

# LTP Performance on Near Earth Optical Links

CARLO CAINI, Member, IEEE

University of Bologna, Bologna, Italy

TOMASO DE COLA, Member, IEEE

Institute of Communications and Navigation, German Aerospace Center (DLR), Weßling, Germany

AMITA SHRESTHA

Institute of Communications and Navigation, German Aerospace Center (DLR), Weßling, Germany

ANTONY ZAPPACOSTA

University of Bologna, Bologna, Italy

**ABSTRACT** The Licklider Transport Protocol (LTP) has been designed to cope with long-delay and error-prone scheduled-intermittent links, and thus is envisaged as the Bundle Protocol (BP) “convergence layer” of choice in future Inter-planetary networks (IPN) based on the Delay-/Disruption-Tolerant architecture. Moreover, LTP’s remarkable ability to cope with multiple losses when operating in “red” reliable mode also makes it potentially appealing when coupled with Near Earth optical links. The aim of this paper is to assess LTP performance in this scenario, so we have developed a test bed based on real machines, real implementations of BP and LTP, and a channel emulator; this is based on “erasure vectors”, i.e. time series describing the on/off state of the optical link, derived from real data measurements conducted by DLR. Our results show that, when properly configured, LTP is able to use all available bandwidth even under the most severe conditions, which makes it a perfect match to Near Earth Optical links.

1

Manuscript received xxx 2023; revised xx, 2023; released for publication xxx, 2023.

DOI. No. xx/TAES.xxx

## I. INTRODUCTION

Space networks differ from ordinary terrestrial networks because of at least one or more of the following challenges, which prevent the use of ordinary TCP/IP architecture: long delays, intermittent scheduled connectivity, asymmetry of links, and possibly relatively high packet loss rates due to variable channel conditions. To cope with these, Delay-/Disruption-Tolerant Networking (DTN) architecture [1], [2], [3] is required. This architecture extends ordinary TCP/IP architecture by introducing an additional overlay layer, the Bundle layer, between Application and (usually) Transport. Its corresponding homonymous protocol, the Bundle Protocol (BP) [4], [5], [6] is in charge of transferring “bundles” between DTN nodes, possibly using different Transport protocols on different DTN hops. In DTN jargon the protocol stack below BP is called “convergence layer”, and thus we have “convergence layer adapters”, i.e. interfaces, for several protocols, such as TCP, UDP, LTP (Licklider Transmission Protocol), EPP (Encapsulation Packet Protocol) and SPP (Space Packet Protocol). In more detail, EPP, SPP, UDP essentially encapsulate bundles for the later transmission carried out the CCSDS-specified space link protocols, such as TM, TC, AOS, Proximity-1, and USLP. EPP, SPP, and UDP do not provide any reliability measure, which must be offered by other protocol layers. TCP is typically considered to co-work with BP over the terrestrial segment of the overall network, while in space is not considered a viable solution because of long delays and/or frequent losses due to fluctuations in the signal quality. LTP on the other hand is considered the reference protocol for providing reliable data transfer over bi-directional point-to-point links, as also addressed in this study.

LTP was first standardized by the Internet Research Task Force (IRTF) in [7], [8], and then by the Consultative Committee for Space Data Systems (CCSDS) in [9]. Enhancements of LTP for real missions are currently under study by CCSDS, while an extended variant has recently been proposed by some of the authors in [10].

The aim of this paper is to study LTP performance when applied on Near Earth optical links. The use of Free Space Optical (FSO) technology in space offers many advantages with respect to radio frequency, the most significant being the much higher transmission speed [11], [12]. For this reason, it has been investigated and tested by all major

<sup>1</sup> 0000-0000 © 2020 IEEE

Authors’ addresses: C. Caini and A. Zappacosta are with the Department of Electrical, Electronic and Information Engineering, Advanced Research Center on Electronic System, University of Bologna, Bologna 40126, Italy, E-mail: (carlo.caini@unibo.it,

antony.zappacosta@studio.unibo.it); T. de Cola and A. Shrestha are with the Institute of Communications and Navigation, German Aerospace Center (DLR), Weßling 82234, Germany, E-mail: (tomaso.decola@dlr.de, amita.shrestha@dlr.de). (Corresponding author: Carlo Caini.)

space agencies [13]-[16]; DLR in particular has gained a significant experience in the study of optical downlinks between LEO satellites (including “cubesats”) and ground stations [17]-[19]. This paper builds on this experience to evaluate LTP in this environment, whose importance has dramatically increased in recent years.

With respect to the ample literature on LTP performance [20]-[26], this paper uses a different channel model, based on erasure tracks derived from real measurements campaigns conducted by DLR. The optical link is modeled as an on/off channel, described by these erasure tracks: all packets sent will pass when the channel is on, otherwise they are dropped. This way, segment losses are not independent, as usually assumed in the literature, but highly correlated in time, as in real optical links. The paper investigates the impact of this correlation on LTP performance, leading to a series of original conclusions.

Our study starts with an in-depth revision of LTP basics, with specific focus on ARQ (Automatic Repeat Request) loss recovery of “red” LTP parts (Section II). It continues with an overview of FSO technology followed by the description of the channel model used in experiments (Section III). Then the testbed used is described, with details of the protocol implementations and tools used for the first time in this paper, all of which we have made available to other researchers as free software (Section IV). Numerical results follow, split into two parts: first LTP session duration analysis (Section V); then goodput and channel efficiency (Section VI). Our conclusions summarize the main findings of the study (Section VII).

## II. THE LTP PROTOCOL

### A. Overview

LTP has been designed to counteract the greatest challenges that affect IPN networks, such as long delays, link intermittencies, high loss rates, link asymmetry. To this end, it minimizes interaction (“chattiness”) between transmitting and receiving peers [7]-[9]. LTP can run on top of either UDP (in test beds, as in this paper) or CCSDS-based equivalent protocols (in real deployments). In this paper we assume BP over LTP, i.e. we use LTP as convergence layer of BP. LTP can offer both a reliable and an unreliable service, with red and green parts respectively.

Let us list the key features of LTP:

- No connection-establishment phase (by contrast to TCP 3-way handshake).
- Rate based transmission speed (the Tx rate is specified in “contacts” between DTN nodes, instead of being based on feedbacks as in TCP).
- Unidirectional data flow (the reverse channel is used only for acknowledgments), to cope with possible channel asymmetry.

- Bundles passed by BP are encapsulated in LTP “blocks”, to be transmitted by independent LTP “sessions”, possibly running in parallel to fill the BDP.
- A block could theoretically consist of both a red and a green part, but here only monochrome sessions (either red or green) are considered, as experience has proved that they are preferable to mixed color sessions [10].
- An LTP block is split into a number of LTP “segments”, each passed to UDP, or other CCSDS-based equivalent protocol.
- Unlike with TCP, LTP acknowledgments (“report segments”) are triggered only by red data segments flagged as “checkpoints”.

We will start with red sessions, as the rest of the paper focuses on them.

### B. Red sessions

As the ARQ protocol implemented by LTP red is relatively complex, we prefer to proceed incrementally, starting from the simplest case.

