Bundles Aggregation of Licklider Transmission Protocol over Lossy and Highly Asymmetric Space Network Channels

Yu Zhou[1], Ruhai Wang[2], Lei Yang[1], Siwei Peng[1], and Kanglian Zhao[3]

[1]School of Electronics and Information Engineering, Soochow University, Jiangsu, P. R. China
[2]Phillip M. Drayer Department of Electrical Engineering, Lamar University, Beaumont, TX, USA
[3]School of Electronic Science and Engineering, Nanjing University, Jiangsu, P. R. China
20195228016@stu.suda.edu.cn, rwang@lamar.edu,
20195228043@stu.suda.edu.cn, swpeng@stu.suda.edu.cn,
zhaokanglian@nju.edu.cn,

Abstract. Licklider transmission protocol (LTP) was developed to provide reliable and highly efficient data delivery over unreliable space network channels. Some preliminary studies on LTP's aggregation mechanism in space communication networks have been done. However, the effect on LTP aggregation of data losses caused by lossy space channels has been ignored. Data loss is one of the key features that characterize space networks, and therefore its effect on LTP transmission cannot be left out. In this paper, the LTP aggregation mechanism is studied with focus on its characterization and performance over lossy and highly asymmetric space channels. An analytical model is presented for calculating the minimum number of bundles that should be aggregated within an LTP block for transmission over a lossy channel to avoid report segment (RS) transmission delay caused by highly asymmetric channel rates. The model is validated by realistic file transfer experiments over an experimental testbed infrastructure and packet-level analysis of the results. It is concluded that the aggregation threshold derived from the analytical model functions effectively with respect to resolving RS delay effects and decreasing latency in file delivery, leading to higher transmission efficiency.

Keywords: Space communications, space networks, satellite communications, wireless networks, DTN, Licklider transmission protocol (LTP)

1 Introduction

National Aeronautics and Space Administration (NASA) has recognized delay/disruption tolerant networking (DTN) [1] as the only candidate networking technology that approaches the level of maturity required to provide reliable data delivery service in deep-space communications [2]. As the main data transport protocol of DTN in space networking, the Licklider transmission protocol (LTP) [3, 4] was developed to provide reliable data/file delivery service in unreliable space communications that are characterized by very long propagation delay, frequent link interruptions, and a high data loss rate. The file delivery of LTP is implemented as independent transmissions of LTP data blocks, and each block is fragmented into data segments according to the Maximum Transmit Unit (MTU) at the underlying data link layer. Put simply, the transmission of each LTP block is organized as a "session" which operates as a sequence of data segment exchanges between the sender and the receiver [4].

In additional to a long delay, random link interruptions, and a high data loss rate, asymmetric channel rates is another major feature which characterizes space communications [2]. Channel rate asymmetry in space generally means that the uplink channel rate deployed for acknowledgment (ACK) transmission (from the Earth to the spacecraft or another planet) is much lower than the downlink channel rate deployed for data transmission in the opposite direction [5]. With respect to the operation of LTP, the lower uplink channel rate introduces delay in the transmission of report segments (RSs) (at the receiver) which acknowledge the arrival status of a data block. In other words, it results in a longer RS transmission time and thus, an increase of the round-trip time (RTT) for the block transmission round. For transmission of a very large file conveyed by a large number of LTP blocks, the overall file delivery time will be significantly increased, leading to severe transmission efficiency degradation.

In view of the effect of channel-rate asymmetry in space on LTP, it is reasonable to aggregate multiple bundle protocol (BP) [6, 7] bundles into a single LTP block rather than encapsulating each bundle as an individual block for transmission [3]. A set of preliminary experimental studies on the bundle aggregation mechanism of LTP were done as part of our previous work, mainly with respect to its operation and performance. There are also a few analytical discussions that are supported by the experimental results. This set of work is discussed in Section II—*Related Work*. However, among all these previous studies, the effect of data losses in transmission over lossy space links is completely ignored in analysis. As recognized by the community, data loss is one of the key features that characterize data communications at all levels of space communications. It has especially severe effect over a deep-space channel. Therefore, in practice, the effect of data losses on bundle aggregation in BP/LTP transmission cannot be left out.

In this article, we present an analysis of the LTP aggregation of BP bundles in space communications but focus on its characterization and performance over lossy and highly asymmetric space channels. Analytical modeling is presented for calculating the minimum number of bundles that should be aggregated within a block for transmission over a lossy channel to avoid RS transmission delay caused by highly asymmetric channel rates. The model is validated by realistic file transfer experiments over a testbed infrastructure and packet-level analysis of the experiment results.

