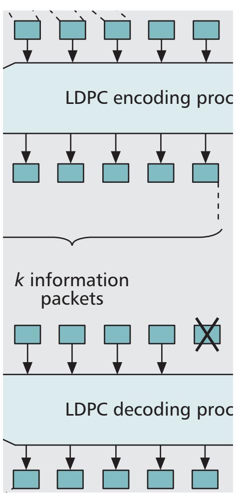
RELIABLE DATA DELIVERY OVER DEEP SPACE NETWORKS: BENEFITS OF LONG ERASURE CODES OVER ARQ STRATEGIES

TOMASO DE COLA, GERMAN AEROSPACE CENTER (DLR)
MARIO MARCHESE, UNIVERSITY OF GENOA



Achieving reliable communications in deep space environments poses formidable networking challenges. Two possible approaches can be considered: automatic repeat request schemes and packet layer coding algorithms applied with long erasure codes.

ABSTRACT

Achieving reliable communications in deep space environments poses formidable networking challenges because of the extreme physical medium peculiarities. In this view, two possible approaches can be considered to carry out reliable data transfers over deep space channels: automatic repeat request schemes and packet layer coding algorithms applied with long erasure codes. In this respect, this article surveys the mechanisms currently available from the Consultative Committee for Space Data Systems protocol stack, by reserving special attention, on one hand, to the ARQ schemes currently implemented at the application layer and, on the other hand, to the potential offered by erasure coding schemes. A comparative analysis gives some insights about the performance improvements the packet layer coding methodology can bring. In particular, the results show that the use of erasure coding is able to attain more satisfactory performance results than ARQ-based schemes in terms of reliability, data transfer delay, resource network utilization, and power consumption.

INTRODUCTION

Recent advances in telecommunication technology, such as more effective modulation and channel coding schemes, reduced size and costs of devices, large storage availability, and fast processing capabilities, paved the way for the extension of the Internet's borders toward outer space regions [1, 2]. The National Aeronautics and Space Administration (NASA) promoted space exploration programs aimed at deploying observation and experimentation centers on the moon and Mars. Consequently, the design and implementation of a complex telecommunication infrastructure suitable to offer connectivity to different nodes located very far from each other will be necessary for the future. In principle, this can be done by simply extending the scope and functionalities of communication solutions commonly adopted for terrestrial environments, where the TCP/IP protocol suite is the de facto standard. However, the challenging peculiarities of the interplanetary environment, such as very large propagation delays, outage events, bandwidth asymmetry, and high link error rates make TCP-based protocols hardly applicable [3]. Sliding window mechanisms and recovery procedures relying on either timeout expiration or triple duplicate acknowledgment reception, as used by TCP, are not effective in this context. The employment of a sliding window approach is severely impaired by very large bandwidth-delay products, which in deep space can be up to tens of megabytes. In addition, very large latencies experienced by deep space links, which can be as long as many seconds, also imply long recovery phases, thus resulting in degraded performance. Finally, yet important, TCP considers each packet loss as due to congestion, and this is the motivation for the input rate reduction. On the contrary, losses are mainly due to channel errors in deep space: the rate reduction penalizes the channel throughput and consequently increases the delay of data transfers.

To contrast the aforementioned TCP shortcomings, an alternative protocol architecture has been devised within the Consultative Committee for Space Data Systems (CCSDS): it consists of a set of protocol recommendations the design of which is specifically tailored to the requirements of space missions and the peculiarities of deep space environments [4]. A fully operational protocol stack, designed from the application layer down to the physical layer, is specified as an alternative to the TCP/IP protocol suite. In particular, special attention has been paid to the protocol specifications of lower layers (i.e., physical and data link) and the CCSDS File Delivery Protocol (CFDP), which may be positioned at the application and transport layers. Both the lower-layer protocols and CFDP implement advanced recovery mechanisms: near-Shannon limit channel coding the former, and powerful automatic repeat request (ARQ) strategies the latter.

Due to physical peculiarities of transmission channels and system design limitations introduced by the available power budget, size, and cost of devices, deep space scenarios pose challenges from both the communication and networking point of view.

The scientific community is active in this field as shown by [5, 6, references therein]: the challenges of the deep space environment has fostered research studies toward specific protocol extensions, aimed at improving the speed and reliability of data transfers. From this viewpoint, erasure coding schemes applied at packet level seem promising to guarantee higher robustness against consistent link errors and information loss [7]. This solution is documented in the literature as packet-layer coding and has also been extensively considered within CCSDS as a focus of the Long Erasure Codes Birds of a Feather (LEC BOF) activity, which is part of the Space Link Service (SLS) area [8]. The principle of this approach is to apply encoding/decoding operations on a packet basis, in order to generate a number of redundancy packets sufficient to contrast the physical environment impairments, recognized as information erasures at higher layers. From this view, the advantages offered by such an approach are attractive for the performance benefits they could bring with respect to ARQ-based solutions, whose retransmission latency is expected to degrade the overall system performance.

The above considerations on ARQ strategies and erasure coding schemes are taken as the starting point in this work. The aim of this article is to review advantages and drawbacks of the two aforementioned approaches over deep space scenarios. In addition, the case of hybrid-ARQ schemes, relying on the combination of ARQ and coding methodologies, is considered as well. On the basis of this investigation, system design guidelines are finally drawn up, taking into account implementation and the physical constraints introduced by the deep space environment.

The remainder of this article is organized as follows. The next section briefly reviews the main characteristics of the interplanetary environment and discusses the research issues that are unexplored or only partially addressed in the literature. We then describe the basic concepts of packet-layer coding, and point out the implementation challenges arising at the different layers in which the erasure code schemes can be applied. We then review the main aspects of the CCSDS protocol stack by paying particular attention to CFDP specifics and proposing possible improvements to it. The performance analysis of relevant case studies is then reported, and our main conclusions are drawn in the final section.

Deep Space Scenarios: Challenges and Open Issues

Due to physical peculiarities of transmission channels and system design limitations introduced by the available power budget, size, and cost of devices, deep space scenarios pose challenges from both the communication and networking points of view [9].