- 1) No losses (ideal channel)

In ideal channel conditions (Figure 1), when the transmission session starts all the segments of the LTP block are sent, the last being flagged as End of Red-Part (EORP), End Of Block (EOB) and Checkpoint (CP) [8]. The time necessary to send all segments is called block “radiation time” and it is usually much less than the RTT (it has been expanded in the figure to improve the clarity of the drawing).

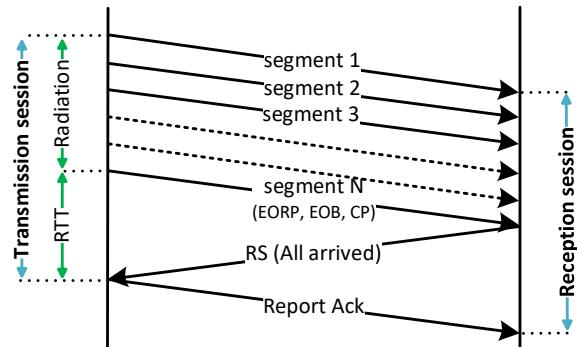


Figure 1: Example of a red LTP session in the absence of losses. The radiation time is usually much shorter than the RTT; it has been expanded for clarity. The RTT includes the two-way propagation delay and the actual processing time at both ends.

On arrival of the first segment, the receiver LTP peer opens the reception session and arriving data start to be buffered; the arrival of the last segment, flagged as CP, triggers a Report Segment (RS), which is a positive acknowledgment, confirming all data received, i.e. the content of the whole block in our case. As the block is complete, its payload (one or more bundles) is passed to BP

and the Rx buffer de-allocated. RS reception confirms the CP and is in turn confirmed by a Report-ACK (RA). As the full block is confirmed, the transmission session closes and the BP agent is notified of the successful delivery of the bundle(s) contained in the block. The bundle(s) can be canceled on the sender side, provided that no other constraints apply.

In interplanetary links, radiation and block processing times are negligible if compared to propagation delay. As a result, block delivery time is roughly equal to one-way propagation delay, ( $1/2$  RTT), and both transmission and reception session lifetimes are each equal to one RTT. This leads to the important conclusion that LTP is ideal in the absence of losses, because both delivery time and confirmed delivery time, are at their theoretical minima.

On near earth links, as those considered here, the propagation delay is orders of magnitude shorter (a few milliseconds instead of tens of minutes) and thus the validity of the above assumption must be verified, as done in the system model section. Let us just anticipate here that the high transmission speed offered by optical links helps limit radiation time.

2) Losses on data segments (“pure” and “mixed”)

We can now go on to consider losses on data segments, “pure” or flagged as CPs (Figure 2). It is essential to distinguish the former (segments 2 and 3 in the figure) from the latter (segment N) as their impact is different. At the sender side, all the segments of the block are sent as before; at reception, however, segments 2 and 3 do not arrive, causing a first gap in the Rx buffer which will need to be filled with retransmission. The same holds true for the missed reception of the last segment, N. However, this segment is also flagged as a CP and its loss has greater consequences, as it prevents the reception peer from sending the RS that would inform the sender of the need to retransmit the missing data. The stall situation is resolved after one RTO (Retransmission Time Out) by sending a copy of segment N, flagged as before. This is triggered by a retransmission timer that fires if the CP is not confirmed by an RS in the due time; obviously the RTO must be significantly greater than one RTT, exactly as in TCP and in many other ARQ protocols.

Moving back to our session, the arrival of segment N triggers an RS which will confirm all data but those contained in segments 2 and 3; more precisely, as we have a gap within the block, we will have two “claims”, one confirming data before the gap and one after the gap. Note that if we had had more than  $N_{max}$  claims, two (or more) RSs would have been sent. The arrival of the RS is immediately confirmed by a Report Ack, followed by the missing segments, 2 and 3, the latter flagged as CP (but not EORP and EOB). This time all segments arrive and everything continues as in the ideal case. Transmission and

reception sessions both increase by the time necessary to perform one retransmission cycle, which we will call Re-Tx cycle penalization. In interplanetary links, the radiation time of retransmitted segments is negligible and Re-Tx time penalization becomes equal to RTT which in turn is the same as twice the propagation delay, as both radiation time and processing delays can be neglected. In near-earth, the Re-Tx time is only roughly the same as RTT (including processing times); the accuracy of the approximation depends on many factors, especially the Tx speed. A few considerations are in order:

- All unacknowledged data segments but CPs are retransmitted in one retransmission cycle, which greatly differentiate LTP from TCP. Once segment 2 is lost, the additional loss of segment 3 adds a negligible increment.
- The loss of the last segment, flagged as CP, is much worse than the loss of previous segments as it adds one RTO to the session duration, independently of other losses, as shown in Figure 2.
- As RTO and Re-Tx penalizations are additive, in the unlucky case of consecutive losses these delays may be added on many times, as we will see later. Every lost CP adds one RTO, while consecutive losses on pure Re-Tx data segments require extra retransmission cycles, with all possible combinations.

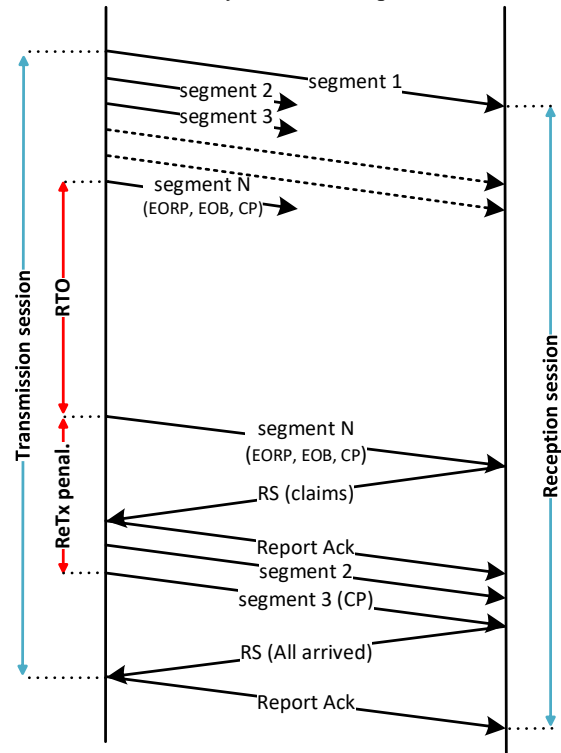


Figure 2: Example of red LTP session in presence of losses on pure data segments (2 and 3) and on checkpoints (segment N, flagged as CP). The session time increase is one RTO, for CP loss, and one Re-Tx cycle (about one RTT). Note that Re-Tx cycle penalization is basically independent of number of losses.

3) Losses on data and other signaling segments.

All LTP signalling segments, except acknowledgments, are protected by RTO timer. This means that not only the loss of CPs but also of RSs increases session time by one RTO. Moreover, if the maximum number of allowed retransmissions of the same CP or the same RS is exceeded, the session is cancelled by sender or receiver side and the opposite peer is informed by CS (Cancel segment from block Sender) and CR (Cancel segment from block Receiver) segments respectively. Even these segments are subjected to retransmissions, unless acknowledged by CAS (Cancel-Acknowledgment to block Sender) and CAR (Cancel-Acknowledgment to block Receiver), respectively. If the maximum number of retransmissions of CS or CR is exceeded, the session is eventually unilaterally closed. The interested reader is referred to RFC 5326 for a comprehensive treatment [8].

