Study and Performance Analysis of ARQ-based and Transport Layer Coding Schemes over Deep Space Networks

Tomaso de Cola*, Harald Ernst°, Mario Marchese*
*CNIT - Italian National Consortium for Telecommunications
Genoa Research Unit, University of Genoa, Via Opera Pia 13, 16145, Genoa, Italy
Email: {Tomaso.decola, Mario.Marchese}@cnit.it
°DLR- Deutschen Zentrum für Luft- und Raumfahrt, German Aerospace Center
Institute for Communications and Navigation, 82234 Wessling, Germany
Email: harald.ernst@dlr.de

Abstract — **The extension of telecommunication frontiers towards deep space scenarios has opened new horizons in the way of designing network infrastructures and has raised the necessity of developping novel communication paradigms, more suited to this specific environment. Under this view, the exploitation of TCP-based transmission schemes does not offer satisfactory results. On the contrary, the use of erasure coding schemes and more appropriate Automatic Repeat reQuest(ARQ) schemes, available within the Transport Layer Coding-based and the Consultative Committee for Space Data Systems (CCSDS) protocol architectures, respectively, assure better performance results.**

In this paper, the adoption of erasure codes within CCSDS protocol stack is considered and its effectiveness is evaluated with respect to ARQ-based transmission schemes available within the CCSDS File Delivery Protocol.

I. INTRODUCTION

SINCE the end of eighties, the exploration of space and the proliferation of scientific experiments have shown, on the one hand, the necessity of reliable telecommunication infrastructures and, on the other hand, have revealed the shortcomings deriving from the use of TCP-based protocols. In particular, the large latencies experienced by typical deep space environments negatively affect the TCP performance because of its transmission paradigm based on a feedback scheme [1]. In this perspective, the features offered by the Consultative Committee for Space Data Systems (CCSDS) recommendations in terms of suspending and resuming capabilities are an effective resource to assure reliable data communication over space networks. Moreover, the support of highly efficient ARQ schemes available within the CCSDS File Delivery Protocol (CFDP) helps improve the overall data communication performance in terms of both throughput and loss recovery effectiveness. Another possibility is also represented by the implementation of erasure coding schemes, adopting the Transport Layer Coding approach [2]. Starting from the aforementioned issues, this paper analyses the use of the Transport Layer Coding approach within the CFDP implementation and hence proposes a combined use of erasure coding and ARQ schemes to improve the overall performance.

The remainder of the paper is organized as follows. The state of the art is addressed in Section II, while Section III considers the characteristics of the deep space environment and introduces a suitable model to represent its behaviour. The CCSDS protocol architecture, the Transport Layer Coding approach and the issues regarding their joint use are shown in Section IV. The investigation completes in Section V, where the performance analysis of the different protocol solutions is shown; in Section VI the conclusions are drawn.

II. BACKGROUND

Over last years, the scientific community has made strong efforts to design appropriate protocols and architectures able to guarantee reliable data communication over space networks. From the standardization point of view, relevant contributions have been provided by the CCSDS institution together with the Delay Tolerant Network [3] working group within IRTF.

Furthermore, the study of alternative mechanisms, based on erasure coding schemes and aimed at guaranteeing reliable communications deserves a particular attention. In particular, the advantages offered by the long erasure codes, and in particular by Low Density Parity Check codes (LDPC) hint at employing these coding schemes in order to make data communication more robust against strong link degradations. Their adoption over the transport layer is identified as Transport Layer Coding and proposed in [2]. Further considerations about the software complexity issues, arising from LDPC implementations, and the related performance are addressed in [4]. Finally, protocols for transmitting efficiently data over deep space networks have been analysed and devised in [5], where TP-Planet solution, working at the transport layer, emerges as promising alternative to TCP.

This work takes the CCSDS File Delivery Protocol (CFDP) as reference and applies the Transport Layer Coding approach to improve the overall data communication performance over deep space networks.

III. THE DEEP SPACE ENVIRONMENT

A. *The Reference Scenario*

To better capture the environment peculiarities and hence properly study protocol implementations able to counteract the hazardous conditions in which the data communication is achieved, the following scenario is assumed. Two remote stations, placed on the Earth and on a remote planet (e.g. Mars or Moon), communicate each other by means of a deep space link established between two satellite platforms orbiting around Earth and remote planet, respectively. All the nodes implement a full CCSDS protocol stack, and in particular the CCSDS File

1-4244-0357-X/06/$20.00 ©2006 IEEE
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE GLOBECOM 2006 proceedings.