2 Related Work

A series of studies have been done for DTN protocols in space, with most of them done jointly by NASA's Jet Propulsion Laboratory (JPL), California Institute of Technology and other research teams. While some of them focus on BP [5, 8-10], numerous studies of LTP examine its design, analysis and performance [11-17]. Among these studies, a set of baseline performance comparisons were conducted between LTP and the most commonly-used TCP/UDP protocols using a statistical analysis method [11]. The comparative evaluations indicate that LTP has a performance advantage for reliable data delivery in a challenging space networking environment. With the joint presence of the very long link delay, highly lossy channel, and highly asymmetric channel rates on the data links, the performance advantage of LTP over other data transport protocols is significant.

As mentioned, some preliminary experimental studies on the bundle aggregation of LTP and its performance in space communications was done in our previous work of DTN in space [18-20]. Most of them focus on an understanding of its operation based on experimental results with a few supported by analytical discussions. However, among this set of previous studies of the bundle aggregation mechanism, the critical effect of data losses in LTP transmission caused by channel error of space channels is completely ignored in analysis. At present, no work has been done in analyzing the bundle aggregation mechanism of LTP with the inevitable effect of data loss taken into consideration for reliable data delivery with asymmetric channel rates.

3 A Scenario of LTP Transmission over Lossy and Asymmetric Space Channels

As discussed, the file delivery of LTP is implemented as independent transmissions of LTP data *blocks*. To achieve the reliable delivery of the file (i.e., to guarantee all the bytes of the entire file successfully delivered), each block is set as 100% red. The transmission of each block is organized as a "session" which operates as a sequence of data segment exchanges between the sender and the receiver. The last segment of a block is transmitted with a checkpoint (CP) flag. A CP segment is intended to trigger the receiver for checking the arrival status of all the segments of the block for data loss or transmission error. If none of the segments is lost or received with error, the receiver acknowledges to the sender by returning a positive acknowledgment, i.e., an RS, to confirm the successful, cumulative reception for the block. If the RS indicates that any data byte of the block is lost, the segment conveying the data byte is retransmitted.

To ensure reliable delivery of CP and RS, both segments are transmitted with a timer set, leading to two important timers, *CP timer* and *RS timer*. These two countdown timers are set to detect a possible loss of CP and RS, respectively, and retransmit them as necessary upon the expiration of the corresponding timer.

To illustrate the effect of data loss on block transmission over lossy and highly asymmetric channels, a scenario

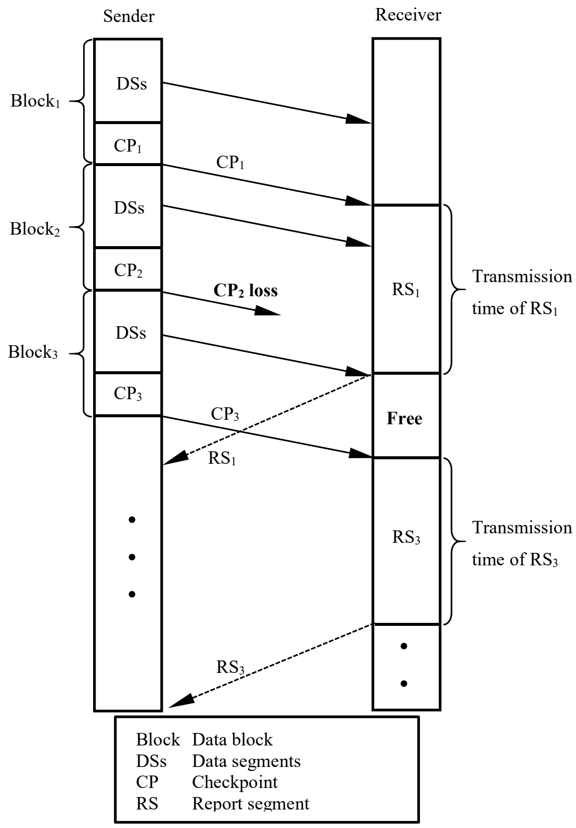

Fig. 1. A scenario of LTP data transmission illustrating the effect of CP loss on block transmission over lossy and highly asymmetric channels.

of LTP block transmission is illustrated in Fig. 1 based on the interactive segments exchange between the sender and the receiver over a lossy space channel. For the sake of simplicity, the transmission scenario is shown only for three blocks numbered as $Block_1$, $Block_2$, and $Block_3$. As illustrated, each of three blocks is fragmented into multiple data segments (DSs) numbered as DS_1-DS_n. The last segment of each block, DS_n, is flagged as the CP, and it is named as CP_1, CP_2, and CP_3 for three blocks in Fig. 1, respectively. The blocks and the segments of each block are transmitted in a continuous manner. For transmission of $Block_1$, as soon as CP_1 segment (which is also a DS) is successfully received, the receiver checks all the segments of $Block_1$ for possible data loss. Assuming that none of the segments of $Block_1$ are lost or corrupted, the RS corresponding to CP_1, here named RS_1, is immediately transmitted as a positive report of the cumulative reception for the block. The sender then sends a report acknowledgment (RA), RA_1, to acknowledge the receipt of RS_1, which closes the transmission session of $Block_1$.