Concerning the former, particular attention must be paid to the large distances that usually separate the telecommunication nodes and imply significant signal propagation delays. In principle, by overlooking space missions currently ongoing or scheduled for the future, it is possible to classify the application scenarios as either *near-Earth* or *deep-space* [8], depending on the position of spacecraft with respect to Earth stations. The first case considers data communications performed between Earth control centers and other nodes located at altitude below $2 \cdot 10^6$ km: signals experience propagation delays lower than 6.6 s. The second case considers communications established between Earth and other nodes (e.g., spacecrafts and landers) located at distances farther than $2 \cdot 10^6$ km. Recent space missions related to Mars (the Mars Global Surveyor exploration program) and Saturn (the Cassini-Huygens exploration program) belong to the deep-space scenario. Propagation delays range from tens of minutes up to hours. From this preliminary analysis, it is straightforward to see that the signal strength is severely degraded by the free space loss, which can be as high as 290 dB for Saturn–Earth communications (when Saturn is in the closest position with respect to Earth), achieved in the X frequency band (7–12.5 GHz). In addition, other environment impairments such as solar wind, flares, and space thermal noise also sum up to further reduce the received signal-to-noise ratio. Finally, outage events are also likely to happen due to synchronization loss between receiving and transmitting stations and bad weather conditions, such as rain in the Ka frequency band (26.5–40 GHz) and wind. All the aforementioned aspects give rise to frequent information losses that can be quantified with raw channel bit error rates ranging from 10^{-1} to 10^{-3}.

Other important factors influencing the overall system performance are scarcity and asymmetric bandwidth availability. Usually, the uplink (connecting Earth to spacecraft) is used to transport command messages and offers an available bit rate in the order of 10 kb/s. The downlink handles image and measurement data, and offers a bit rate up to a few megabits per second.

These characteristics (large latency, highly asymmetric bandwidth, error rates, and outage events) make TCP-based protocols hardly applicable, since their performance is strictly influenced by feedback delays and retransmission procedures that are expected to take place in response to congestion events. On the other hand, the implementation of more sophisticated transmission protocols has to be carefully traded off with the current device hardware constraints in terms of limited processing and storage capability, power budget, and size. This has immediate consequences on the telecommunication system design, mostly concerning the downlink. The availability of higher bit rates envisaged for future space missions calls for more advanced coding and modulation techniques (longer codes and higher-order modulations), the feasibility of which has to be checked against the requirement of lower complexity demanded for in situ devices. Another option is represented by the possibility to increase the power supply availability to ensure satisfactory signal-to-noise ratio figures along with larger data rates, but the limited power budget usually available on spacecraft orbiting around remote planets (e.g., Saturn), and landers and rovers acting on the planet surface make this option unpractical.

Other design issues arise from the networking point of view. NASA programs envision the deployment of interplanetary Internet [5] based on a meshed topology in which the space nodes, serving as routers, are responsible for data storage and forwarding. From this perspective, attention must be given to the nodes' limited storing capabilities, which could result in congestion events, rare, for now, in this environment. In view of the future space Internet, another important aspect is represented by quality of service management: although still missing in current space network deployments, it deserves attention for the next-generation networks that will entail both terrestrial and interplanetary links, and switching nodes. Traditionally, service level requirements are matched in terrestrial Internet through differentiated services (DiffServ) or integrated services (IntServ) methodologies applied at the network layer. However, the effective and dynamic application of these schemes requires signaling mechanisms that are not feasible in deep space environments because of the large propagation delays and high error rates, which may severely impair the QoS management strategy. A more viable alternative consists of defining static policies able to differentiate between low-priority data flows, such as image and measurement retrieval, and high-priority ones such as telemetry messages and emergency notifications, and simple mechanisms to match the QoS requirements in terms of loss and delay.

As a partial response to the design issues raised above, this article explores the benefits brought by the packet layer coding approach, which seems the key to guaranteeing satisfactory information loss rates and data delivery delay. Also, the impact on limited network resources and power budget are investigated by introducing proper performance indicators.

Packet-Layer Coding: Main Concepts

Overview

Shadowing and fading events result in oscillations of the signal-to-noise ratio measured at the destination, thus giving rise to bit errors within frames. An effective way to combat link errors is the employment of channel coding techniques that are effective in either error correction or detection. However, for long fading events, channel coding alone is not sufficient to protect transmitted data against errored bits. As a result, the corrupted frame will be discarded at the data link layer upon failed computation of the checksum or cyclic redundancy check (CRC). This results in *packet erasures* occurring in bursts, which should be recovered by the upper layers. The idea is to tackle them through appropriate erasure coding schemes, which complement the error protection already provided by the channel coding schemes implemented at the physical layer. Erasure schemes basically follow the principles of forward error correction (FEC) schemes: k source information units (hereafter generally referred to as packets) are encoded into n units, of which $n - k$ are redundancy packets. An important parameter is the code rate, defined as the ratio between information and total coded units (k/n): its proper tuning is fundamental to achieve satisfactory performance [9]. Decoding operations at the destination are accomplished as soon as a sufficient number of packets is received. In case of maximum distance separable (MDS) codes, such as Reed Solomon (RS) [10], just k out of n packets are needed for successful decoding operations. Unfortunately, RS codes are nonlinear block codes [10] whereby the high software implementation complexity imposes the use of short codewords, resulting in encoding/decoding algorithms that follow a time quadratic law. As an alternative, schemes relying on low density parity check (LDPC) codes offer linear encoding/decoding complexity as they implement simple XOR operations. Unfortunately, unlike RS codes, they are not MDS codes, and decoding operations complete as soon as at least $k(1 + \varepsilon)$ packets are received correctly. In more detail, ε gives an indication of the code inefficiency; it decreases to 0 as the number k of coded packets increases [8]. Finally, coding solutions applying the concept of the *digital fountain*, such as LT, Tornado, and Raptor codes [11], also deserve great attention and were initially considered within the CCSDS standardization process. Despite the advantages they may offer in terms of *rateless* coding (raptor codes), they require the availability of a return channel for signaling the completion of the decoding procedure to the sender side, which otherwise would continuously transmit new redundancy symbols, wasting power and deep space link bandwidth. Hence, LDPC codes are currently regarded as more appropriate to meet the performance requirements of deep space communications. Accordingly, the focus of this article is only on LDPC-based codes achieving near-Shannon limits [12], as presently done in the CCSDS standardization process.

Packet Layer Coding through Layers

Packet layer coding approach can be applied at different layers of the protocol stack, from the application down to the data link layer, where actually a packet unit may be defined.