Delivery Protocol at the upper layers.

The whole scenario is depicted in Fig. 1.

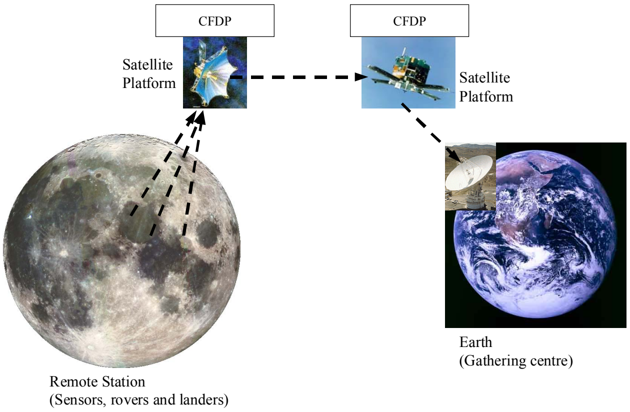

Fig.1. The Reference Scenario

B. *The Deep Space Link*

The strong impairments introduced by deep space links, such as deep fading periods, blackout events and variable propagation delays, have to be properly taken into account while designing transmission schemes suited to space environments. In this view, the adaptation of common models employed to characterize wireless transmission channels is an appropriate solution. In particular the use of Discrete-Time Markov Chains (DTMC) has been assumed to represent the channel behaviour; in detail, the use of a first order Markov chain with 4 states is proposed.

The transition between two arbitrary consecutive states, i and j , is ruled by the transition probability matrix $P = \{p_{i,j}\}$. On the other hand, the steady-state probability of being in the ith state is denoted as π_i, where $i \in \{0,1,2,3\}$. Each state accounts the channel reliability by means of the Bit Error Ratio (BER), evaluated at the receiver side after the channel decoding procedures. In practice, a BER value equal to BER_i is assigned to the ith state; for consecutive states, the following inequality holds: $BER_i < BER_j \ \forall i,j \in \{0,1,2,3\}$, with $i < j$. Besides, a relevant parameter influencing the link behaviour is the mean permanence time within the ith state, indicated in the following as τ_i.

Finally, to fully evaluate the impact of corrupted bits on the transmission performance, it is also necessary to provide a statistical characterization of the packet loss process. Under this view, the use of the GAP error length model is promising. In practice, error and error-free gaps are defined as occurrences of consecutive successful and unsuccessful received packets, respectively.

IV. THE PROPOSED PROTOCOL ARCHITECTURE

A. *The CCSDS File Delivery Protocol (CFDP)*

CFDP transmitting entity assembles data into PDUs, identified in the following as CFDP blocks, whose payload can carry up to 65536 bytes, while the header length is assumed here of 20 bytes. The reliability issues are addressed by the CFDP entity in dependence of the operating mode in which it is configured, either acknowledged or unacknowledged. In the latter, no specific options for assuring the communication reliability are implemented. On the other hand, when CFDP operates in acknowledged mode, the communication reliability is assured my means of negative acknowledgments (NAK, issued by the receiving CFDP entity). Once the loss of a data block is detected, the recovery mechanism is ruled by four different algorithms: immediate, prompted, asynchronous and deferred. In particular when the deferred option is set, the receiver checks if CFDP blocks are missing only at the end of the data communication. When missing blocks are detected, the recovery phase is invoked at the receiver side, by sending NAK blocks to the sender, which will be responsible for retransmitting the lost blocks.

Finally, a particular note has to be dedicated to suspending and resuming features provided by CFDP. In particular, when the protocol entity is configured to operate as in “extended operations”, it is able to suspend the transmission on the basis of notifications, indicating the unavailability of the transmission medium, issued by lower layer protocols. Afterwards, data blocks are temporally stored in a local CFDP buffer; the transmission is resumed again once positive notifications about the channel availability are provided.