### C. Green sessions

Monochrome green sessions are very simple and require just a few words. The block is sent to the other peer segment-by-segment, as usual. After sending the last LTP segment, the BP on the sender side is notified by the local LTP engine of session “success”, which in this case does not mean successful reception of the full block on the other side, but only that all segments have been radiated to the other side, a great difference indeed. As green sessions are unreliable, there are no feedbacks; an advantage is that they can be used on unidirectional links.

## III. FREE-SPACE OPTICAL (FSO) COMMUNICATIONS

Free-Space Optical communications use light to transmit data through air, vacuum, or any other free propagation medium, in contrast to Optical Communications that use guided propagation, e.g. through fibers. For short-distance low data rate links LED are normally used, while for long distance links (satellite to ground, aircraft to ground etc.) infrared lasers and photo-detectors are preferred, with telescopes acting as optical antennas.

### A. Applications

A laser link can be very effective way to transmit data from point to point at high Tx speeds with low transmit power. In terrestrial applications their widespread use is limited by the terrestrial atmosphere, in particular rain, fog, dust and heat, thus guided optical communications with fibers are preferred when feasible. In space wired solutions are impossible, and FSO technology is a promising alternative to traditional RF (Radio Frequency) techniques, not only for the much higher transmission speed that FSO can offer, but also for the lower power required and the

smaller dimensions of optical devices (telescopes present in optical transmitters and receivers can be smaller than antennas in their RF equivalent).

#### 1) Space-to-space links

When both end-points are in space (i.e. on space-to-space links), atmosphere effects do not exist, which makes FSO very suited to inter-satellite communications. This technology was pioneered by many space agencies, including NASA, JAXA, ESA and DLR; for example, the EDRS (European Data Relay System) project by ESA, involving an optical link between one GEO and other LEO satellites at 1.8 Gbit/s [13]. At present, Space-X and other mega constellation satellites use or are going to use this technology, with speeds that could reach 100 Gbit/s. The use of FSO application on interplanetary links, i.e. between a spacecraft orbiting around the Earth and another around Mars, seem quite interesting too, as it could offer transmission speeds orders of magnitude higher than their RF counterparts.

#### 2) Earth-to-space and Space-to-Earth links

There are also, and of primary interest here, point-to-point links where one end-point is on Earth and the other in space (Earth-to-space links and vice versa). In this case, rain and clouds can be counteracted by means of space diversity granted by the use of multiple ground-stations, as the correlation of cloud coverage between two locations usually decreases for distances larger than 80 kilometers [11]. On the other hand, other possible impairments might be more severe, such as errors in pointing of telescopes, as the laser beam is very narrow and the distances are much longer than in terrestrial applications.

Several experiments have been carried out, either between Moon and Earth or satellite and Earth, most notably including the Lunar Laser Communication Demonstration (LLCD), considering an optical downlink from a device on the Moon orbit, at 600 Mbit/s, in 2013-14 [15]; the Laser Communications Relay Demonstration (LCRD) by NASA, studying a link from a GEO satellite to Earth [16]; DLR experimental deployments involving small LEO sats and Earth, such as CubeLCT, OSIRISv1 and v3 [17], OSIRIS4CubeSat [18], with FSO downlink data rates from 100 Mbit/s to 10 Gbit/s.

### B. Channel emulation

In this paper we assess LTP performance on near Earth FSO, i.e. between a LEO satellite and a ground station. In addition to atmospheric effects due to fog, clouds, rain etc., which can be counteracted by means of space diversity, the biggest challenges for this scenario are atmospheric turbulence and vibrations that can result in time-correlated fading. In order to evaluate their impact on higher layer protocols, like LTP, it is very important to use a suitable model for the optical channel. This can be achieved either

by starting from theoretical FSO channel models, or, for a more practical approach, from samples of received instantaneous power, measured on real links. Unfortunately, a large measurement database is not available, so the Optical Satellite Link department (OSL) of the institute of Communication and Navigation at DLR (DLR-KN) has developed the Power Vector Generator Tool (PVGet) [27], which can derive artificial power vectors from real satellite link measurements collected by DLR in the course of last decade.

For this particular study, the channel will be emulated as an on/off channel starting from power vectors and BER (Bit Error Rate) tracks provided by the OSL department of DLR-KN.

#### 1) Power Vectors and BER tracks

BER tracks describe the variable BER that can be expected on a link before the application of any FEC (Forward Error Correcting codes) at physical layer, with a time granularity of 0.1 ms, i.e. with a sampling frequency of 10 kHz. These tracks were obtained by simulation, using power vectors and receiver model [28]. Power vectors and BER tracks [28] depend on several parameters like telescope aperture diameter, beam divergence, pointing jitter, atmospheric scintillation strength, wind orthogonal velocity and platform movements.

In particular, the power vectors used in this work describe the state of the channel during a contact between a LEO satellite and an optical ground station. Accordingly, another parameter affecting BER is the satellite elevation angle with respect to the horizontal plane at the ground station. The higher the elevation angle, the closer the satellite is to the Zenith resulting in better propagation conditions as the signal must cross less of the Earth's atmosphere.

#### 2) Erasure traces, scenarios A, F and H

From power vectors and BER tracks it is possible to derive erasure vectors describing the time-varying and time-correlated on/off state of the channel, by considering many parameters, including the use of FEC at physical layer and the expected bit rate. We eventually obtained three traces of 100s each with a time granularity of 0.1 ms (corresponding to 100,000 samples), describing three scenarios with increasing impairments. These traces do not pretend to be representative of average conditions and are all particularly severe. Their characteristics are summarized in Table I.

Trace A provides 4.9% of entries at '1', meaning that all packets (more precisely, link 2 frames) sent in the corresponding time interval are lost; equivalently, we can say that the channel is in the bad state for 4.9% of the time. The average fading duration (or length of the bad state) is 0.95 ms, the standard deviation 0.66 ms. On the other hand,

in erasure Trace A we have 95.1% of 0, which means that the channel will be in the good state for the same fraction of time; the average duration of the good state is 18.17 ms and the std. deviation 17.37 ms. Moving to other traces, data in the table show that with 13.7% of erasures trace F is worse than A, while H is the worst, with a really extreme value of 28.8 % erasure rate.

TABLE I. TRACE CHARACTERISTICS

Trace	Bad state %	Bad state mean (ms)	Good state mean (ms)	Bad state dev.std. (ms)	Good state dev.std. (ms)
A	4.98	0.95	18.17	0.66	17.37
F	13.7	2.14	13.46	3.15	18.19
H	28.8	5.46	13.48	5.02	12.84

Both the erasure rate and its time correlation increase from A to H, as we can see by comparing the bad state length CDFs (Cumulative Distribution Functions) in Figure 3. Median values rapidly increase from Trace A to H (0.8 ms, 1.2 ms and 4.1 ms respectively) and the same holds true for 90th percentiles, which are significantly higher (1.8 ms, 4.4 ms and 11.6 ms).

