The Utilization Profiles of the CCSDS Unified Space Link Protocol (USLP)

G.Kazz,[1] and E.Greenberg[2]
Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA, 91109

The purpose of this paper is to identify the utilization profiles for interfacing the Data Protocol Sub-layer using the Unified Space Link Protocols (USLP) (reference 1) with the space link coding procedures as specified in the CCSDS Coding & Synchronization Blue Books (references 2 through 5), used in both telecommand and telemetry applications. This paper describes how the USLP Protocol utilizes the coding and synchronization sublayer to support:

a. Direct to Earth (DTE) telemetry links for engineering and science data

b. Direct to Earth (DTE) telemetry links for very high rate science data

c. Direct from Earth (DFE) command, sequencing and flight software loads

d. Space to Space Links (Proximity) utilized by orbiters for data exchange to/from surface bound assets

The CCSDS has divided the functions of the Data Link Layer into two sublayers: the Data Link Protocol Sublayer (DLP-SL) and the Coding and Synchronization Sublayer (CS-SL). The Data Link Protocol Sublayer (DLP-SL) interfaces to the users, accepting the data that is to be transported, on the sending side of the link, and delivering that data on the receiving end. The Transfer Frame is the data unit that is transferred across the Data Link Protocol Sublayer and the Coding and Synchronization Sublayer boundary. The Coding and Synchronization Sublayer (CS-SL) provides the encoding, randomization, and frame synchronization functions that prepares the USLP Transfer Frame for transport across the space link. The CS-SL is divided into 2 processes: 1) The Frame Interface Processes (FIP) performs the interface functions required to prepare the data for delivery to the Coding/Decoding Process (CDP). This process includes prepending a Frame Start Marker to the provided frame, when management has designated that the frame is not to be aligned to the codeblock or when there is no block code used. 2) The Coding/Decoding Process (CDP) performs the forward error correction processes that are used to optimize the performance of the link and minimize the error rate. The CDP creates the symbol stream that is delivered to the Physical Layer.

The transfer of the USLP transfer frames across different types of space links is the focus of this paper. The Protocol Data Unit (PDU) that is passed in both directions between the Data Link Protocol Sublayer (DLP-SL) and Coding and Synchronization Sublayer (CS-SL) is the transfer frame. The USLP frame structure provides flexibility that can be constrained by the functions utilized within the CS-SL that prepare the transfer frame for transit. For example, the USLP transfer frame contains a length field that enables the frame to be of variable length but CS-SL under certain conditions may constrain the frame to be fixed in length.

This paper describes 5 operational modes available for use by the Data Link Layer to provide data exchange across the USLP space link. These modes are different because different operational requirements apply to vastly different types of space links and thus the communications implementation requirements differ. The environmental issues include the power or energy available, the distance between the end points of the link, the complexity of the equipment available at those end points, the atmospheric conditions and

---

[1] Group Supervisor/Principal Engineer, Project Software and End-to-End Information Group, MS 301-490.
[2] Principal Engineer, Project Software and End-to-End Information Group, MS 301-490.

radiometric frequency selection. The CS-SL utilizes different forward error correcting codes supported by specific operational modes to configure the data for transit. This paper describes all of the operational modes in a series of data models which decompose the functionality between the Data Link Protocol Sublayer and the Coding and Synchronization sublayer.

The operational modes described are:

1. **Uncoded Mode: has been used for short links that contain significant available power to provide an acceptable frame error rate. The frames in this mode can be variable in length and typically use an error detection algorithm (i.e., CRC) to determine if there are errors in the received frame.**
2. **Convolutional Only Mode: is currently the prime forward error correction coding used for the proximity links. The frames in this mode can be variable in length and typically use an error detection algorithm (i.e., CRC) to determine if there are errors in the received frame.**
3. **Variable Length Frame Aligned to Variable Length Codeblock (TC): is used for Direct from Earth links were power levels are high and the simple, least complex code i.e., the BCH code is used. This mode has been in use since the early 1970s. The BCH code is a short code and the decoder is easy to implement.**
4. **Fixed Length Frame Aligned to Fixed Length Codeblock (AOS/TM): was introduced when the concatenated Convolutional and Reed-Solomon Code was formulated to provide significant reduction in link data error rate and the ability to determine if there was an error in the decoded codeblock. The frame is aligned to the codeblock so that there is a one to one relationship of frame errors to codeblock errors without additional error detection coding being added. This mode requires the protocol frames to be the exact size of the message portion of the codeblock.**
5. **Frames Unaligned to Fixed Length Codeblocks (Currently used for very high rates and space to space links): This mode is currently used for missions that have a very high data rate that can be controlled adaptively as the environment changes and as the next generation operating mode for the proximity link. This mode from a coded data stream point of view is exactly like that described in 4. above, except that the frame need not be aligned to the codeblock. There is no requirement on frame length when using this mode. Thus when using USLP it can be used to support links that require short or long frames. There is also no mandatory requirement that frames cannot be separated by idle data reducing the tight data rate connection requirements between the data link protocol sublayer and the coding & synchronization sublayer.**

In conclusion, how these operational modes can be put to use in mission operational scenarios is described for Direct from Earth links (DFE), Direct to Earth links (DTE), and Proximity links.