B. *Proposed CFDP improvements*

In this work, CFDP working in both acknowledged and unacknowledged modes is investigated. The proposed CFDP improvements regard the use of erasure coding schemes, aimed at guaranteeing reliable exchange of data also when the communication is performed in very hazardous conditions. In practice, two protocol proposals have been conceived, namely “CLDGM” and “CLDGM-deferred”, whose description follows.

CLDGM. It concerns the integration of erasure coding schemes into CFDP protocol when running in unacknowledged mode, by applying the Transport Layer Coding approach as shown in [2] and [6]. In practice, the adoption of LDGM codes, derived from the Low Density Parity Check codes, is considered for their capacity of protecting data communication against bursty data losses. In facts, the integrated scheme works as follows: CFDP aggregates different data blocks, split them into k information "packets", and hence encode them into n packets, by means of the LDGM generator matrix. It is straightforward that the LDGM performance strictly depends on the ratio among the number of encoded packets and the total number of generated packets, referred in the following as code-rate. In particular, in this work, “k” has been fixed to 1000, and code-rate values ranging from 0.125 up to 0.875 have been considered and, for the sake of the completeness, block and packet sizes varying from 1024 to 65536 bytes and 128 to 1024 bytes, respectively have been taken into account in order to evaluate the impact of link errors on the overall performance. In the following, this approach will be referred as *CLDGM* (which stands for CFDP with LDGM codes).

CLDGM-deferred. The second approach combines the use of NAK PDUs with LDGM codes in order to allow data retransmission when LDGM effectiveness is not sufficient. In practice, the integration of LDGM codes within CFDP follows the implementation adopted in the CLDGM proposal. In particular, even in this case the number of encoding packets (k) has been fixed to 1000. The deferred issuance of NAK PDUs, on the other hand, conforms the CFDP specification. Code-rate and packet sizes have been varied, during the tests, within the same intervals as CLDGM. This proposal will be referred in the following as *CLDGM-deferred*. For the sake of the completeness, the two proposals have been compared with CFDP working in the following configurations:

- acknowledged mode, with deferred NAK. This scheme is indicated in the performance analysis as *CFDP-deferred*;
- unacknowledged mode, with extended operations. In this case, the *a priori knowledge* of the transmission medium availability help achieve reliable communications without necessity of either data retransmissions or employment of erasure codes. In practice, the transmission of new data blocks is scheduled once the channel is in state 0. This solution is actually an "ideal solution" and has been taken into account in order to assess the effectiveness of the other solutions. This scheme is indicated in the following as *CFDP-extended*.

Finally, for the sake of the clarity, only the configurations in terms of code-rate, CFDP block and packet size, providing the highest performance results have been considered, as reported in next section.

V. THE PERFORMANCE ANALYSIS

A. The testbed

The investigation has been focused on the transfer of data between two remote peers, implementing a full CCSDS stack. For the sake of the analysis, a transfer of 100 Mbytes has been considered. Tests have been accomplished through a simulation tool designed for the aim. A number of runs sufficient to obtain a width of the confidence interval less than 1% of the measured values for 95% of the cases has been imposed.

As far as the deep space transmission medium is concerned, the forward-link bandwidth is set to 1Mbit/s, while the reverse link has availability for 1Kbit/s. The propagation delays in the reverse and forward directions are equal and ranging from 0.250s to 200s for each experiment. The states within the DTMC model assume BER values equal to 10^{-8}, 10^{-6}, 10^{-4} and 10^{-2}, for states 0, 1, 2, and 3, respectively. Moreover, the steady state probability π_2 and π_3 has been fixed together with the average permanence times τ_0 and τ_3 within states 0 and 3 in order to evaluate the effectiveness of the proposals. In particular four case studies have been identified (scenarios 1, 2, 3 and 4), in dependence of τ_0 and τ_3 values, in order to show the different impact of bursty losses on the communication reliability.