Application/Transport Layer Coding — It is applied on end-to-end basis: the coding strategy can be configured according to the content carried by data packets and the error protection they may need. This approach allows the underlying protocol stack to remain unmodified, offering several advantages in terms of flexibility and modularity of the whole deep space communication system design.

Network Layer Coding — It works on a point-to-point basis, thus allowing efficient contrast of packet erasures experienced with different loss patterns in a multihop environment. The main drawback is represented by the necessity to modify the different network layer protocol specifications that may be present on the network segments, depending on the space missions. This may become too burdensome from the implementation point of view.

Data Link Layer Coding — As outlined also for the network layer, it works on a point-to-point basis. The coding strategy can be tuned according to

In case of long fading events, channel coding alone is not sufficient to protect transmitted data against errored bits. As a result, the corrupted frame will be discarded at the data link layer upon failed computation of the checksum or CRC.

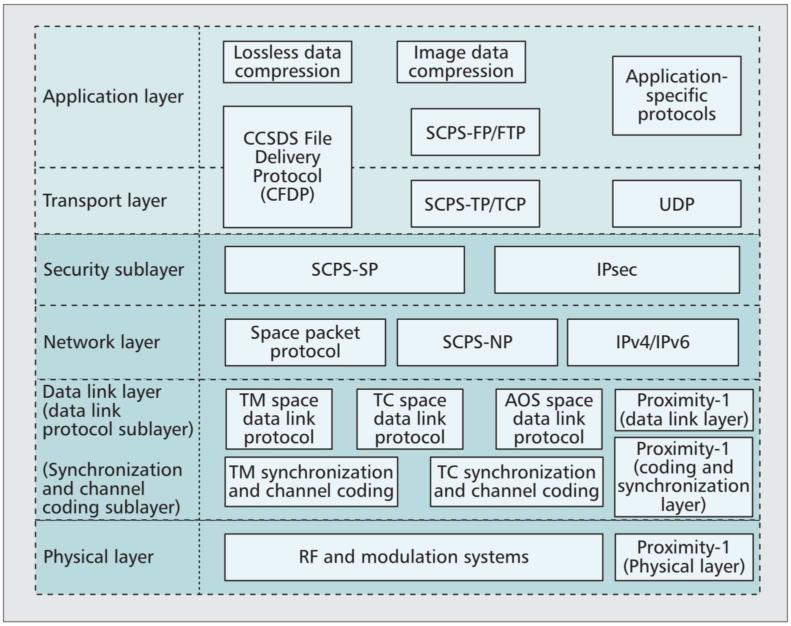


Figure 1. *CCSDS protocol architecture: higher (light blue) and lower layer protocols (dark blue) [4].*

the channel quality, provided that information about the signal degradation is available from the physical layer. Also in this case, modifications of different data link layer protocol specifications would be required, with an impact on the flexibility of the overall system design.

The Reference Protocol Architecture

CCSDS Protocol Stack: An Overview

CCSDS activity is primarily focused on the definition and implementation of a protocol architecture alternative to the existing ones (e.g., TCP/ IP suite) to support effective data transfer over long delay and lossy networks, as in the case of interplanetary networks. A full protocol stack, including all the protocols from the application to the physical layer, has been recommended, designed, and deployed in spacecraft, satellites, and Earth stations. The protocol stack composition is shown in Fig. 1, where the separation between higher- and lower-layer protocols is highlighted. A short summary of each layer is reported in the following.

Physical layer: CCSDS recommendations on RF and modulation systems focus on the most suitable transmission schemes to be adopted in space missions, where either long-haul links (long-range and bidirectional), established to allow communication between spacecraft and satellites very far from each other, or proximity links (short-range and bidirectional) are employed.

Data link layer: CCSDS has developed four protocols: Telemetry (TM), Telecommand (TC), Advanced Orbiting Systems (AOS), and Proximity-1 space link protocol-data link [4]. Their basic function is to forward transfer frames of fixed or variable length to the physical layer, by taking care of synchronization and channel coding functions, along with encapsulation and framing operations.

Network layer: Two protocols have been proposed as alternatives to IP: the Space Packet Protocol and the Space Communication Protocol Specification-Network Protocol (SCPS-NP). Both take care of addressing and routing operations.

Transport layer: CCSDS has developed the Space Communications Protocol Specification-Transport Protocol (SCPS-TP) to provide end-to-end reliable communication, as an alternative to TCP.

Application layer: CFDP is designed to get reliable file transfers. It follows an FTP-like paradigm. Its implementation spans over application and transport layers. In addition to CFDP, other specific applications such as asynchronous message service are defined within the CCSDS protocol stack.

In this article particular attention is given to CFDP, whose main features are pointed out in the following. For a complete description of the other aforementioned protocols, it is possible to refer to [4, references therein].

CFDP

The CCSDS File Delivery Protocol (CFDP) [13] aims at transferring files from one filestore to another, located in spacecraft and space stations.

The file to be transmitted is encoded into a file delivery unit (FDU), composed of the file itself and metadata necessary for data management. The CFDP entity splits the FDU into CFDP protocol data units (PDUs) of variable length. CFDP PDUs are structured into payload, containing up to 65,536 bytes, and header, containing CFDP source and destination identifiers, transfer file sequence number, as well as other fields suited to allow the reconstruction of the FDU at the destination. Data transmission is performed by CFDP entities according to two operative modes, unacknowledged and acknowledged. The former implements no mechanisms to ensure complete data delivery; communication reliability, where required, should be ensured by proper mechanisms implemented within the underlying layers. The latter provides reliable delivery of data by means of ARQ strategies, relying on negative acknowledgment (NAK). The detection of missing CFDP PDUs is performed by the receiver, which notifies the loss of data to the sender by issuing NAK blocks according to four different algorithms: Immediate, Deferred, Asynchronous, and Prompted. In the first case a NAK issuance is performed as soon as the loss of CFDP PDUs is detected. Deferred mode allows postponing the issuance of NAKs to the end of the file transfer. As far as prompted and asynchronous modes are concerned, the detection of missing blocks is triggered by external events, such as explicit (asynchronous mode) or periodical (prompted mode) requests by the sender.