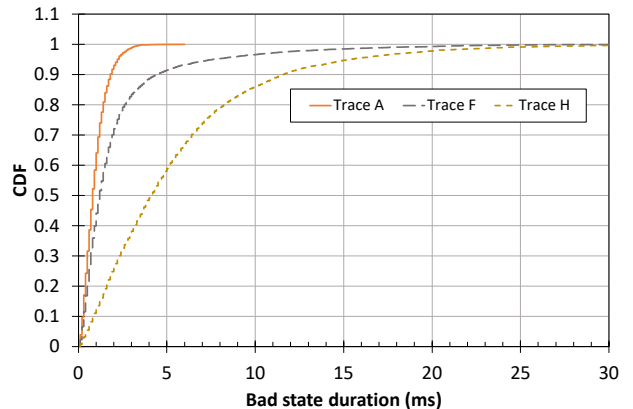


Figure 3 Cumulative distribution function of bad-state duration for Traces A, F and H.

## IV. TESTBED CONFIGURATION

To analyze LTP performance we implemented a minimal testbed consisting of three real machines: one DTN source, one DTN destination and an intermediate non-DTN node acting as channel emulator. The use of real machines was dictated by the need to achieve relatively high data rates. Bundle traffic was generated by means of the DTNperf tool [30], [31]. The corresponding protocol stack is presented in Figure 4.

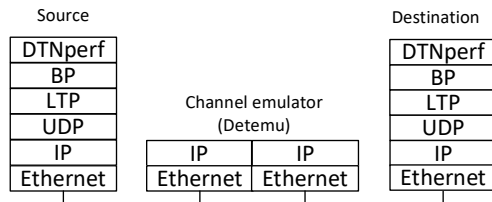


Figure 4: Protocol stack of the 3 testbed machines. Bundles generated by DTNperf client are sent to DTNperf server via the channel emulator node, where incoming frames are dropped if channel state is off at frames' transit time.

### A. Hardware and DTN software

The three machines are normal off-the-shelf desktop PCs, with Debian 11 GNU/Linux distribution. The experiment environment is “clean”, i.e. isolated from any other traffic, thanks to the use of additional dedicated Ethernet NIC cards. We used Unibo-BP as BP implementation [32] and Unibo-LTP for LTP. Both were recently released as free software under the GPLv3 license and can be freely downloaded from [33], [34].

Preliminary tests [35] were carried out by means of a similar testbed at DLR premises, running either ION [36], [37] or DTNME [38] for both BP and LTP. We preferred to use Unibo implementations as first we needed high Tx rates, as power vectors made available by DLR refer to bit rates ranging from 500 Mbit/s up to 1 Gbit/s. As ION LTP cannot go faster than about 100 Mbit/s, it was discarded after the very first tests run at DLR. Then we tried DTNME, with satisfactory results concerning speed, but with a significant limitation in RTO granularity, whose importance will be discussed later.

In this regard, is worth noting that ION and DTNME follow two different methods to set the LTP RTO timer. The former computes it as the sum of the nominal delays from A to B and from B to A, declared in range instructions, plus the processing delays at the LTP sender and receiver. These values are all expressed in seconds and input as integers, thus in ION the RTO minimal value greater than 0 is 1 s. DTNME follows a less sophisticated rule, as RTO is directly input as an integer, but it has the same coarse granularity of 1 s. However, we discovered that by setting 0 s, we actually obtained 0.1 s, a value much closer to the actual RTT of LEO satellites, but still too large in our case, as shown later. The impossibility of further reducing the RTO led us to abandon DTNME and develop a modified version of Unibo-LTP that allows the user to insert the RTO in ms directly, as an alternative to the ION-style indirect setting. This option is included in the latest official version.

A third important point in favor of Unibo-LTP was the possibility of tailoring its logs to our needs: we made Unibo-LTP produce a .csv (comma separated values) file where each line, i.e. a record, corresponds to a received or sent LTP segment, with all useful data (timestamp, session

number, session originator, LTP type, etc.), reported. These data are then elaborated by the “LTP performance analyzer” as shown later.

### B. Bundle flow

On the source node DTNperf client generates bundles of 500 kB destined to DTNperf server.

#### 1) Bundle generation (DTNperf client)

The client works in window mode [30], i.e. it generates first  $W$  bundles and then it awaits the arrival of one DTNperf server confirmation (DTNperf ACK) before generating a new bundle and so on, in a way similar to TCP with the important difference that here  $W$ , the equivalent of TCP congestion window, is fixed (it is an input parameter of the DTNperf client). In this way we are sure that after the first  $W$  bundle, the source generates bundles exactly at the pace sustained by the channel, which is obviously variable.

Generated bundles are saved in a local database by BP, as prescribed by BP specifications, waiting to be passed to LTP. Although there is not any bundle routing problem, as we have a point-to-point layout, it is necessary to pass both range and contact instructions to Unibo-BP, as the CGR/SABR routing protocol is invoked in any case [39], [40]. Contacts and ranges are also used by LTP.

#### 2) From bundles to LTP segments

The actual passage of a bundle to LTP is allowed only when one of the LTP Tx buffers is free, as LTP limits the maximum number of parallel sessions (another parameter in input to LTP). The bundle is directly encapsulated into one LTP block (no bundle aggregation is performed), then the block is divided into segments of 1024 bytes. These segments are passed to UDP at a steady pace, to avoid a long burst which could result into buffer losses. Unibo-LTP uses a token bucket pacer, whose token rate depends on the nominal bit rate declared in contacts (500 Mbit/s in the tests). A part from the token bucket, we significantly enlarged UDP buffers to avoid internal losses, i.e. losses caused not by the channel but by UDP buffer overflows; we carefully checked this critical point, with a few experiments on an ideal channel where we verified that we had no losses at all.

#### 3) The channel emulator

UDP datagrams are encapsulated into IP packets which must travel through the channel emulator node via Ethernet. Losses on this flow, and more specifically on incoming Ethernet frames, are induced by “Detemu”, our specifically designed tool, according to the erasure traces received in input [41]. Although conceptually simple, this operation is complicated by the need to process a huge quantity of frames without introducing any significant delay; to this end Detemu uses the very fast PcapPlusPlus



C++ library [42]. Note that Detemu operates only in the forwarding direction, as in the reverse direction the channel is assumed to be ideal. An exhaustive description of Detemu would be beyond our scope here but the interested reader can find all details in [35]. To complete the emulation of the LEO channel, a delay of 10 ms is added by means of the Linux command “tc-netem” [43].

#### 4) From LTP segments to Bundles

The LTP receiver collects LTP segments belonging to the same session. When a CP arrives, if the block is completed it is passed to BP, and an RS confirming complete reception of the full block is sent to the LTP sender. Otherwise, the previously described recovery mechanism of LTP red is applied.

#### 5) Bundle reception and confirmation (DTNperf server)

Bundles are then passed to the DTNperf server, which acknowledges them to the client with a DTNperf ACK. This very short bundle is sent back in a green LTP session, as the channel in the opposite direction is ideal. In this way we also greatly reduce processing time (the green session containing the DTNperf ACK consists of a sole LTP segment) and we avoid any interference with the forward traffic.