I. Introduction

THE CCSDS Space Data Link Layer is composed of two sub-layers: the Protocol Sub-layer and the Coding and Synchronization Sub-layer. The Protocol Sub- Layer exclusively utilizes USLP while the Coding and Synchronization Sub-layer can use any of the CCSDS codified coding algorithms. Figure 1 on the next page illustrates the relationship of the CCSDS Space Data Link Layer to the reference model of the Open Systems Interconnection[6].

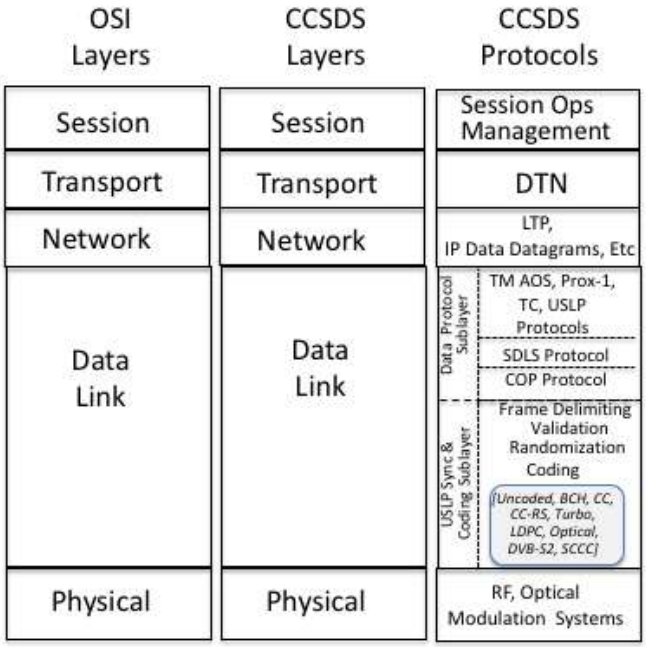


Figure 1. OSI and CCSDS Protocol stack

The USLP protocol has been designed to meet the requirements of space missions for efficient transfer of space application data of various types and characteristics over space-to-ground, ground-to-space, or space-to-space communication links (hereafter called space links). USLP provides a common Protocol Data Unit, i.e., the USLP Transfer Frame that enables all data link applications over all the afore mentioned space links to benefit from software and hardware reuse. USLP also contains a common set of services and service primitives that can be generically used across these links as well. In addition, the protocol provides for the inclusion of a simple go-back-n protocol to provide reliable delivery and the Space Data Link Layer Security (SDLS) Protocol[7] that provides optional security services (Authentication, Encryption, Authenticated Encryption) within the USLP Protocol Sub-layer.

The Synchronization and Channel Coding Sublayer provides the forward error correction coding processes including, as needed, the randomization process and the synchronization process for providing the transfer frame interface with the USLP Protocol Sub-layer. The CCSDS has codified numerous forward error correcting coding algorithms that can be used within the Coding and Synchronization Sub-layer. The concept presented here provides generic methods for interfacing to a link transmitter/receiver that will operate with any CCSDS recommended forward error correction code for the space link.