B. The metrics

The probability of missing a CFDP block, indicated as Loss Probability (P_{loss}) and defined as one minus the ratio among the transmitted and received blocks, is the performance metric together with the real use of the channel, indicated as Effective Throughput. The latter is measured as the product of $(1-P_{loss})$ and the ratio of the Transfer Size and the Transfer Time evaluated as the time elapsed from the transmission of the first bit and the reception of the last one. In facts:

$$P_{loss}=1-(\text{Received Blocks}/\text{Transmitted Blocks})$$

$$\text{Effective Throughput} = (1-P_{loss})\cdot\frac{\text{Transfer Size}}{\text{Transfer Time}}\cdot\frac{1}{\text{Bandwidth}}$$

In order to characterize the different performance constraints of the traffic transported through CFDP blocks, five classes of service have been introduced, A, B, C, D, E, presenting different constraints in terms of the maximum P_{loss} and Transfer Time acceptable. Actually, three thresholds for P_{loss}, namely P_{loss_1}, P_{loss_2}, P_{loss_3}, and equal to 0.025, 0.05 and 0.15, respectively, are chosen. As regards the constraints on the Transfer Time, taking as reference the minimum time, T_{min}, required to accomplish the whole transfer of data (equal to the ratio between the Transfer Size and the Bandwidth, plus twice the propagation delay), two thresholds T_1 and T_2 have been set. The whole classification is shown in Table I.

TABLE I
CLASSES OF SERVICE AND RELATED PERFORMANCE CONSTRAINTS

Class of Service	Delivery Time	Loss Probability
A: spacecraft location data and classes of telemetry data updates .	$< T_1=2\ T_{min}$	$< P_{loss\_2}$
B: critical instrument status notification or urgent remote control commands.	$< T_1=2\ T_{min}$	$< P_{loss\_3}$
C: measurements, planet's surface images.	$< T_2=4\ T_{min}$	$< P_{loss\_3}$
D: periodic notifications bulks of data sent on a best-effort basis.	$<T_2=4\ T_{min}$	$< P_{loss\_2}$
E: other file transfers.	any	$< P_{loss\_1}$

C. The results

Scenario 1 (τ_0=20s, τ_3=5s)

In this configuration, since the average time spent in state 0 is much longer than state 3, the error gaps have a moderate length. Consequently Loss Probability requirement has no great impact on all the tests. In particular, since loss probabilities less or equal to 0.05 are experienced for classes A and B as well as C and D, the investigation (reported in Fig. 2) addressed to the performance exhibited by classes A, D and E, as shown in Fig. 2.

In general, it is possible to see that CFDP-extended, which represents an ideal protocol solution, outperforms the other proposals because of its capabilities of transmitting data when the channel is reliable. As far as class A is concerned, CLDGM achieves the best performance results independently of the propagation delays. In fact, the Effective Throughput measured for CLDGM ranges from 0.85 to 0.45 as propagation delays vary from 0.25 s to 200 s. CLDGM-deferred too performs efficiently in the case of delays ranging from 0.25s to 50s, achieving performance results very close to CLDGM one: from 0.85 down to 0.72, while CLDGM gives 0.78 for 50s. For larger delays, the recovery phase get longer and, consequently, Effective Throughput drops to 0.45. Finally, CFDP-deferred shows poor performance, and in particular in the case of 100s and 200s, it is unable to match the performance requirements. As far as class D is concerned, CLDGM provides the best performance results from 0.855 to 0.69 as delay varies from 0.25 to 100s. Once the delay raises up to 200s, CLDGM-deferred gives more satisfactory results because combining erasure codes and retransmission procedures help reduce the loss recovery operations. In fact, Effective Throughput achieves 0.52 for 200s, while CLDGM does not come over 0.45.

Finally, with class E, given the total relaxation of the delay constraint combined with the severe loss constraint and with the limited length of the error gaps, CLDGM and CLDGM-deferred present the best results, ranging from 0.85 to 0.45, and from 0.85 to 0.52, respectively.

Scenario 2 (τ_0=60s, τ_3=5s)

In this study case, the effect of link errors on the overall performance is even more limited since the mean time spent in state 0 is much longer than in state 3. The discussion of the results, shown in Fig. 3, can be limited to classes A, D and E since identical performance results are offered by classes A and B, and C and D, respectively. In facts, CFDP-extended guarantees the highest effective-throughput values (for each class), ranging from 0.988 to 0.673 as the propagation delay is varied from 0.25s to 200s. As for class A, all the solutions present very similar results. In particular, CLDGM-deferred gives the best performance results, ranging from 0.88 to 0.55, as delay varies from 0.25s to 200s. As far as class D is concerned, CLDGM-deferred and CLDGM confirm performance observed for class A. In fact, they show results ranging from 0.865 to 0.381 and from 0.868 to 0.464, respectively. Finally, as in scenario 1, CLDGM and CLDGM-deferred provide the best results for class E, ranging from 0.86 to 0.46 and from 0.88 to 0.55 (for delay varying from 0.25s to 200s), respectively.