### C. Summary of test characteristics and analysis of results

#### 1) Test characteristics

All tests consider a continuous bundle transfer with bundles of 500 kB each; the latest version of the BP, BPv7, is used [6]; data bundles are sent to destination via LTP red while DTNperf ACKs via LTP green. The duration of the transfer is equal to the duration of the Erasure tracks, i.e. 100 s. During this period, we have only one contact with a nominal Tx rate of 500 Mbit/s. This speed represents an upper limit on the achievable goodput and it also determines the bundle radiation time (8 ms with bundles of 500 kB).

#### 2) Analysis of results

Each test produces one .csv Unibo-LTP log, with one row per transmitted or received LTP segment. As each test encompasses several thousand sessions of about 500 LTP segments each, the exact number depending on the channel characteristics, these log files are far too large to be manually inspected or processed in a spreadsheet and thus are elaborated by a dedicated program, the “LTP performance analyzer” [35] [44]. After performing a huge number of calculations, this produces a second, much lighter .csv file, this time consisting of only one row per LTP session; each row contains the desired per-session statistics, such as Tx session duration, number of RTOs, number of retransmission cycles, etc. This file is later

imported to a spreadsheet to calculate the average values presented in the next two sections. The main advantage of using a spreadsheet in this last step is greater flexibility and control of session results (possible anomalies on specific sessions can be easily identified by visual inspection).

## V. ANALYSIS OF TX-SESSION DURATION

A thorough assessment of LTP goodput and channel utilization requires a preliminary study of the factors that influence the LTP session duration in the presence of correlated losses.

### A. RTO impact

We will start our analysis by stressing the impact of RTO settings. For the sake of brevity, we will limit the analysis to Trace A, having fixed the maximum number of permitted parallel sessions to 7. Figure 5 shows data averaged over all the sessions completed in the 100s covered by Trace A, namely, 5862 for RTO=1 s, 10932 for RTO=100 ms and 11210 for RTO=30 ms. Each bar represents the length of each component, specifically the average length of a session without losses (“Ideal”), the estimated penalization due to retransmissions cycles (calculated as average number of Re-Tx cycles per expected duration of each retransmission cycle, about 15 s), the estimated penalization due to CP losses (average number of CPs lost per RTO duration), the error on estimating these two penalization times (DeltaPen), and, lastly, the error on estimating the ideal session length (DeltaId). As the last two components are low, we can be confident of the relative accuracy of previous estimates.

From the first bar, we can see that with RTO=1 s, the penalization due to lost CPs (EsPenCP, 74 ms) dominates all other factors, as it is 3.34 times as long as the ideal session time (22 ms, given by about 8 ms of radiation time, plus 10 ms of added two-way delay on the channel, plus about 4 ms ascribable to processing delays). This clearly shows that with RTTs consisting of only a few tens of milliseconds, as in LEO to Earth communications, the one second granularity is too large as it prevents proper setting of the RTO. Results improve if we examine the second bar, referring to an RTO of 100 ms, which was the case considered in preliminary tests [35] (this RTO is the lowest achievable by DTNME). The last bar, referring to an RTO of 30 ms, has the best performance, with a reduction of the average length to 33 ms, i.e. only 50% more than the average length of an ideal session (22 ms). This bar also shows that there is no point in further reducing the RTO, as now the most important penalization is due retransmission cycles. Note, as further confirmation, that 30 ms is only moderately longer than the actual RTT, which can be estimated at about 15 ms (radiation time of retransmitted

segments, in the range from 0 to 8 ms, plus 10 ms two-way delay added by the channel, plus processing time). From now on, all other results will refer to this 30 s RTO value.

If we had considered traces F and H, we would have obtained the same qualitative results, but with an even greater dominance of lost CP penalizations, which would have given even greater emphasis to the need of a proper RTO setting.

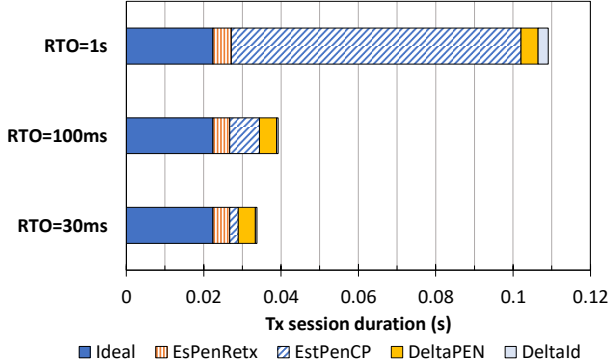


Figure 5: Average components of Tx session duration, trace A (number of permitted parallel sessions=7). With RTO=1s the average penalization due to CP losses (EsPenCP, 74 ms) dominates all other penalization factors, leading to a total duration of 135 ms. By reducing RTO to 30 ms, the total session time is reduced to 33 ms, only 50% longer than the ideal session (22 ms).

### B. Correlated vs. independent identically distributed losses

By reducing the impact of RTO, the time spent in retransmitting lost segments has become the most prominent penalization factor. We are thus in the right position to assess the impact of loss correlation on this penalization time. To this end, we have compared the average Tx session duration obtained with original traces A, F and H, with that achieved on a channel introducing exactly the same rate of independent losses. This is achieved by setting as PER the corresponding percentage of bad state given in the first column of Table I, and asking the channel emulator “Detemu” to produce Independent and Identically Distributed (IID) losses, disregarding erasure traces. The results presented in Figure 6 show that correlated losses on LTP segments (left bars) are better than independent losses (right bars). This result, which may appear counterintuitive, actually depends on the LTP ability to recover multiple segment losses in only one re-transmission cycle, so it is better to have losses concentrated than spread in time.

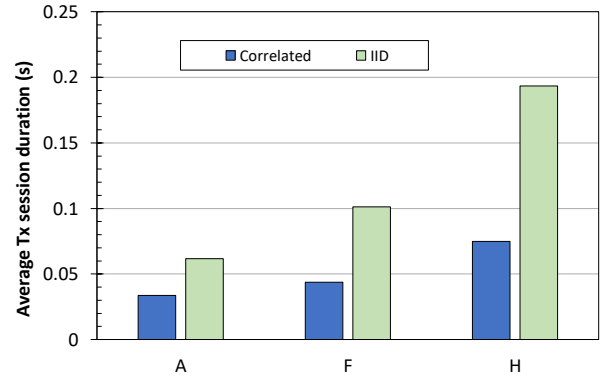


Figure 6: Correlated vs. uncorrelated losses: average Tx session duration (number of permitted parallel sessions=7, RTO=30 ms). The comparison shows that LTP can recover correlated losses faster than IID losses.

A more detailed analysis shows that there are two factors leading to this result. First, if losses are concentrated, the number of sessions with errors, i.e. requiring at least one retransmission cycle, is a fraction of the total (less than 40 % for Trace A, which is to say that more than 60 % of sessions are ideal), as shown in Figure 7 (left bars), while when PER is high and losses are independent (right bars) all sessions require at least one re-transmission cycle (no ideal sessions at all).

The second factor is that even when a session is not ideal, the number of consecutive losses, requiring further re-transmission cycles, is lower (data not shown).