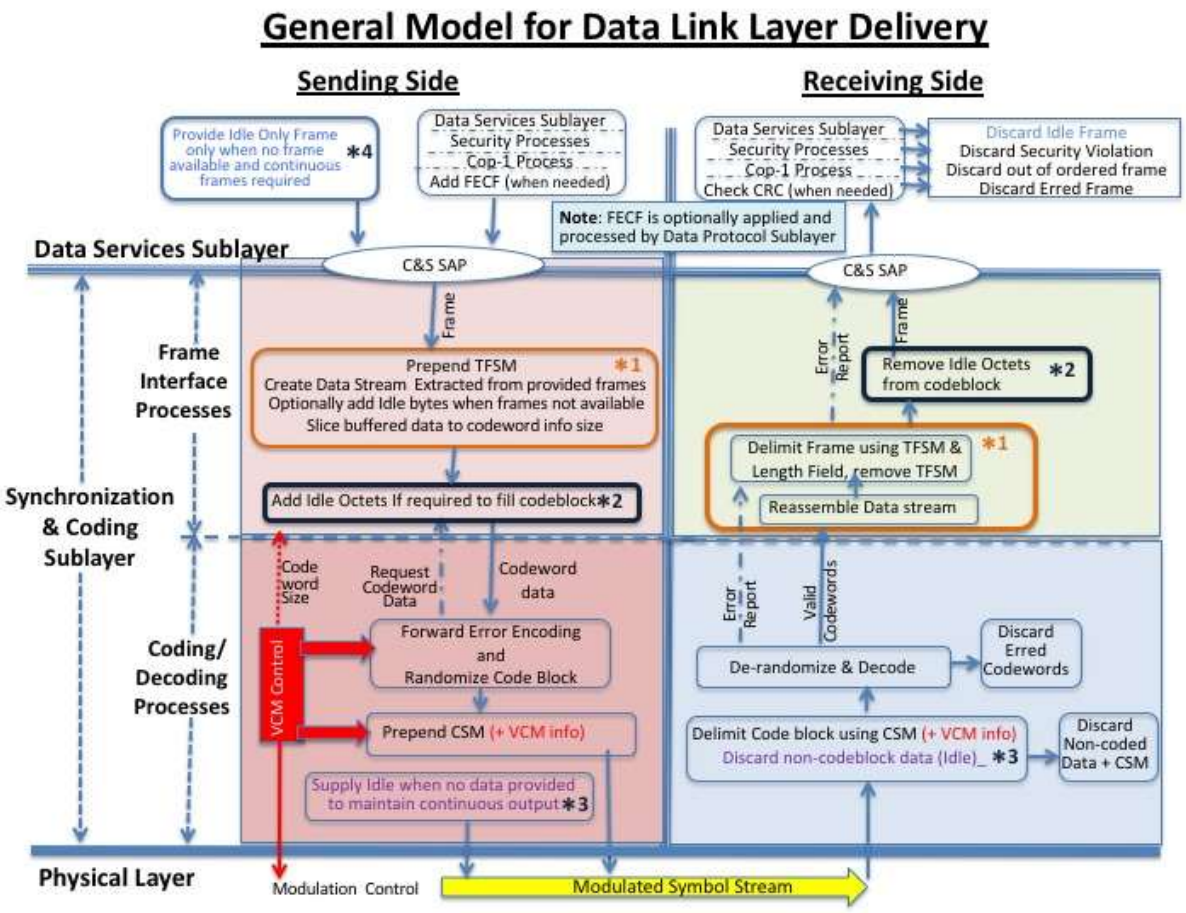


Figure 2. Functions within the CCSDS Data Link Layer Processes

Figure 2 provides a general model of the processes that are performed by the DSL-SL and the CS-SL. Each of the modes require a slightly different combination of the functions included in the diagram that are required to meet the requirements of that specific link. The coding algorithm selected for use on a link is primarily driven by performance requirements and the complexities of power utilization and hardware complexity required for implementation. This paper highlights the operational differences between the types of space links and the configuration within the Frame Interface Process in the CS-SL. Figure 2 above illustrates and identifies the processes within the Link Layer and how the processes are distributed. This is a generalized model containing all the processes that are necessary to provide all the modes; not all of the processes are needed in a specific mode. The specific utilization model for each operational mode is provided in separate figures that contain only those processes necessary for that mode.

II. Types of Data Links

Before describing the various modes and their application to specific link types, let's identify the primary characteristics of each type of link:

1) Direct from Earth (DFE) are used to provide control commands, sequencing, flight software and firmware loads to a spacecraft. Implementation complexity has been a limitation on the performance that can be achieved from spacecraft receivers but on the other hand ground transmitting entities can provide high power uplinks thus enabling reasonable throughput from the limited forward error coding algorithm that

can be supported by the flight receivers. Therefore, this type of link must be able to deliver variable size frames at extremely low error rates.

2) Direct to Earth (DTE) Telemetry Links typically carry engineering and science data. The transmitters in space are constrained by the power available but they are transmitting to ground receivers that can include the maximum of complexity that link designers and forward error correcting algorithm can provide. Engineering data is often transmitted in real time while science data is typically stored onboard and transmitted in non-real time. Each of these data types are typically supported with fixed length frames because of implementation issues. DTE telemetry for very high rate science data for near Earth missions, where the ground can provide rapid feedback to the spacecraft to optimize the link performance, are optimized for high rates using long fixed length frames and very complex forward error correction algorithms that could include Adaptive Modulation and Coding (ACM).
3) Space to Space Links (Proximity) are utilized by Orbiters for data exchange to and from surface bound assets. The orbiters can incorporate complex hardware and firmware that are incompatible with the limitations of landers and rovers. The link dynamic is every changing because the Orbiter is not in an aerostationary orbit thus the path length of the space link changes. The link from the Orbiter typically is limited by the complexity available in the landed element's receiver.

III. Operational Modes

A. Uncoded Mode requires the FIP to prepend a Frame Start Marker (FSM) to the frame. This combination is referred to as a Link Data Unit (LDU). When the data rate required for delivery to the CDP is greater than that produced exclusively from the LDUs, Idle data are introduced into the data stream to provide the required rate. Since the mode is uncoded the CDP delivers the data stream to the Physical Layer with minimal processing.

B. Convolutional Coded Mode requires the FIP to operate exactly like the uncoded mode. The CDP in this mode operates on the provided data stream with a Convolutional encoder whose output is provided to the Physical layer.