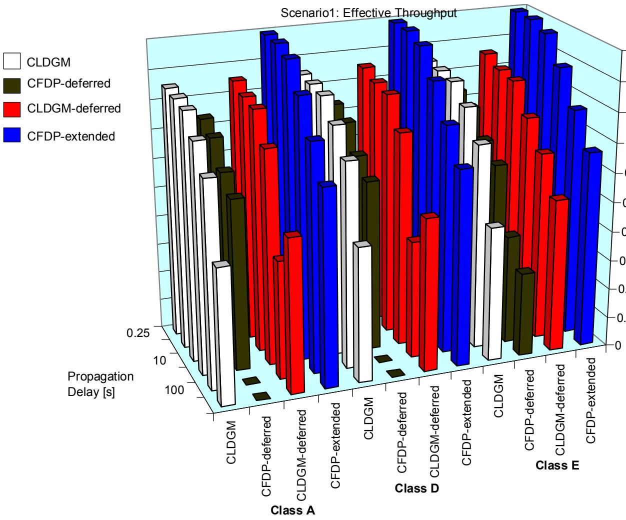

Fig. 2. Scenario 1: the overall performance

Scenario 3 (τ_0=5s, τ_3=20s)

The longer permanence in state 3, if compared to state 0, implies an increased length of error gaps and hence less effective results are expected. For the sake of the simplicity, the investigation does not take class D into account, since it does not add further information with respect to class C evaluation. In practice, apart from CFDP-extended that exhibits the most satisfactory results (effective throughput of 0.90-0.602), the other three solutions present performance results strictly dependent of the service classes, as shown in Fig. 4. In more details, for class A, CLDGM-deferred achieves the best performance results, varying from 0.69 to 0.38. As for class B, CLDGM-deferred again provides satisfactory results (0.68-0.59) as delay ranges from 0.25s to 50s. Once delay further increases, CLDGM performs better, giving Effective Throughput of 0.56 and 0.37 for 100s and 200s, respectively. Class C results show that CLDGM-deferred and CFDP-deferred prove to be powerful, achieving performance ranging from 0.69 to 0.38 in both cases. Finally, the strict constraints on Loss Probability of class E can be efficiently matched by CLDGM-deferred, achieving performance from 0.69 to 0.38.

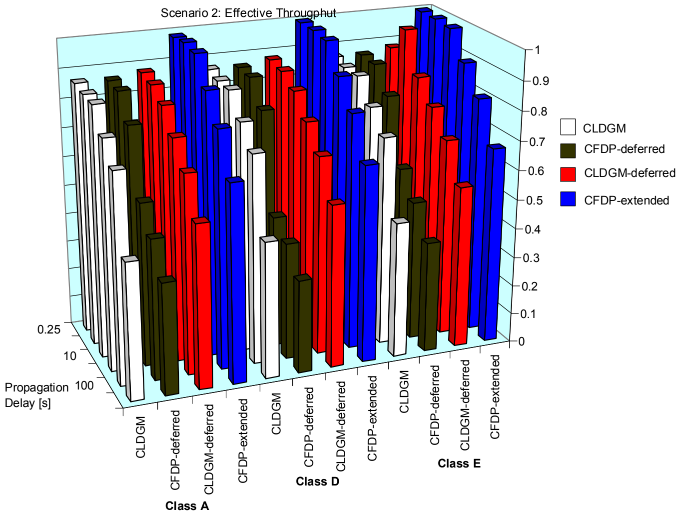

Fig. 3. Scenario 2: the overall performance

Scenario 4 (τ_0=5s, τ_3=60s)

As the permanence in state 3 gets longer (60s), the length of error gaps increases accordingly, severely affecting the global performance. In this case, meaningful results are provided by class A, C, D and E, while results for class B are aligned with class A. In particular, from Fig. 5, it is possible to realize that, apart from CFDP-extended that behaves almost ideally, CFDP-deferred and CLDGM-deferred completely outperform CLDGM, since the long error runs cannot be only counteracted by erasure code application. In more details, as far as class A is concerned, CFDP-deferred is the most promising for delays lower than 200s (Effective Throughput: 0.86-0.42). In the case of 200s, CLDGM-deferred is more performant, achieving 0.4. As for classes C and D, again CFDP-deferred ensures high performance results ranging from 0.86 to 0.25 in both cases. It is worth noting that CLDGM is unable to match class D performance constraints because of the strict constraints on Loss Probability. Finally, for class E, CFDP-deferred is efficient when the delay is lower or equal to 50s, achieving performance ranging from 0.86 to 0.61. Otherwise, in presence of larger delays, CLDGM-deferred offers better performance values, corresponding to 0.47 and 0.4 for delay of 100s and 200s, respectively.