Proposed Protocol Solutions

The advantages offered by erasure codes, pointed out in the previous section, support the idea of implementing a packet-layer coding strategy within the CCSDS protocol stack, as proposed by the LEC BOF working group [8]. The frame error detection and correction functions performed by the CCSDS data link layer, depending on the specific implemented protocol, can give rise to frame discarding events. The use of erasure codes could be beneficial to recover from frame losses that otherwise would trigger long retransmission periods performed by the CFDP entity, hence penalizing the overall system performance.

Four possible coding strategies can be taken in consideration:
1. Pure FEC
2. Type-I Hybrid ARQ
3. Type-II Hybrid ARQ
4. Weather Genie [14]

The first one consists of the generation and transmission of information and redundancy units over the forward link. Solutions 2 and 3 combine advantages of FEC and ARQ strategies: Type-I Hybrid ARQ allows retransmitting the information symbols that could not be recovered at the destination through erasure decoding; Type-II Hybrid ARQ consists of sending additional redundancy symbols upon notification of failed erasure decoding at the receiver side. Weather Genie exploits the availability of a return channel to acquire information about the deep channel state and adapt the coding strategy accordingly. It is immediately apparent that some challenges can arise particularly for solutions 3 and 4, by virtue of the need for a return channel and for a protocol specifically designed to use it. Hybrid ARQ-II demands a dedicated protocol implemented at the receiver side to request additional redundancy symbols. Likewise, Weather Genie requires a dedicated protocol able to estimate the channel state and transmit it to the sender side. On the other hand, Type-I Hybrid ARQ, although demanding for the return link like solutions 3 and 4, can be implemented within layer protocols that already implement retransmission procedures (e.g., CFDP) to recover from information losses. According to these observations and taking under consideration the limited implementation complexity allowed on space nodes, solutions 3 and 4, while attractive, cannot presently be adopted in deep space environments and are not investigated hereafter. On the other hand, solutions 1 and 2 easily meet the technological requirements of space nodes and are thoroughly investigated in the following.

Although several alternatives could be considered, CFDP is taken as the reference in this article for the integration of both pure FEC and Type-I Hybrid ARQ schemes.

The rationale under this choice is twofold. First, the implementation of packet-layer coding schemes within CFDP allows leaving the underlying protocol stack untouched, which defines different protocol specifications at the same layer (e.g., the network layer may implement IP, CCSDS SPP, or SCPS-NP). Second, this approach allows selection of the coding parameters (e.g., parity check matrix and coding coderate), taking control, at the same time, of the CFDP parameters (e.g., CFDP PDU size) appropriate to meet the QoS requirements demanded by the files to be transmitted.

Pure FEC and Type-I Hybrid ARQ schemes can be implemented within CFDP depending on its operative modes: unacknowledged and

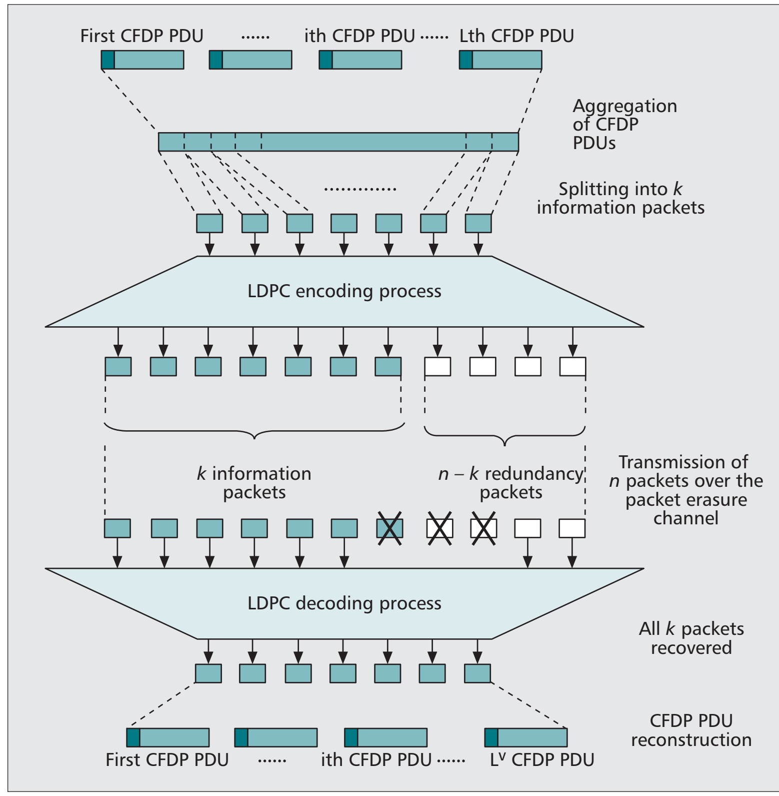


Figure 2. *LDPC encoding/decoding process for CFDP.*

acknowledged. In the first case the pure FEC scheme can be applied: the LDPC encoding schemes apply a high number of encoded information symbols for the sake of efficiency. To this end, at the source side, L CFDP PDUs are aggregated together and then split into k information units, which will be submitted to the LDPC encoding process responsible for the generation of the coded units, as shown in Fig. 2. Conversely, at the destination side, decoding operations will be successful upon reception of a sufficient number of coded units, thus allowing the aggregated CFDP PDUs to be reconstructed. If decoding operations cannot complete, a number of CFDP PDUs get lost. It is straightforward to see that the optimal selection of code rate, size of encoded information units, as well as number of aggregated CFDP PDUs is essential to improve system performance. This solution will be referred hereafter to as CFDP — Unacknowledged with Coding (CFDP-UC).

On the other hand, if CFDP works in acknowledged mode, it is possible to take advantage of both erasure codes and retransmission functions, thus implementing Type-I Hybrid ARQ. CFDP-Deferred is regarded as the most appropriate configuration, since data retransmissions are performed at the end of the file transfer, which is more effective in the presence of large propagation delays.[1] Encoding and decoding operations are performed exactly as for CFDP-UC. When decoding operations cannot be accomplished, the CFDP entity keeps track of the lost CFDP PDUs and issues deferred NAKs in order to recover the missing information units. The source side, upon reception of NAKs, re-encodes the missing PDUs and retransmits

[1] *In the presence of large delays, the use of immediate asynchronous, or prompted retransmission schemes results in long recovery phases occurring during the transmission of data PDUs, implying much longer data transfer latencies.*

The efficiency of packet-layer coding has to be checked against different performance indicators, according to specific service requirements such as information loss and data delivery latency.

them over the space link. The use of coding techniques also during the recovery phase is considered beneficial to increase the robustness against packet erasures, thus reducing the number of retransmission loops. This solution is hereafter referred to as CFDP — Acknowledged Deferred with Coding (CFDP-ADC).