C. Variable Length Frame Aligned to Variable Length Codeblock: the FIP accepts the frame and delivers it to the CDP without any processing. The CDP inserts a Codeblock Start Marker (CSM) prior to the codeblock in order to enable the start of the codeblock to be located by the decoder at the receiving side. The CDP slices the frame into parcels that are the size required to fit in the codewords of the codeblock. If the delivered parcel is not the exact size of the message portion within the codewords in the codeblock then idle data are required to fill the last codeword. Since the CDP must deliver a continuous data output stream to the Physical Layer, Idle data must be included in the output stream between codeblocks when the frame delivery rate is inadequate to match the output channel rate.

D. Fixed Length Frame Aligned to Fixed Length Codeblock: the FIP accepts the frame and delivers it to the CDP without any processing. The rules for this mode require the frame to fit within the message portion of the codeblock. The CDP is constrained to encode the received frame within a single codeblock. It is the requirement of the DLP-SL to supply frames at a rate that maintains a continuous output, thus the DLP-SL may need to supply Only Idle Data (OID) frames when frame production does not provide an adequate frame rate. The output symbol stream is a continuum of codeblocks separated only by CSMs.

E. Frames Unaligned to Fixed Length Codeblocks: This is the ubiquitous operational mode that can be used for all types of space links. This mode is presently codified within the CCSDS for use with very high rate telemetry, even those that include adaptive rate and modulation and for space to space links. This mode supports the delivery of variable length frames allowing it to be used for control commanding for space to space links as well as Direct from Earth links. The coding process is turned on at the beginning of the communication pass and remains continuous for the entire pass. Frames are prepended with a frame start marker so that after the received symbol stream is decoded and the error free codewords are concatenated

the FIP would use the marker to delimit the start of the frame and then use the header contents to delimit the frame. Today's forward error correction algorithms are very powerful and have very low error floors that keep frame error rates to a negligible value.

Model for Data Link Layer Delivery Variable Unaligned Mode

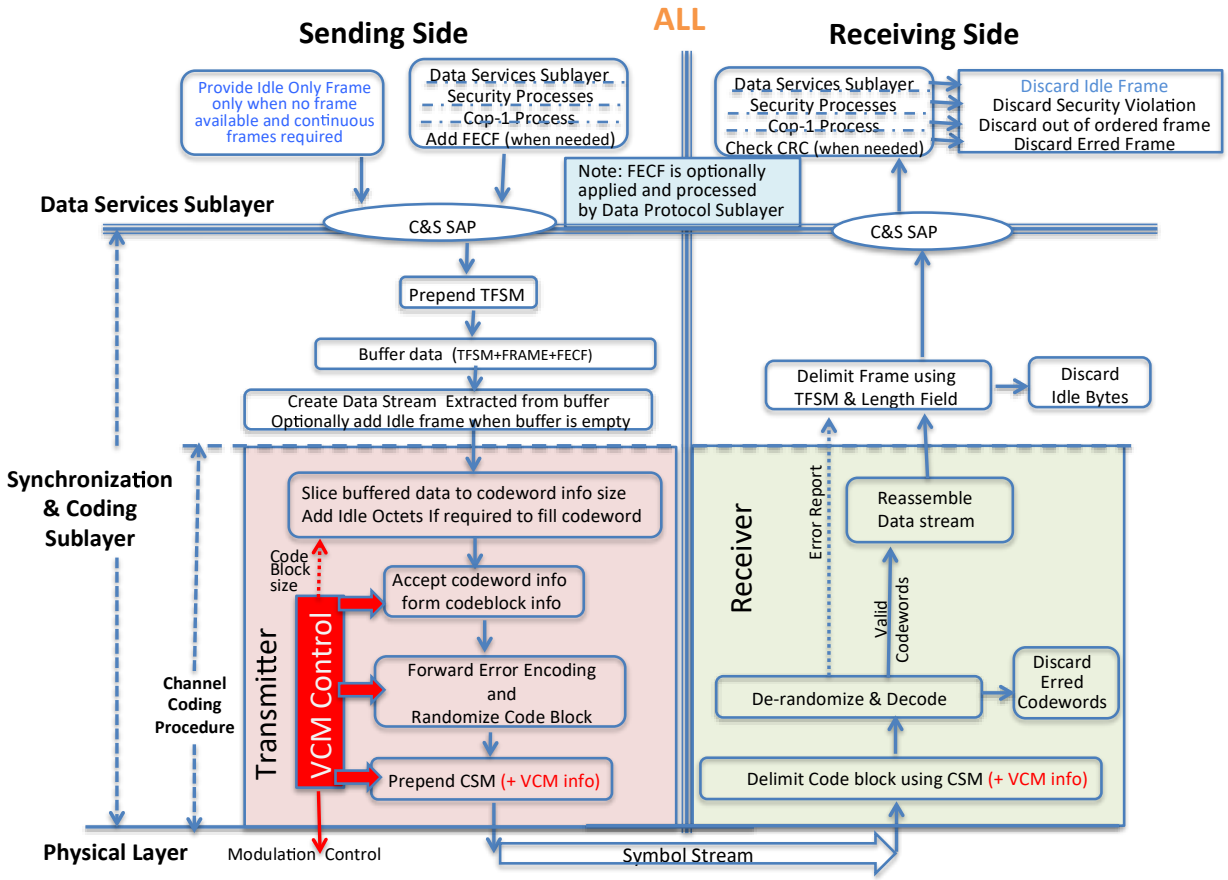


Figure 3. Frames Unaligned to Fixed Length Codeblocks (Data Slicing) Functional Model