D. *Comparison*

In order to evaluate completely the effectiveness of the proposed protocol solutions, CFDP-deferred, CLDGM and CLDGM-deferred are compared with CFDP-extended, in term of Efficiency (%), defined as ratio between Effective Throughput achieved by the above solutions (indicated as CFDP variants) and CFDP-extended:

$$Efficiency(\%) = \frac{\text{Effective Throughput (CFDP variants)}}{\text{Effective Throughput (CFDP-extended)}} \cdot 100$$

Since class B and C performance is aligned with classes A and D, respectively, the attention is paid only to classes A, D, and E, considering a delay of 100s.

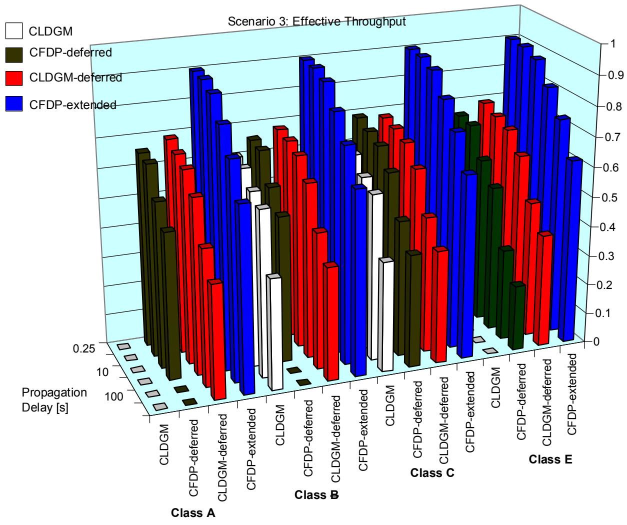

Fig. 4. Scenario 3: the overall performance

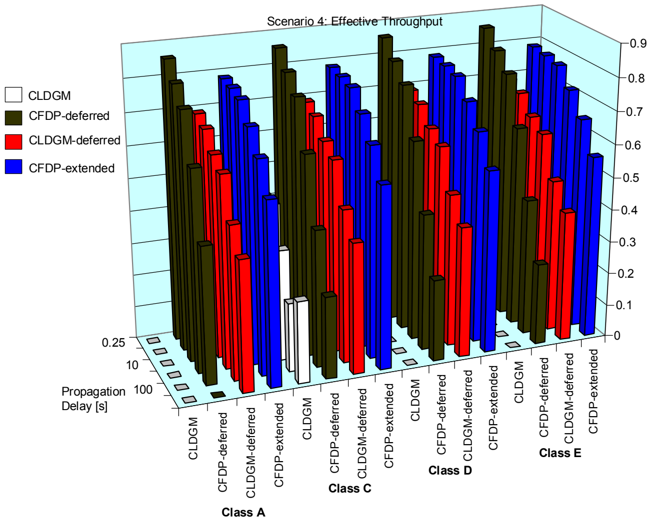

Fig. 5. Scenario 4: the overall performance

From Fig. 6, it is possible to see that when class A constraints have to be satisfied, CLDGM-deferred offers the best efficiency results when strong impairments (scenarios 4 and 3) are introduced by the channel. Actually, the combined use of erasure codes and retransmissions allows achieving the best performance results, corresponding to 72.30% and 62.87% for scenarios 4 and 3, respectively. On the other hand, when minor losses are exhibited, it is CLDGM that offers the most satisfactory efficiency results, equal to 88.5% and 90.06% for scenarios 2 and 1, respectively. As for class D and E, which require loss probabilities lower than 0.05 and 0.025 respectively, CLDGM-deferred always behaves better than CFDP-deferred, presenting efficiency values ranging from 84.05% to 55.30% and from 82.58% to 52%, respectively. Finally, in scenarios 1 and 2, when the class E requirements have to be satisfied, CLDGM-deferred performance is very close to CLDGM. In facts, CLDGM-deferred achieves results from 82.58% to 84.05% for scenarios 1 and 2 respectively, while CLDGM achieves 90.0% and 88.5%.