Using erasure codes to ensure reliable data delivery is expected to be an effective alternative to ARQ schemes since retransmission strategies performed over deep space links are likely to degrade because of very large propagation delays. On the other hand, the implementation of coding techniques alone is not sufficient to guarantee successful file delivery, because there is some residual information loss that might be fully recovered by ARQ schemes. Nevertheless, it is also worth noting that the importance of file transfer reliability and respect for delivery constraints basically depend on the specific file content. Images and measurement file transfers can tolerate some information loss, but emergency or system messages should be delivered in a timely manner and without any information loss. This differentiation opens the door to QoS management performed at the CFDP entity by tuning protocol settings in order to match specific file transfer requirements. Also, power consumption and implementation complexity issues cannot be neglected because spacecrafts and remote planet stations (landers and rovers) have strict system design constraints. The implementation of erasure codes may imply waste of bandwidth and power roughly proportional to the amount of generated redundancy symbols, thus requiring attentive configuration of the coding parameters. In addition, the encoding process requires the storage of CFDP PDUs in proper buffers, performed before the encoding process. At the receiver side, decoding operations need memory space sufficient to accommodate the symbols actually necessary to successfully reconstruct the original CFDP PDUs. In addition, if the erasure codes are complemented by retransmission operations (as for CFDP-ADC), the source side has to provide space sufficient to store the entire file. Bandwidth and power waste is approximately proportional to the amount of retransmitted data.

All these factors along with performance figures (i.e., file transfer reliability and delivery delay) play a fundamental role in the design of an effective space telecommunication system. Some relevant case studies are illustrated in the following in order to identify the most effective protocol configurations.

Protocol Performance Evaluation

Deep Space Link Modeling

As introduced earlier, the physical peculiarities of deep space environments (solar winds, flares, thermal noise), the reduction of the link budget margin due to atmospheric events occurring on the downlink, along with the possible loss of synchronization at the receiver side give rise to correlated symbol erasures, which may range from a few up to hundreds of data link layer frames. From the viewpoint of the performance observed at the application layer (i.e., CFDP), the successful delivery of data alternates with occurrences of missed PDUs. Given the correlated nature of the deep space link, the erasure channel can be modeled as a first order two-state discrete-time Markov chain (DTMC) embedded in the transmission of each PDU at the application layer. Two states, ON and OFF, are considered: no information loss is observed in state ON, whereas erasures are experienced in state OFF. According to [15], erasure rates ranging from 0.1 up to 0.4 can be considered by varying the average duration of the OFF state depending on the frame length; on the other hand, the average duration of the ON state is kept fixed. In this way, it is possible to relate the performance of the application layer to the length of the data link layer frames. In particular, frame lengths ranging from 128 to 1024 bytes are considered, with a corresponding average number of erased symbols ranging from 300 kb up to 1 Mb (in line with values presented in [15]).

As far as propagation delay configurations are concerned, attention is focused on delays ranging from 1 up to 200 s, in order to consider both near-Earth and deep space missions. The link bandwidth is set to 1 Mb/s and 1 kb/s for downlink and uplink, respectively.

Case Study: Main Results

The efficiency of packet-layer coding has to be checked against different performance indicators according to specific service requirements such as information loss and data delivery latency. The comparison of the proposed CFDP enhancements, CFDP-UC and CFDP-ADC, with respect to standard CFDP-Deferred is performed accordingly. In particular, the following protocol configurations have been considered in order to identify the role played by the key parameters in system performance:

- CFDP-UC and CFDP-ADC: CFDP PDUs are aggregated into information vectors carrying 1 Mbyte. Code rates varying between 0.125 and 0.875 are considered. LDPC codes achieving near-Shannon limits are considered, with code inefficiency (ε) equal to 0.04
- CFDP-Deferred. The CFDP PDU length is varied between 1024 and 65,536 bytes.
- As reported earlier, the length of data link layer frames ranges between 128 and 1024 bytes. Encapsulation issues and overhead introduced by the overlying layers are considered accordingly.

Three metrics are considered primarily: information loss (*ILoss*), data delivery latency (*DDLatency*), and normalized goodput (*NGoodput*). *ILoss* is defined as the ratio between the number of correctly received and transmitted CFDP PDUs; *DDLatency* as the time duration elapsing from the transmission of the first CFDP PDU and the correct reception of all the CFDP PDUs. *NGoodput* is defined as the ratio between the amount of data correctly received at the destination and the time duration required by the transfer, normalized to the available bandwidth. The performance of the protocol solutions is tested by considering transfers of 100-Mbyte files, achieved between two space nodes that implement a full CCSDS protocol stack.

ILoss gives interesting indications, especially in the case of CFDP-UC, since the other solutions can guarantee full delivery of data by using retransmission functionalities. In particular, the impact of data link layer frame length and code rate deserves some attention (Fig. 3). The adoption of code rate 0.875 does not offer satisfactory performance because the reduced number of redundancy packets is not able to counteract the erasures caused by link errors independent of the coded packet size. If the code-rate is equal to 0.750, registered performance starts with *ILoss* equal to 1 when the sent frames carry 128 bytes; then the loss falls down below 0.3 as the frame size increases its length to 256 bytes. *ILoss* increases again while the coded packet size increases its length from 384 to 768 bytes and keeps approximately the same value for 1024. Concerning the other code rates, all registered *ILoss* values overlap for any frame size, even if it is possible to observe that when the frame size is set to 512 bytes, the most satisfactory result is achieved: *ILoss* of 0.07.

DDLatency and *NGoodput* performance is obviously dependent on the propagation delay, especially in the case of CFDP-ADC and CFDP-Deferred since they can resort to retransmission procedures in case of CFDP PDU erasures.