A functional description of the Frames Unaligned to Fixed Length Code blocks follows (See Figure 3):

a. On the send side the Frame Interface Process in the Data Link Coding and Synchronization Sub-layer would accept transfer frames from the Protocol Sub-layer. This process would form a continuous data stream by prepending each transfer frame with a Transfer Frame Start Marker (see note 2 below) and insert idle bytes after each transfer frame if no new transfer frame is available when required.

b. In the Code/Decode Process (CDP) the created continuous data stream would be 'sliced' into codeword sized data units for the encoding and randomization processes. The encoding process would use the management designated encoding engine with the designated number of codewords that are aggregated to form a single codeblock. The encoder would then prepend a Codeblock Start Marker (CSM) to the beginning of the codeblock and then pass a continuous bit stream to the transmitter to modulate the generated symbol stream onto the carrier.

c. The receiver would extract the symbol stream from the modulated carrier and deliver that symbol stream to the CDP in the Data Link Coding and Synchronization Sub-layer. The decoding process would search the received symbol stream and initiate a search for the Codeblock Start Marker. Once the CSM is detected the immediately following codeword's worth of symbols would be delivered to decoding process. The CDP includes a derandomizer and a decoder. It would process the codeword symbols, and if no errors were found in a codeword then the information content of

that codeword would be output. If the decoder found any errors in any codeword in the codeblock, the data in that codeword would be discarded, and the decoder would signal an erred codeword, terminating the decoding process. Thereafter the process of locating a CSM would be initiated. The decoding process would be terminated and the search for a CSM would be initiated also when no errors are found after the last codeword in the codeblock.

d. The frame extraction portion of the Data Link Coding and Synchronization Sub-layer would accept the output bit stream from the decoding process and search for the TFSM (see note 2 below). Once the TFSM is located the contents of the transfer frame length field in the USLP transfer frame header will be extracted to provide the data necessary to delimit and extract the transfer frame for delivery to the USLP Protocol Sub-layer. The TFSM search process would restart after each transfer frame is delimited.

Note 1: The data processing steps within the Data Link Coding and Synchronization Sub-layer are the same even where only the Convolutional Code or where no code is used on the link.

Note 2: Because the Continuous Encoding Method outputs continuously encoded codeblocks, each separated by a Codeblock Start Marker, the encoding and decoding processes resemble the current CCSDS TM Synchronization and Coding specification[2]. This specification dictates that the FIP provides a frame to the encoding process synchronous to the start of a new code block. In this case a TFSM would not be prepended to the frame because the frame would be synchronized to the codeblock. At the receive side the FIP will receive a codeblock whose information content is equivalent to a transfer frame. The frame is then output to the USLP Protocol sublayer. This requires all transfer frames to be fixed length and be the same size of the message portion of the codeblock.

Figure 4 provides example use cases for the USLP operational modes:

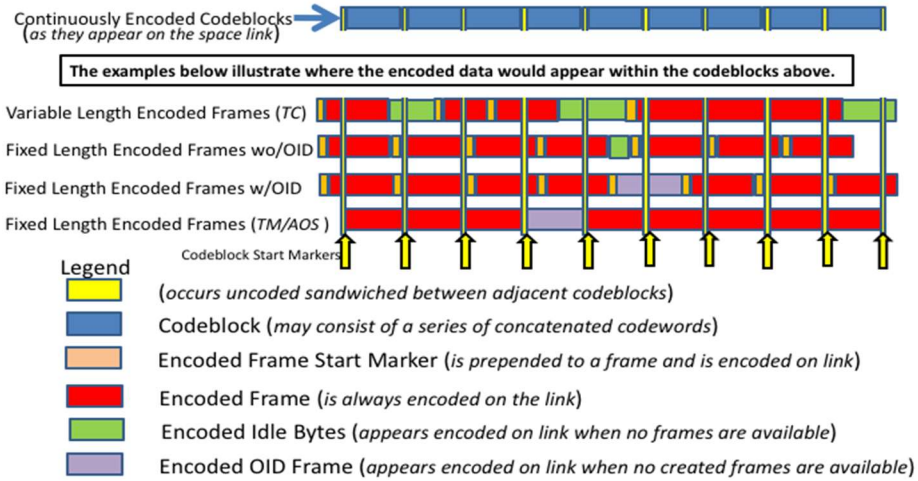


Figure 4. Example Use Cases for the USLP Operational Modes

The Variable Length Frame aligned to Variable Length Codeblock mode described in Figure 5 below could also be used to provide universal frame service. This method accepts a frame and forms a codeblock to contain it. It will form the codeblock from a series of as many codewords as required to contain a transfer frame. This method does not require the addition of a Frame Start Marker because the frame start is aligned with the start of the codeblock. The codeblock would be preceded by a Codeblock Start Marker that identifies both the start of the codeblock and the encoded frame. When there is no constraint on size of the frame, the last codeword in the codeblock may require fill bytes to complete the codeword when the transfer frame is shorter than the data area in the created codeblock.