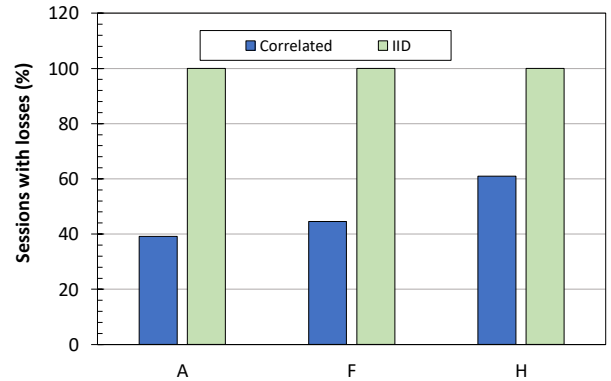


Figure 7: Correlated vs. uncorrelated losses: number of sessions with losses (number of permitted parallel sessions=7, RTO=30 ms). The comparison shows that with IID losses all sessions are affected by losses and thus require at least one Re-Tx cycle.

The two factors, considered together, lead to an average number of re-transmission cycles which is definitely lower in the case of correlated losses, as shown in Figure 8. This figure clearly demonstrates the outstanding capacity of LTP in dealing with multiple correlated losses, which makes it an excellent candidate for FSO channels, even when RTT is short, as in the Leo-to-Earth links considered here.



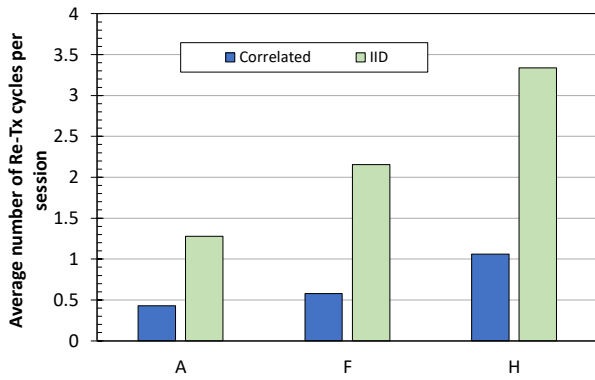


Figure 8: Correlated vs. uncorrelated losses: average number of Re-Tx cycles per session (number of permitted parallel sessions=7, RTO=30 ms). The comparison shows that with IID losses the average number of Re-Tx cycles is much higher, thus leading to longer sessions.

### C. Variability with the number of permitted sessions

As the previous results refer to a maximum number of 7 parallel sessions, one might wonder what influence this parameter has on session duration, so we carried out a number of tests varying this value. Results are shown in Figure 9, where for clarity we have linked markers with curves. Staring from the bottom curve, which refers to the ideal case (no losses), we can see that the average session duration for parallelism 7, found in previous tests (about 22 ms), is actually the same for all values greater than 2, while for 1 and 2 it is marginally longer. The same trend is shown by results for Trace A, but the increase at very low values (1, and 2) is more pronounced. The same holds true for traces F and H, although the latter also shows a mild non-monotonic behavior, which could be ascribed to interference between retransmission cycles (particularly frequent and long in this trace) and new blocks.

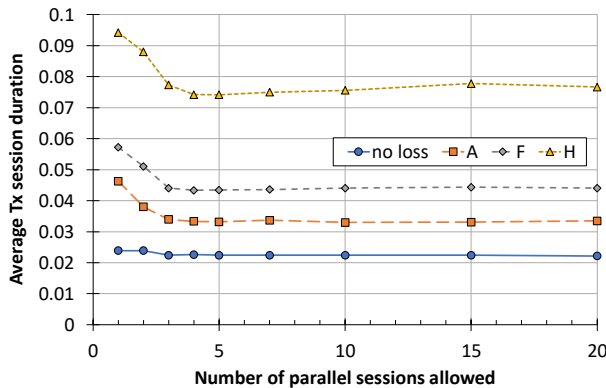


Figure 9: Average LTP Tx session duration vs. the number of permitted parallel sessions (RTO=30 ms). The session duration is approximately constant when the number of sessions allowed exceeds 3.

The reasons for the longer average length for a parallelism level less than 3, shown by all curves, could be ascribed to LTP design or to other factors (Unibo-LTP implementation, operating system scheduler, etc.), but are difficult to investigate and at present unknown. Here it is

enough to say that session length proved roughly independent to parallelism for values greater than 3, which extends the validity of results presented in the previous section from parallelism 7, for which they were originally obtained, to all values greater than 3.

## VI. GOODPUT AND CHANNEL EFFICIENCY

Having analyzed the different factors that influence average Tx session duration, we are ready to study performance in terms of goodput and channel utilization.

### A. Goodput

If only one session at a time were possible, the maximum achievable throughput on an ideal channel would be one block per RTT plus block radiation time. This is why even in these ideal conditions it would be necessary to allow for parallel sessions in order to “fill the available bandwidth”; a fortiori, a greater level of parallelism is obviously required with losses, as session duration is increased by timeouts and retransmission cycles, as shown in the previous section.

To quantitatively assess goodput really achievable on our system, we carried out a series of tests by increasing the level of parallelism from 1 to 20 (Figure 10). Starting from the ideal curve (no losses) we can observe that by moving from one to 3 parallel sessions goodput saturates at about 470-480 Mbit/s. With trace A a higher level of parallelism, seven, is required to reach saturation because of losses, as expected. Moreover, the saturation level is lower, corresponding to about 450 Mbit/s. With traces F and H the trend is the same. Trace F requires at least 10 parallel sessions to saturate at about 405 Mbit/s, while Trace H requires at least 15 to saturate at 350 Mbit/s.

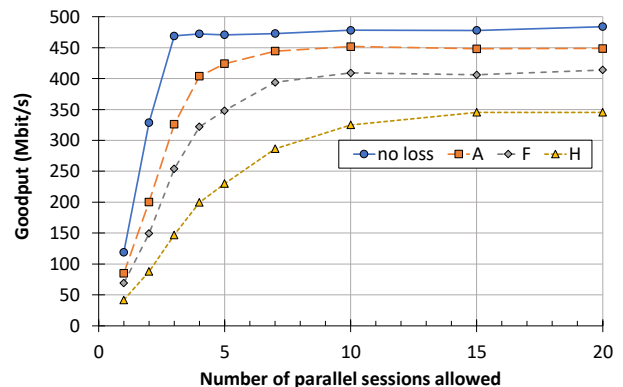


Figure 10: Goodput at application layer vs. number of permitted parallel LTP sessions (RTO=30ms). Moving from the ideal channel (no losses) to the most challenging trace (H), a higher level of parallelism is required to reach saturation point.

### B. Channel utilization efficiency

Goodput results described so far hide the fact that the channel is actually available, i.e. on the good state, for only

a fraction of the time. It is therefore interesting to plot channel utilization efficiency, i.e. the goodput normalized to the actual average bandwidth available, given by the Tx nominal speed (500 Mbit/s) for the percentage of the good state (100% with no losses, 95% with trace A, 86% trace F, 71% trace H). Results plotted in Figure 11 show that, if a fair level of parallelism is provided, LTP is able to exploit the available channel at over 90 % efficiency even in the most challenging case of Trace H. This exceptional result once again proves LTP's ability to deal successfully with severe losses, in the presence of low RTTs.