Besides, the CFDP PDU and frame length play an important role in CFDP-Deferred performance. In general, large PDUs at both application and data link layerd help reduce the overall transmitted overhead, thus improving the protocol performance in case of no packet erasures. In the investigated case, however, the larger the frame length, the longer is the average duration of OFF periods, according to the erasure channel model assumed in this article. Hence, the frame size also impacts on the number of packet erasures, and then on the amount of retransmissions to be performed during the recovery phase. On the other hand, the CFDP PDU length influences the transmitted overhead and the duration of the retransmission phases: the larger the CFDP PDUs, the longer the recovery phase is likely to be. Collected results show that setting CFDP PDU and frame length to 4096 bytes and 512 bytes, respectively, represents the best compromise to achieve the highest performance. As far as CFDP-ADC is concerned, the performance is ruled by both code rate and frame length. A large number of transmitted redundancy symbols helps recover the information erasures at the cost of bandwidth waste; on the other hand, the frame length, as already observed, impacts on link reliability in terms of average number of erasures. In this case it is important to point out that the combined use of packet-layer coding and retransmissions allows achieving satisfactory performance even with high code rate. Actually, the best protocol configuration is given with frame length and code rate set to 512 bytes and 0.75, respectively. The application of a higher amount of redundancy proved to not be beneficial to performance, because, although reducing the number of retransmissions, it caused non-negligible waste of bandwidth. Conversely, in the case of low code rate, a larger number of retransmissions takes place, thus affecting the overall performance because of the large propagation delays experienced in the investigated environment.

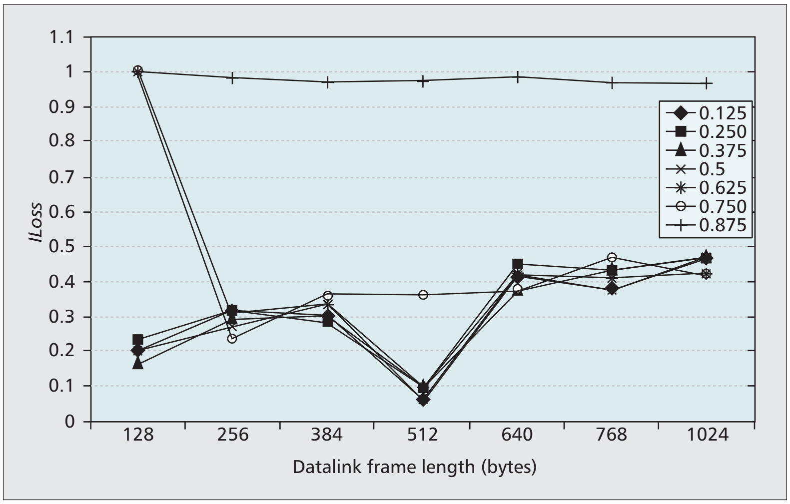


Figure 3. *CFDP-UC performance: information loss vs. data link frame size for different code-rates.*

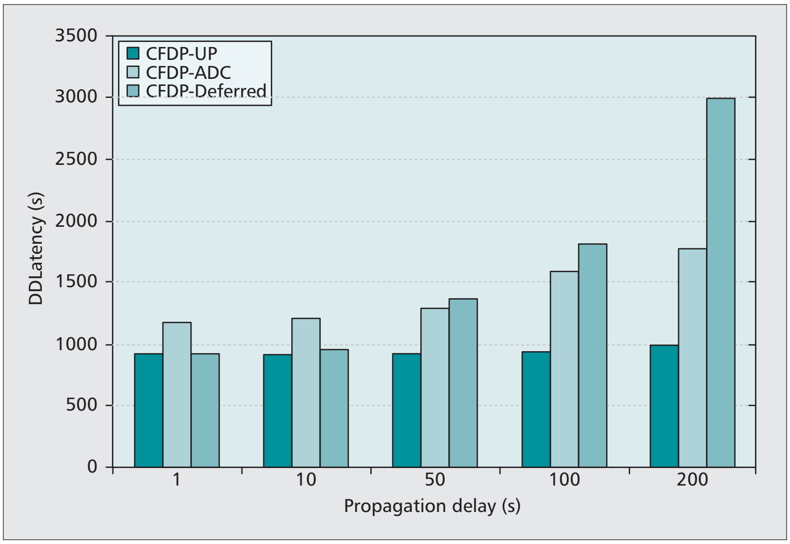


Figure 4. *Protocol solution performance: data delivery latency.*

As observed for CFDP-ADC, the performance of CFDP-UC is also influenced by code rate and frame length. The main difference is that no retransmission can be used to help recover erasures; hence, an attentive configuration of the aforementioned two parameters is necessary to attain satisfactory results. As anticipated in the analysis of *ILoss*, the most effective combination is given by code rate 0.750 and frame length 512 bytes.

The overall protocol performance is depicted in Figs. 4 and 5, where only the most effective configurations are considered for the sake of simplicity. In particular, Fig. 4 shows the *DDLatency*: CFDP-ADC performance is worse than CFDP-Deferred for lower delays since the retransmission of both information and redundancy packets is performed. On the other hand, as delay jumps over 50 s, the trend inverts and the *DDLatency* of CFDP-ADC is almost half of the *DDLatency* of CFDP-Deferred for a propagation delay of 200 s. *DDLatency* values registered for CFDP-UC are always below 1000 s since no retransmission is performed, at cost of partial data delivery, as shown in Fig. 4.

The *NGoodput* is depicted in Fig. 5: CFDP-Deferred allows using network resources efficiently (*NGoodput* varies between 0.9 and 0.6) when the propagation delay is lower than 50 s. As the delay increases (i.e., from 50 s up to 200 s), the use of pure ARQ schemes is not effective. CFDP-UC and CFDP-ADC are less efficient for low delays but offer better results, varying from 0.6 to 0.4, depending on the propagation delay (Fig. 5); larger delays highlight the main performance limits of pure ARQ mechanisms.

Finally, a very important factor that impacts on system design is given by hardware/software constraints, considered here in terms of power budget and storage capacity. In order to take into account power consumption issues, the total number of transmitted data (comprehensive of both redundancy and retransmitted bytes) is considered as a rough measure of expended power. Storage requirements are considered in terms of buffer space availability, which is needed by each considered protocol solutions to either perform retransmissions or coding/decoding operations. The sum of these two factors (both measured in Mbytes), referred to as Network Resource Cost (*NRCost*) allows shedding light on the relation between achieved performance and implementation cost. In particular it is interesting to see how *NRCost* scales in correspondence of different *ILoss* targets. In particular three *ILoss* targets were considered: 0.025, 0.05, and 0.15 respectively. These *ILoss* targets are referred in terms of three different profiles: A (*ILoss* $\leq$ 0.025), B (0.025 < *ILoss* $\leq$ 0.05), and C (0.05 < *ILoss* $\leq$ 0.15).