When the transfer frames arrive at a rate that is insufficient to allow the created codeblocks with their codeblock start markers to be joined together then fill bytes are placed into the output symbol stream. When the physical layer's managed parameters require fixed length transfer frames that are aligned to the codeblock (as in TM[8] and AOS[9]) then the Protocol Sub-layer is required to provide transfer frames consistent with the information rate of the encoding process. This method is compatible with both fixed and variable length frames and thus has the characteristics necessary to support DFE and space to space links in addition to the DTE link.

The current TC Space Data Link Protocol[10] requires that the codeblock be terminated by an erred codeword i.e., the tail sequence. This method was chosen because it decoupled the decoding process from the transfer frame format and used the decoding process to determine the end of the codeblock. This method is not very costly in overhead when the codeword is short but becomes inefficient for larger code words (even as small as 1024 bits). When using the 1024 bit or larger codes, alternative methods could be used to determine the number of codewords in the codeblock.

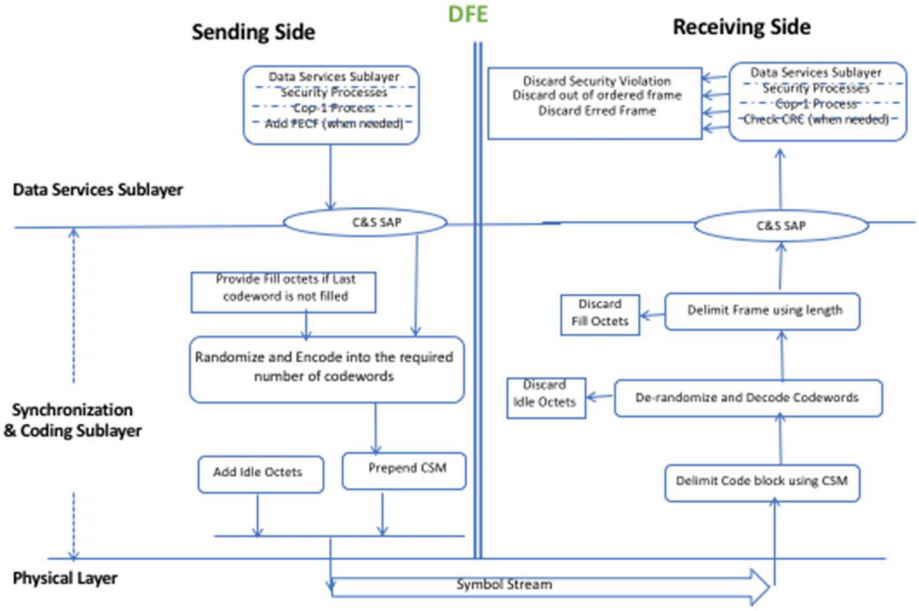


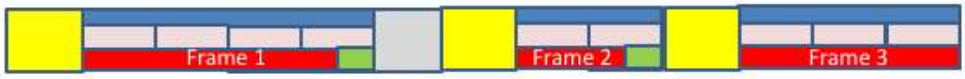
Figure 5. Variable Length Frame aligned to Variable Length Codeblock Mode Functional Model

There are many ways to delimit the codeblock. Below are two alternative codeblock delimiting approaches:

1. One byte of the codeword could be reserved as a signaling byte that points to the end of the last codeword of the codeblock, allowing the decoder to find the end of the codeblock independent of the data link layer protocol.
2. The USLP transfer frame header would always be in the same location for all links and it would always be in the first codeword of the codeblock. Therefore after decoding the first codeword, the transfer frame length would be available for delimiting the transfer frame and therefore how many code words are contained within the codeblock.

Since there are a number of CCSDS recommended codes, the question is what codes are appropriate for each of these operational modes and for which type of link are they appropriate. First, all of the CCSDS codes except for the Convolutional code (k=7, r=1/2) are block codes. Typically a single coding algorithm is selected for use on a link because of its performance and the complexity of construction and operation of the hardware. For links that terminate in space, decoding complexity is an important feature along with receiver capability. When short code words are chosen performance overhead (due to frequency of Codeblock Start Markers) and codeblock/transfer frame rate become important issues. For this reason, the codeblock for the Variable Codeblock Mode is defined as consisting of a codeblock synchronization marker followed by 1 or more codewords. For the Continuously Encoded method the codeblock is fixed but the number of codewords per codeblock could be managed and selectable for different data rates.