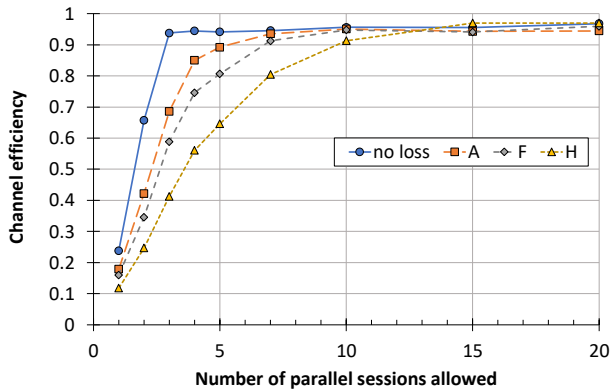


Figure 11: Channel utilization efficiency, i.e. goodput at application layer normalized to actual availability rate of channel. Provided that an adequate level of parallelism is allowed, LTP is always able to fully exploit available bandwidth when the channel is in good state (efficiency saturates at over 90 %).

## VII. CONCLUSIONS

The aim of the paper was to assess performance achievable by LTP when coupled with optical links, in LEO to Earth environments characterized by very low RTTs, high PER and correlated losses. We first reviewed the basics of LTP red recovery, to highlight the factors that increase session duration: retransmission timeouts and retransmission cycles. Their quantitative impact was then studied on a real testbed with an emulated channel based on erasure traces obtained by power vectors provided by DLR. The first part of the numerical result analysis led to three significant preliminary conclusions: 1) importance of the proper setting of RTO; 2) LTP ability to recover multiple losses in one Re-Tx cycle, hence making independent segment losses more challenging than correlated losses; 3) session duration largely independent of the maximum number of sessions allowed. These preliminary considerations are instrumental to understanding the goodput and channel efficiency results presented at the end of the paper. In particular the latter shows that with a proper level of parallelism, LTP is always able to exploit the bandwidth available, independently of poor channel conditions. This outstanding result proves that LTP is a perfect match for optical links in near Earth environments. Future directions of the work may include

the development of an analytical framework to get further insights about LTP performance and/or the extension of this study to trans-lunar and deep-space links.

## ACKNOWLEDGEMENTS

The authors would like to thank all the students who contributed to the development of the DTN protocols and tools used in this paper and in particular Lorenzo Persampieri (for Unibo-BP) and Andrea Bisacchi (for the original version of Unibo-LTP).

## REFERENCES

- [1] S. Burleigh *et al.*, "Delay-tolerant networking: an approach to interplanetary Internet," in *IEEE Commun. Mag.*, vol. 41, no. 6, pp. 128-136, June 2003, doi: 10.1109/MCOM.2003.1204759.
- [2] V. Cerf, A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall, H. Weiss "Delay-Tolerant Networking Architecture," Internet RFC 4838, Apr. 2007.
- [3] CCSDS 734.0-G-1, "Rationale, Scenarios, and Requirements for DTN in Space," 2010, <https://public.ccsds.org/Pubs/734x0g1e1.pdf>.
- [4] K. Scott, S. Burleigh, "Bundle Protocol Specification," Internet RFC 5050, Nov. 2007.
- [5] CCSDS 734.2-B-1, "Bundle Protocol Specification," 2015. <https://public.ccsds.org/Pubs/734x2b1.pdf>
- [6] S. Burleigh, K. Fall, E. Birrane, "Bundle Protocol Version 7", Internet RFC 9171, Jan. 2022.
- [7] M. Ramadas, S. Burleigh and S. Farrell, "Licklider Transmission Protocol – Motivation," Internet RFC 5325, Sept. 2008.
- [8] M. Ramadas, S. Burleigh and S. Farrell, "Licklider Transmission Protocol – Specification," Internet RFC 5326, Sept. 2008.
- [9] CCSDS 734.1-B-1, "Licklider Transmission Protocol (LTP) for CCSDS". 2015. <https://public.ccsds.org/Pubs/734x1b1.pdf>.
- [10] A. Bisacchi, C. Caini and T. de Cola, "Multicolor Licklider Transmission Protocol: An LTP Version for Future Interplanetary Links," in *IEEE Trans. on Aerosp. and Electron. Syst.*, vol. 58, no. 5, pp. 3859-3869, Oct. 2022, doi: 10.1109/TAES.2022.3176847.
- [11] A. Carrasco-Casado and R. Mata-Calvo, "Free-space optical link for space communication networks," in *Springer Handbook of Optical Networks*, Springer 2020, doi: [https://doi.org/10.1007/978-3-030-16250-4\\_34](https://doi.org/10.1007/978-3-030-16250-4_34)
- [12] H. Kaushal and G. Kaddoum, "Optical Communication in Space: Challenges and Mitigation Techniques," in *IEEE Commun. Surv. & Tutor.*, vol. 19, no. 1, pp. 57-96, First quarter 2017, doi: 10.1109/COMST.2016.2603518.
- [13] H. Hauschildt, S. Mezzasoma, H. L. Moeller, M. Witting and J. Herrmann, "European data relay system goes global," in *Proc. IEEE ICSOS*, Naha, Japan, 2017, pp. 15-18, doi: 10.1109/ICSOS.2017.8357204.
- [14] H. Hauschildt *et al.*, "HydRON: High thRoughput Optical Network," in *Proc IEEE ICSOS 2019*, Portland, OR, USA, 2019, pp. 1-6, doi: 10.1109/ICSOS45490.2019.8978985.
- [15] D. M. Boroson, J. J. Scozzafava, D. V. Murphy, B. S. Robinson and M. I. T. Lincoln, "The Lunar Laser Communications Demonstration (LLCD)," in *Proc. IEEE SMC-IT 2009*, Pasadena, CA, USA, 2009, pp. 23-28, doi: 10.1109/SMC-IT.2009.57.
- [16] D. J. Israel, B. L. Edwards and J. W. Staren, "Laser Communications Relay Demonstration (LCRD) update and the path towards optical relay operations," in *Proc. IEEE Aerospace Conf. 2017*, Big Sky, MT, USA, 2017, pp. 1-6, doi: 10.1109/AERO.2017.7943819.
- [17] C. Schmidt and C. Fuchs, "The OSIRIS program — First results and Outlook," in *Proc. IEEE ICSOS 2017*, Naha, Japan, 2017, pp. 19-22, doi: 10.1109/ICSOS.2017.8357205.
- [18] B. Rödiger, C. Fuchs, J. R. Nonay, W. Jung and C. Schmidt, "Miniaturized Optical Intersatellite Communication Terminal – CubeISL," in *Proc IEEE ICC Works. 2021*, Montreal, QC, Canada, 2021, pp. 1-5, doi: 10.1109/ICCWorkshops50388.2021.9473747.
- [19] Deutsches Zentrum für Luft- und Raumfahrt, "OSIRIS4CubeSat," Available: <https://www.dlr.de/kn/en/desktopdefault.aspx/tabid-17840/#gallery/36173>.