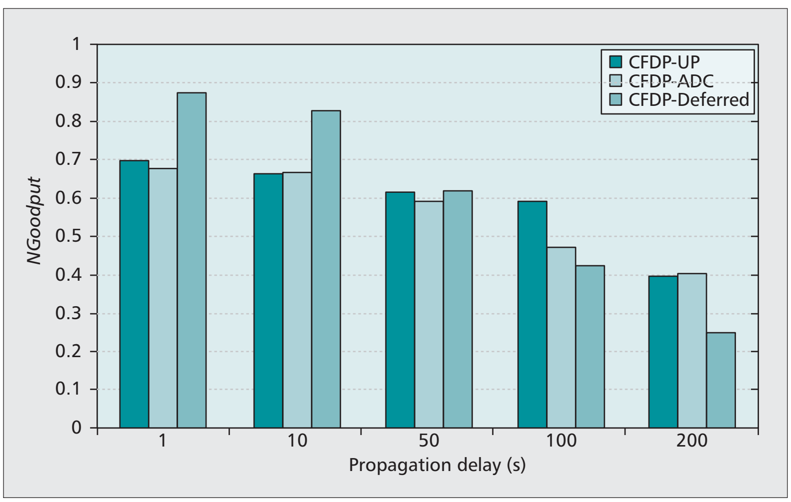


Figure 5. *Protocol solution performance: normalized goodput.*

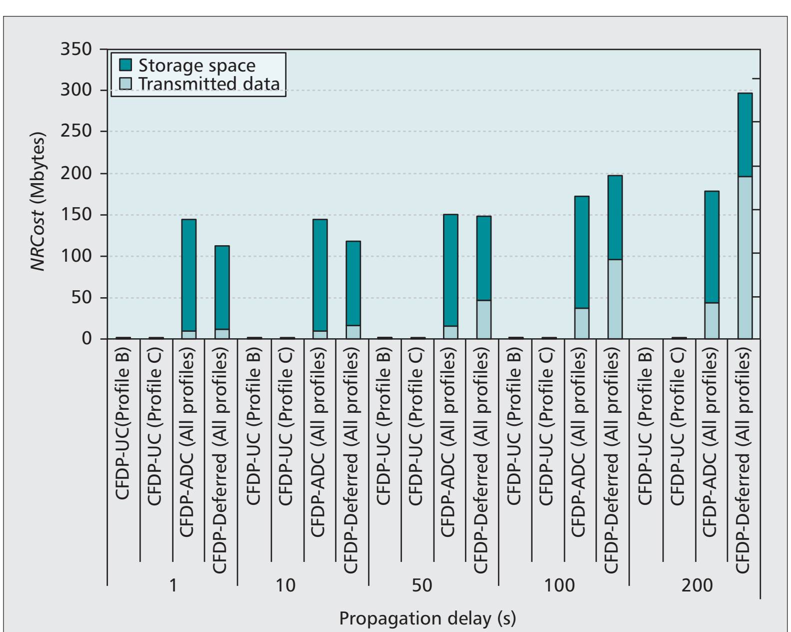


Figure 6. *Protocol solution performance: network resource cost.*

It is relevant for space system design to identify the cost components (in terms of retransmitted data and storage requirements) needed by the protocol configurations to minimize the data delivery delay, subject to the *ILoss* constraints imposed by profiles A, B, and C. The performance offered by the protocol solutions in terms of the *NRCost* is shown in Fig. 6 vs. the propagation delay and the profile that can be satisfied. *All profiles* in Fig. 6 implies that profile A (the most demanding) is satisfied together with the other two, which are less restrictive. The differentiation between the number of transmitted bytes and the storage capability required by each node is highlighted through two different colors. Immediately we see that CFDP-Deferred and CFDP-ADC can match all *ILoss* targets as they allow reliable communications. On the contrary, CFDP-UC can match only profiles B and C, but with very limited cost, whereas CFDP-ADC and CFDP-Deferred require higher cost because of the retransmission procedures that imply a larger amount of data to be transmitted along with increased storage capacity. Nevertheless, it is important to point out that only CFDP-ADC and CFDP-Deferred can match profile A's target; in this view, CFDP-ADC, especially for long propagation delays, offers a better trade-off between retransmitted data and storage space. This observation further confirms the advantages offered by erasure coding, which is helpful in matching profiles B and C at low cost (CFDP-UC) and also profile A (CFDP-ADC), though at much higher cost.

Finally, it is also worth noting that as the cost is a rough measure of the power consumption, the current power budget available on spacecraft and planetary nodes pushes toward CFDP-UC for its quite satisfactory performance results.

Discussion of Results

The performance figures presented in the previous section show the relationship between protocols' effectiveness and their configuration in terms of key parameters (e.g., code rate, CFDP PDU, and frame length) tested with different propagation delays (1–200 s) and erasure rates (0.1–0.4). The most important indication coming from the performance analysis is that it is possible to identify the most appropriate protocol solution and the related configuration independent of the data content to be transmitted. It is worth noting that in spite of the powerful protection against erasures provided by LDPC codes, the use of ARQ-based schemes such as CFDP-Deferred and CFDP-ADC is more promising in cases where the full integrity of the file has to be guaranteed. On the other hand, when a primary requirement is minimal delivery latency as requested by alarm messages or other immediate notifications, the use of CFDP-UC is to be considered. Finally, special note must be made of the current technology constraints posed in space missions in terms of limited storage and processing capacity, thus calling for low-complexity solutions. This has a straightforward impact on the applicability of ARQ-based solutions, which require larger memorization units with respect to pure coding-based schemes. As a consequence, the use of CFDP-UC seems more appropriate to meet the various performance requirements that can be demanded in space missions. Furthermore, the use of pure erasure coding schemes requires the optimization of a limited number of parameters (code-rate and frame length), actually independent of the application layer implementation. On the contrary, CFDP-ADC and CFDP-Deferred also require precise tuning of CFDP PDU length according to that of data link layer frames in order to avoid encapsulation and fragmentation troubles, which are usually not desired in data communications.