Variable Length Codeblocks encapsulate the encoded frame and consist of a series of codewords. The encoded frame starts in the first byte of the codeblock. The last codeword may contain fill bytes when the encoded frame does not fill the codeblock. When no frames are available the link will contain uncoded idle.



Notes:

1. **Since no frame was available immediately after frame 1, uncoded idle data were inserted**
2. **Since a frame was available immediately after frame 2, the next codeblock started immediately**
3. **Since sufficient number of frame bytes were provided in frame 3 to fill out the codeblock, no fill bytes needed to be inserted.**
4. **An AOS style mode is achieved if all frames are the same size and they totally fill the information space within the codeblock.**

Legend

- Codeblock Synchronization Marker (*occurs uncoded prepended to each codeblock*)
- Codeblock (*variable length containing a series of concatenated codewords*)
- Codeword (*a series of codewords may be required to encapsulate an encoded frame*)
- Encoded Frame (*is always encoded on the link*)
- Encoded Idle Bytes (*appears encoded when required to fill last codeword*)
- Uncoded Idle Bytes (*appear uncoded on link when no frames are available*)

Figure 6. Example Use Cases for 'Variable Length Frame aligned to Variable Length Codeblock' Mode

IV. Operational Issues Associated with these Modes

A. Operational 'Frames Unaligned to Fixed Length Codeblocks' Mode

In this mode, the transmitting side of the link creates an output symbol stream consisting of a continuum of a CSM followed by a codeblock. This method provides a level of independence between the Data Link Layer and the transmitter. The transmitter accepts a data stream and encodes and randomizes the data stream forming a symbol stream that is modulated on to a carrier for transmission to the link terminus. The data content of the output stream could be variable length transfer frames that occur randomly or as a continuum of transfer frames. This method could be slightly constrained to require the transfer frames to be fixed sized and or to require them to be inserted continuously without gap. There is a slight decrease in link performance because the data may be spread across an additional codeblock. Another issue is that an emergency command would prefer to be aligned with the codeblock to require the shorted view period for acceptance. This situation could be handled with slightly different managed parameters because this type of emergency command is only relevant at very low data rates.

B. 'Variable Length Frame aligned to Variable Length Codeblock' Mode

In this method the transmitting side of the link creates a codeblock from a series of codewords that fully contain the transfer frame. The codewords on the link are always the same size and use the same encoding algorithm. The transfer frame will always start in the first byte of the first codeword in the codeblock. Each codeblock will be preceded by a CSM. When a transfer frame does not fit exactly into the codeblock, fill bytes are appended to the transfer frame to fill the message portion of the last codeword in the codeblock. When there is no codeblock to send

on the link then uncoded Idle bytes are inserted in order to maintain a continuous output symbol stream. Various management controls can be placed on the Data Link Layer that supplies the transfer frames that can change the operational look of this method. What remains invariant is that a transfer frame is always contained within a single codeblock. This fact significantly reduces the need for a Frame Error Control Field containing a CRC.

a) When there are no constraints on the Data Link Layer then variable length transfer frames of any size could be created and arbitrarily released transfer frames would have the link include Idle at times between encoded transfer frames. This resembles the link dynamics for Telecommand.
b) Imposing a constraint that requires all transfer frames to be the same size as the message portion within a codeblock eliminates the inclusion of fill. This allows for there to be different size transfer frames but they must be sized to contain an integer number of codewords.
c) Imposing a constraint that all transfer frames be of a specific size equal to the message portion within a codeblock and to require that there be no idle between transfer frames, creates a link that resembles both TM and AOS.

Note that this link operational mode could at times operate as described in any or all of the sub-modes described in a, b and c above. Where for example a) would be used for operational commands including Proximity-1 Hailing commands; Where b) could be used for sending sequences or file loads across DFE or Space to Space links; and where c) mimics high rate data transfer.

V. Conclusions

Why adopt the Continuous Encoding Method over the Variable Codeblock Encoding Method? And why adopt a coding and sync methodology that is transparent to the protocol sublayer?

Pros:

1. It functionally supports the use of every CCSDS codified coding algorithm (including uncoded).
2. It accepts a bit stream and transfers that very same bit stream with minimum error across a link.
3. It supports all CCSDS Protocol formats.
4. It supports native UDP placed directly into the bit stream.
5. It is compatible with every space link environment (DFE, DTE and Space to Space).
6. It allows for the design, development and testing of a space link transponder/transceiver that is agnostic about the protocol sublayer.
7. It allows for the design, development and testing of the protocol sublayer that is agnostic about the space link transponder/transceiver.

