic pattern much less likely.

RECOMMENDATIONS

Based on the above discussion and analysis, it is recommended that operators:

- Resist the urge to configure profiles that are usable by only one or a few modems on the plant. Although this may appear to offer better performance by allowing use of modulation orders more tailored to individual devices, it could have the opposite effect by creating suboptimal traffic patterns and forcing codeword shortening that reduces channel efficiency. An ideal profile set would have similar amounts of traffic on each profile, with enough different Service Flows to make aggregation likely at the codeword builder.
- Become familiar with the MULPI recommendations for profile latency targets. Ideally, configurations which result in high recommended latency targets should be avoided. If this is not possible, the operator should be prepared to tolerate either high latency (if recommended targets are observed) or low efficiency (if more aggressive targets are configured).
- Be aware that capacity loss due to codeword shortening may not be reflected in reported system statistics. In particular, operators should look for imbalance in traffic load across profiles, especially in systems using higher numbers of profiles and/or narrower channels. If these conditions are present, reported utilization values may contain inaccuracy of as much as a few percent.
- Rely on more than just a single statistic when planning node splits or plant upgrades. Utilization statistics remain an excellent guide, but they may vary over time depending on which modems are active, even if the plant has been configured to avoid hidden capacity loss. When utilization nears a value of interest, operators should monitor other statistics, such as those reflecting buffer fullness or packet loss, to determine whether a channel is overloaded.
- Incorporate the concept of normalization into network planning. When a node is split, subscribers added, SLAs increased, and so forth, it should be recognized that devices using lower-throughput profiles will impact the network more than devices using higher-throughput profiles, even if the SLA to both devices is the same.

SUMMARY AND CONCLUSIONS

This paper has offered an overview of how multiple modulation profiles may be defined and used in a DOCSIS 3.1 system. The concept of normalization was introduced, showing that the contribution to channel utilization of bytes sent on a profile is proportional to the reciprocal of the profile’s maximum throughput. In examining the details of FEC codeword overhead and scheduling of profiles, it became clear that there are certain cases where inefficiencies can result in lost channel capacity. This occurs when some profiles are lightly loaded and carry few Service Flows, so that the operation of DOCSIS QoS results in infrequent, widely-spaced packets on those profiles. This traffic condition forces the transmitter to use less-efficient shortened codewords, reducing the bandwidth available for carrying data. This lost capacity may not be reflected in reported utilization statistics.

It is important for the operator to be aware of the conditions under which this capacity loss can occur, ideally to configure the plant so as to avoid it, or, if this is not possible, to know when it might be taking place so that it can be accounted for in network evaluation and planning.

The best performance is achieved when small numbers of profiles are used on wide channels with high rates. When the number of profiles is larger or channel bandwidth is smaller, a tradeoff between latency and efficiency must be made. The DOCSIS 3.1 MULPI specification recommends latency values which can be expected to give good efficiency. If more aggressive latency targets are configured, efficiency can be expected to drop sharply.

For any configuration, the chance of capacity loss is minimized if traffic is distributed more-or-less evenly across profiles. This maximizes the chance that enough packet aggregation will occur to prevent forced codeword shortening. Profiles that serve only a few modems are the most likely to require codeword shortening that results in reduced capacity.

With an understanding of the factors at work and some attention to system configuration and monitoring, operators can deploy multiple modulation profiles with high efficiency and account for the impact of this feature on network utilization.

ACRONYMS AND ABBREVIATIONS

CM – Cable Modem
CMTS – Cable Modem Termination System
DBC – Dynamic Bonding Change
DOCSIS® – Data Over Cable Service Interface Specifications
DPD – Downstream Profile Descriptor
FEC – Forward Error Correction
MAC – Media Access Controller
MAP – Bandwidth Allocation MAP
Mbps – Megabits per second
MHz – Megahertz
MIB – Management Information Base
MULPI – MAC and Upper Layer Protocols Interface
OCD – OFDM Channel Descriptor
ODS – OFDM Downstream Spectrum
OFDM – Orthogonal Frequency-Division Multiplexing
OPT – OFDM Downstream Profile Test
OSSI – Operations Support System Interface
PHY – Physical layer
PLC – PHY Link Channel
QAM – Quadrature Amplitude Modulation
QPSK – Quadrature Phase-Shift Keying
QoS – Quality of Service
RxMER – Receive Modulation Error Ratio
SLA – Service Level Agreement
SNMP – Simple Network Management Protocol
SNR – Signal to Noise Ratio
UCD – Upstream Channel Descriptor

REFERENCE ACQUISITION

DOCSIS® specifications: Cable Television Laboratories, Inc. “CableLabs - Specifications” [Online] Available: http://www.cablelabs.com/specs