Conclusions

This work focuses on the performance requirements that future telecommunication infrastructures for space environments may pose, by analyzing the features the CCSDS protocol stack may offer, with particular respect to the higher layers. In particular, the packet-layer coding methodology possibly combined with ARQ mechanisms has been explored and investigated. The rationale behind the use of erasure codes stems from the necessity of limiting data retransmission operations due to the very large latencies experienced in interplanetary networks. To this end, two proposals whose design is inherited from CFDP-Deferred have been tested in order to show the benefits the application of erasure coding may bring in deep space communications. The performance analysis confirmed the potential of this approach by highlighting the advantages offered by the new strategies in terms of file transfer reliability, data delivery time, resource network utilization and power consumption.

Acknowledgments

This work was partially funded by the European Community in the framework of the FP6 SatNEx NoE project. The project developed in two phases: SatNEx I and II, contract no. 507052 and no. 027393, respectively. Finally, the authors would also like to thank Harald Ernst from ESA and Gianluigi Liva from DLR for their very helpful suggestions and comments about aspects related to packet-layer coding.

References

[1] K. Bhasin *et al.*, "Advanced Communication and Networking Technologies for Mars Exploration," *19th Annual AIAA Int'l. Commun. Satellite Sys. Conf.*, Toulouse, France, Apr. 2001.

[2] S. Burleigh *et al.*, "The InterPlaNetary Internet: A Communications Infrastructure for Mars Exploration," *53rd Int'l. Astronautical Congress — The World Space Congress*, Houston, TX, Oct. 2002.

[3] I. F. Akyildiz *et al.*, "The State of the Art in Interplanetary Internet," *IEEE Commun. Mag.*, vol. 42, no. 7, July 2004, pp. 108–18

[4] CCSDS, "Overview of Space Communications Protocols," *CCSDS 130.0-G-2, Green Book*, no. 2, 2001.

[5] K. B. Bhasin, "Interplanetary Internet," *Comp. Net.*, vol. 47, no. 5, Apr. 2005.

[6] E. Modiano, "Space Networks," *Int'l. J. Satellite Commun. Net.*, vol. 25, no. 2, Mar.–Apr. 2007.

[7] T. de Cola, H. Ernst, and M. Marchese, "Performance Analysis of CCSDS File Delivery Protocol and Erasure Coding Techniques in Deep Space Environments," *Comp. Net.*, vol. 51, no. 14, Oct. 2007, pp. 4032–49.

[8] G. P. Calzolari *et al.*, "Channel Coding for Future Space Missions: New Requirements and Trends," *Proc. IEEE*, vol. 95, no. 11, Nov. 2007, pp. 2157–70.

[9] L. Rizzo, "Effective Erasure Codes for Reliable Computer Communication Protocols," *ACM Comp. Commun. Rev.*, vol. 27, no. 2, Apr. 1997, pp. 24–36.

[10] I. S. Reed and G. Solomon, "Polynomial Codes over Certain Finite Fields," *J. Soc. Industrial Applied Math.*, vol. 8, no. 2, June 1960, pp. 300–4.

[11] A. Shokrollahi, "Raptor Codes," *IEEE Trans. Info. Theory*, vol. 52, no. 6, June 2006, pp. 2551–67.

[12] E. Paolini *et al.*, "Generalized IRA Erasure Correcting Codes for Hybrid Iterative/Maximum Likelihood Decoding," *IEEE Commun. Lett.*, vol. 12, no. 6, June 2008, pp. 450–52.

[13] CCSDS, "CCSDS File Delivery Protocol, Recommendation for Space Data System Standards," *CCSDS 727.0-B-4, Blue Book*, no. 3, 2007.

[14] M. Chiani, G. Liva, and E. Paolini, "Long Erasure Correcting Codes: A New Appealing Chance for Space Applications Protocols?," CCSDS Meeting, Montreal, Canada, May 2004; http://cwe.ccsds.org/sls/docs/SLS-CandS

[15] D. Divsalar and S. Dolinar, "Long Erasure Correcting Codes," CCSDS Meeting, Washington, DC, Mar. 2008; http://cwe.ccsds.org/sls/docs/SLS-CandS

Biographies

Tomaso de Cola (tomaso.decola@dlr.de) received his Laurea degree (summa cum laude) in telecommunication engineering from the University of Genoa, Italy, in 2001 and his Qualification degree as a Professional Engineer in 2002. From 2002 until 2007 he worked with the Italian Consortium of Telecommunications (CNIT), University of Genoa Research Unit as a scientist researcher. Since 2008 he has been with the German Aerospace Centre (DLR), where he is involved in different European projects focusing on different aspects of DVB standards, CCSDS protocols and testbed design. He is a co-author of more than 30 papers, including international conferences and journals. His main research activity concerns TCP/IP protocols, satellite networks, transport protocols for wireless links, and interplanetary networks, as well as delay-tolerant networks.

Mario Marchese [S'94, M'97, SM'04] (mario.marchese@unige.it) received his Laurea degree cum laude at the University of Genoa in 1992 and his Qualification as a Professional Engineer in April 1992. He obtained his Ph.D. (Italian Dottorato di Ricerca) degree in telecommunications at the University of Genoa in 1996. From 1999 to 2004 he worked with CNIT, University of Genoa Research Unit, where he was head of research. Since February 2005 he has been an associate professor at the Department of Communication, Computer, and Systems Science (DIST), University of Genoa. He is the founder of and still technically responsible for the CNIT/DIST Satellite Communications and Networking Laboratory (SCNL) at the University of Genoa. He chaired the IEEE Satellite and Space Communications Technical Committee from 2006 to 2008. He is an author and co-author of about 200 scientific works in international magazines, international conferences, and book chapters, and the book *Quality of Service over Heterogeneous Networks* (Wiley, 2007). His main research activity concerns are satellite and radio networks, transport layer over satellite and wireless networks, quality of service and data transport over heterogeneous networks, and emulation and simulation of telecommunication networks and satellite components.

The performance analysis confirmed the potentials of this approach by highlighting the advantages offered by the new strategies in terms of file transfer reliability, data delivery time, resource network utilization and power consumption.