- [20] L. Clare and G. Miles, "Deep space optical link ARQ performance analysis," in *Proc. IEEE Aerospace Conf. 2016*, Big Sky, MT, USA, 2016, pp. 1-11, doi: 10.1109/AERO.2016.7500874.
- [21] Q. Yu, S. C. Burleigh, R. Wang and K. Zhao, "Performance modeling of Licklider Transmission Protocol (LTP) in deep-space communication," in *IEEE Trans. on Aerospace and Electron. Sys.*, vol. 51, no. 3, pp. 1609-1620, July 2015, doi: 10.1109/TAES.2014.130763.
- [22] G. Yang, R. Wang, S. C. Burleigh and K. Zhao, "Analysis of Licklider Transmission Protocol for Reliable File Delivery in Space Vehicle Communications With Random Link Interruptions," in *IEEE Trans. on Vehicular Tech.*, vol. 68, no. 4, pp. 3919-3932, April 2019, doi: 10.1109/TVT.2019.2897305.
- [23] L. Yang, R. Wang, Y. Zhou, J. Liang, K. Zhao and S. C. Burleigh, "An Analytical Framework for Disruption of Licklider Transmission Protocol in Mars Communications," in *IEEE Trans. on Vehicular Tech.*, vol. 71, no. 5, pp. 5430-5444, May 2022, doi: 10.1109/TVT.2022.3153959.
- [24] L. Clare and G. Miles, "Deep space optical link ARQ performance analysis," in *Proc. IEEE Aerospace Conf. 2016*, Big Sky, MT, USA, 2016, pp. 1-11, doi: 10.1109/AERO.2016.7500874.
- [25] R. Lent, "Analysis of Bundle Throughput Over LTP," in *Proc. IEEE LCN 2018*, Chicago, IL, USA, 2018, pp. 271-274, doi: 10.1109/LCN.2018.8638093.
- [26] N. Alessi, S. Burleigh, C. Caini, T. De Cola, "Design and Performance Evaluation of LTP Enhancements for Lossy Space Channels", *Int J Satell Commun Network*, Vol.37, No.1, pp.3-14, Jan-Feb 2019, DOI: 10.1002/sat.1238
- [27] D. Giggenbach, S. Parthasarathy, A. Shrestha, F. Moll and R. Mata-Calvo, "Power vector generation tool for free-space optical links — PVGeT," in *Proc. IEEE ICSOS 2017*, Naha, Japan, 2017, pp. 160-165, doi: 10.1109/ICSOS.2017.8357228.
- [28] D. Giggenbach and R. Mata-Calvo, "Sensitivity modeling of binary optical receivers," *Appl. Opt.* 54, 8254-8259, 2015. doi: 10.1364/AO.54.008254
- [29] D. Giggenbach and A. Shrestha, Atmospheric absorption and scattering impact on optical satellite-ground links. *Int J Satell Commun Network*. 2022; 40(2): 157-176. doi:10.1002/sat.1426
- [30] C. Caini, A. d'Amico and M. Rodolfi, "DTNperf\_3: A further enhanced tool for Delay-/Disruption- Tolerant Networking Performance evaluation," in *Proc. IEEE GLOBECOM 2013*, Atlanta, GA, US, 2013, pp. 3009-3015. doi: 10.1109/GLOCOM.2013.6831533
- [31] DTNsuite code (including DTNperf). Available: <https://gitlab.com/unibo-dtnsuite>
- [32] Lorenzo Persampieri, "Unibo-BP: an innovative free software implementation of Bundle Protocol Version 7 (RFC 9171)", Master's thesis, University of Bologna, 2023. Available: <https://amslaurea.unibo.it/27740/>
- [33] Unibo-BP code. Available: <https://gitlab.com/unibo-dtn/bp>
- [34] Unibo-LTP code. Available: <https://gitlab.com/unibo-dtn/ltp>
- [35] A. Zappacosta, "Performance evaluation of Licklider Transmission Protocol over Free Space Optical communication testbeds", Master's thesis, University of Bologna, March 2023. Available: <http://amslaurea.unibo.it/view/cds/CDS0937/>
- [36] S. Burleigh, "Interplanetary Overlay Network (ION) an Implementation of the DTN Bundle Protocol, Proc. IEEE Consumer Commun. and Networking Conference 2007, 2007, pp. 222-226.
- [37] ION code. Available: <http://sourceforge.net/projects/ion-dtn/>.
- [38] DTNME code. Available: <https://github.com/nasa/DTNME>.
- [39] CCSDS 734.3-B-1 "Schedule-Aware Bundle Routing", Blue Book, July 2019, <https://public.ccsds.org/Pubs/734>
- [40] C. Caini, G. M. De Cola, L. Persampieri, "Schedule-Aware Bundle Routing: Analysis and Enhancements", *Int J Satell Commun Network*, p. 1-13, Dec. 2020. DOI: 10.1002/sat.1384
- [41] "Detemu" code. Available: [https://gitlab.com/antony\\_lion/detemu](https://gitlab.com/antony_lion/detemu).
- [42] "PcapPlusPlus" code. Available: <https://pcapplusplus.github.io/>.
- [43] F. Ludovici and H. Pfeifer, "tc-netem - Linux manual page," 2011.
- [44] "LTP performance analyser" code: [https://gitlab.com/antony\\_lion/ltp\\_performance](https://gitlab.com/antony_lion/ltp_performance).



**Carlo Caini** (Member, IEEE) received the Master's degree in Electrical Engineering "summa cum laude" from the University of Bologna, Italy, in 1986. Since 1990 he has been with the Department of Electrical, Electronics and Information Engineering of the same University, where is working as

Associate Professor of Telecommunications. His main research interests are focused on Delay-/Disruption-Tolerant Networking and Transport Protocols for Space and Satellite Networks. Prof. Caini is co-author of more than 100 papers on international conferences and journals; he has also supervised the development of many free software implementation of DTN protocols and applications.



**Tomaso de Cola** (Member, IEEE) received the "Laurea" degree (with honors) in Telecommunication Engineering, in 2001, and the Ph. D. degree in Electronic and Computer Engineering, Robotics and Telecommunications in 2010 from the University of Genoa, Italy. 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 co-author of more than 100 papers on international conferences and journals. He currently serves as editor of IEEE Communication Letters, IEEE Transactions on Vehicular Technology, IEEE Network, and Elsevier Computer Networks. He is also area director of the Space-Internetworking Services (SIS) within the CCSDS standardization body, as well as chair of the SatCom working group within the ETP NetworkEurope platform.



**Amita Shrestha** received the Masters (MSc) degree in Communications, Systems, and Electronics from the Jacobs University in Bremen in 2009. In 2010, she joined Institute of Communications and Navigation in German Aerospace Center (DLR). In DLR, she worked in the design, development, and operation of operation software for optical terminals. Since few years, she is actively

involved in the CCSSDS standardization of Optical Communication technology on behalf of DLR. She has also supported and managed different projects related to development of optical transceivers, study of free-space optical channels, variable data rate techniques for such channels, and study of quantum communication technology. Her special research interest is the free-space high speed optical downlink from satellite to the ground.



**Antony Zappacosta** Antony Zappacosta received the Bachelor's and Master's degree in Computer Science Engineering from the University of Bologna in 2020 and 2023, respectively. He carried out his Master's thesis at the German Aerospace Center (DLR) in Weßling, Bavaria,

working on the performance evaluation of the Licklider Transmission Protocol in near-Earth optical communications. He is currently involved in artificial intelligence research at DLR.