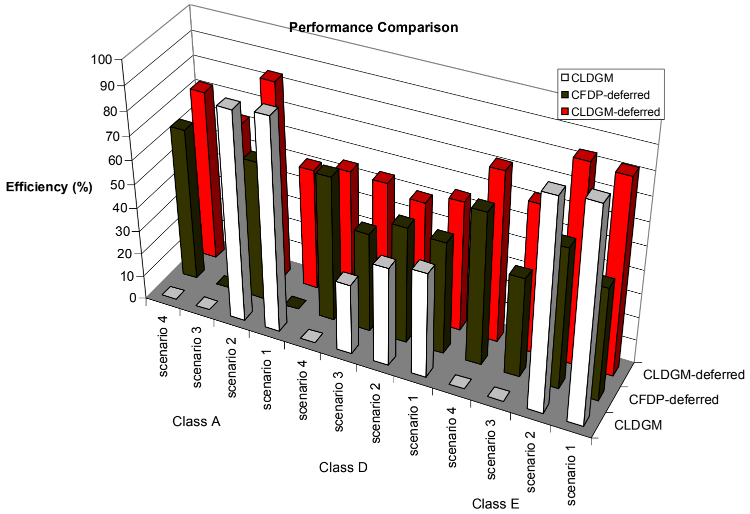

Fig. 6. Performance comparison (Efficiency %)

VI. CONCLUSIONS

This work has been devoted to the design of novel protocol solutions, based on the CCSDS File Delivery Protocol (CFDP), to achieve data communications over long-delay networks. Two proposals CLDGM and CLDGM-deferred have been introduced in this work and deeply investigated with respect to CFDP-deferred and CFDP-extended. The performance analysis, carried out for different scenario configurations, has identified CLDGM together with CLDGM-deferred as promising solutions, able to match the specific constraints of five classes of service. In particular, CLDGM, thanks to the powerful LDGM erasure codes, offers very satisfactory results in scenarios 1 and 2, where moderate losses are experienced. CLDGM-deferred, in these cases, is less efficient even if its behaviour is very satisfying. On the other hand, the adoption of CLDGM-deferred is "mandatory" when "almost reliable" data communications have to be carried out in very hazardous conditions, such as in scenarios 3 and 4. It allows considering CLDGM-deferred as an efficient solution, whose application is very wide.

REFERENCES

[1] D. C. Lee, W. B. Baek, "Expected file-delivery time of deferred NAK ARQ in CCSDS file-delivery protocol," IEEE Transactions on Communications, vol. 52, no. 8, Aug 2004, pp. 1408-1416

[2] H. Ernst, L. Sartorello, S. Scalise, "Transport layer coding for the land mobile satellite channel," 59th IEEE Vehicular Technology Conference (VTC'04), Milan, Italy, May 2004

[3] S. Burleigh, A.Hooke, L. Torgerson, K. Fall, V. Cerf, B. Durst, K. Scott, H. Weiss, "Delay-tolerant networking: An approach to interplanetary internet," IEEE Communications Magazine, vol. 41, no. 6, Jun 2003 pp. 128-136

[4] V. Roca, "Design, Evaluation and Comparison of Four Large Block FEC Codecs, LDPC, LDGM Staircase and LDGM Triangle, plus a Reed Solomon Small Block FEC Codec," INRIA, Rapport de Recherche, June 2004, http://www.inrialpes.fr/planete

[5] I. F. Akyildiz, Ö. B. Akan, C. Chen, J. Fang, W. Su, "The state of the art in interplanetary internet", IEEE Communications Magazine, vol. 42, no. 7, Jul 2004 pp. 108-118

[6] T. de Cola, H. Ernst, M. Marchese, "Joint Application of CCSDS File Delivery Protocol and Erasure Coding Schemes over Space Communications," IEEE International Conference on Communications (ICC 2006), Instanbul, Turkey, June 2006