Cons:

1. The received frame most likely will be subject to a slightly higher error rate probability than if the frame neatly fit within a single codeblock.
2. The frame is not directly validated by a single codeblock but requires that all the codeblocks that contain portions of the frame are validated. (validation can be accomplished by a FECF).
3. Current time correlation method accuracy is diminished due to the non-alignment of the frame and the codeblock. By time tagging the first byte of a codeblock for time correlation, time correlation can be performed in the coding sublayer independent of protocol layer except for the transfer of the time measurements as they relate to the codeblocks.
4. Emergency commands for short window access for delivery caused by a malfunction e.g., uncontrolled spacecraft tumbling may require special handling.

   Example: Standard command processing would have the station create a continuously encoded data stream where the commands are not aligned to the codeblocks. This process could therefore require the decoding of an extra codeblock in order to receive and then extract the command. Under these type of circumstances, a command that was aligned to the codeblock and constrained to fit within a minimum of codeblocks would require the minimum time to be acquired. Thus a different approach for this very special unique need would require an approach that could be tailored for these conditions that are not necessarily practical for normal commanding. In this case the command would be formulated and encoded in the POCC and radiated by the station as uncoded. The command would be built with an acquisition portion followed by the CSM followed by the command already encoded in the link's prescribed code and if required followed by idle bits.

Appendix A

Acronym List

ACM	Adaptive Coding and Modulation
AOS	Advanced Orbiting Systems (Protocol)
CC-RS	(Concatenated) Convolutional Reed-Solomon
CCSDS	Consultative Committee for Space Data Systems
CDP	Coding/Decoding Process
COP	Communications Operation Procedure
CRC	Cyclic Redundancy Check
CS-SL	Coding & Synchronization Sublayer
CSM	Codeblock Start Marker
DFE	Direct-From-Earth
DLP-SL	Data Link Protocol - Sublayer
DTE	Direct-To-Earth
DTN	Delay Tolerant Networking
DVB-S2	Digital Video Broadcasting - Satellite - Second Generation
FECF	Frame Error Control Field
FIP	Frame Interface Process
IP	Internet Protocol
ISO	International Organization for Standardization
LDPC	Low Density Parity Check
LTP	Licklider Transmission Protocol
NASA	National Aeronautics and Space Administration
OSI	Open Systems Interconnection
PDU	Protocol Data Unit
POCC	Project Operational Control Center
Prox-1	Proximity-1 (Protocol)
RF	Radio Frequency

SAP	Service Access Point
SCCC	Serially Concatenated Convolutional Codes
SDLS	Space Data Link Security (Protocol)
TC	Telecommand (Protocol)
TFSM	Transfer Frame Synchronization Marker
TM	Telemetry (Protocol)
USLP	Unified Space Link Protocol
VCM	Variable Coded Modulation

VI. Acknowledgments

The work described in this paper was performed at the Jet Propulsion Laboratory, California Institute of Technology under a contract with the National Aeronautics and Space Administration. © 2018 California Institute of Technology. Government sponsorship acknowledged.

VII. References

[1] CCSDS Unified Space Data Link Protocol, CCSDS 732.1-R-3. Red Book. Issue 3. August 2017. Consultative Committee for Space Data Systems, www.ccsds.org

[2]CCSDS TM Synchronization and Channel Coding, CCSDS 131.0-B-2. Blue Book. Issue 2. August 2011. Consultative Committee for Space Data Systems, www.ccsds.org

[3]CCSDS Flexible Advanced Coding and Modulation Scheme for High Rate Telemetry Applications, CCSDS 131.2-B-1. Blue Book. Issue 1. March 2012. Consultative Committee for Space Data Systems, www.ccsds.org

[4]CCSDS CCSDS Space Link Protocols over ETSI DVB-S2 Standard, CCSDS 131.3-B-1. Blue Book. Issue 1. March 2013. Consultative Committee for Space Data Systems, www.ccsds.org

[5] CCSDS TC Synchronization and Channel Coding, CCSDS 231.0-B-3. Blue Book. Issue 3. September 2017. Consultative Committee for Space Data Systems, www.ccsds.org

[6]*Information Technology—Open Systems Interconnection—Basic Reference Model: The Basic Model*. 2nd ed. International Standard, ISO/IEC 7498-1:1994. Geneva: ISO, 1994.

[7] CCSDS Space Data Link Security Protocol, CCSDS 355.0-B-1. Blue Book. Issue 1. September 2015. Consultative Committee for Space Data Systems, www.ccsds.org

[8] CCSDS Telemetry Space Data Link Protocol, CCSDS 132.0-B-2. Blue Book. Issue 2. September 2015. Consultative Committee for Space Data Systems, www.ccsds.org

[9] CCSDS AOS Space Data Link Protocol, CCSDS 732.0-B-3. Blue Book. Issue 3. September 2015. Consultative Committee for Space Data Systems, www.ccsds.org

[10] CCSDS Telecommand Space Data Link Protocol, CCSDS 232.0-B-3. Blue Book. Issue 3. September 2015. Consultative Committee for Space Data Systems, www.ccsds.org