Because of the very slow ACK channel rate, the time taken by the receiver in transmitting RS_1 is longer than the time taken by the sender in transmitting $Block_1$. In this case, while RS_1 is being transmitted, the segments of $Block_2$ are already on the way to the receiver. The last segment of $Block_2$, CP_2, is expected to initiate a check of the arrival status of the entire block. The segments of $Block_2$ are expected to arrive at the receiver before the entire RS_1 is completely transmitted. Even if all the segments of $Block_2$ are successfully delivered to the receiver without any data loss, RS_2 is not sent until RS_1 is completely transmitted because of the long transmission time of RS_1. In other words, RS_2 has to wait for transmission until RS_1 is completely transmitted. This is the origin of the RS waiting time and RS transmission queue, leading to an increase of the RTT length for the block transmission round.

However, because the channel is lossy or error-prone, any DS of $Block_2$ can be corrupted by channel error during transmission. Suppose that due to transmission error over the data channel, CP_2 is corrupted or lost while all other

DSs are successfully received, as illustrated in Fig. 1. The loss of CP_2 prevents the receiver from responding with an RS on the arrival status of the block. Therefore, the RS in response to CP_2/$Block_2$, named RS_2, is not generated at the receiver for transmission over the ACK channel. A lack of RS_2 for transmission actually relieves the severely constrained ACK channel transmission capacity/rate. This can be observed in Fig. 1 where there is some "free" time at the receiver after RS_1 is completely transmitted. The receiver is "free" because there is no data to send at this time. The receiver keeps itself "free" until the receipt of CP_3 of $Block_3$ which triggers the receiver to generate RS_3 for transmission.

In the light of the earlier discussions, a high LTP block transmission data loss rate resulting from a lossy space channel is expected to reduce the RS waiting time and transmission round-trip time. As a result, it is expected to improve the transmission performance of LTP over links characterized by extremely asymmetric data rates, especially for transmission of a large file which is conveyed by a large number of bundles for which many blocks is more likely transmitted.

4 Analytical Modeling for LTP Aggregation of Bundles over Lossy Space Channels

The total length of an encapsulated LTP block at the link layer, termed L_{Block_Link}, was formulated as a function of the number of bundles to be aggregated and the fragmented segment size in [18, 20]. The formula is reiterated below

$$L_{Block\_Link} = \frac{N_{Bundle} \times \left(L_{Bundle} + L_{Bundle\_Head}\right) \times \left(L_{Ltp\_Seg} + L_{Frame\_Head}\right)}{L_{Ltp\_Seg}} \tag{1}$$

in which

N_{Bundle} is the number of bundles aggregated within an LTP block,
L_{Bundle} is the length of a bundle,
L_{Bundle_Head} is the header length of each bundle,
L_{Ltp_Seg} is the average length of a fragmented segment, and
L_{Frame_Head} is the total length of the overhead added to a segment because of the encapsulation processes.

The total length of the frame overhead L_{Frame_Head} in (1) is actually the total encapsulation overhead starting from the LTP layer until the link layer. Therefore, it can be simply formulated as a sum of the length of an individual head added at Layer 1 (the link layer) up to the LTP layer, i.e.,

$$L_{Frame\_Head} = \sum_{i} L_{Head\_Layer\,(i)} \tag{2}$$

in which $L_{Head\_Layer\,(i)}$ is the length of head added by the *i*th layer. In specific, from bottom to top, $L_{Head\_Layer\,(1)}$ is the head length at the link layer, $L_{Head\_Layer\,(2)}$ is the head length at the IP layer, $L_{Head\_Layer\,(3)}$ is the head length at the UDP layer, and $L_{Head\_Layer\,(4)}$ is the head length at the LTP layer. In fact, the numerical value of L_{Frame_Head} varies depending on the protocol configuration as in some applications, a specific protocol may not be needed.

It is important to note that the formula of L_{Block_Link} in (1) ignores the effect on LTP aggregation of data losses caused by lossy space channels. In other words, the effect of channel loss or error caused by channel noise is not considered. However, as mentioned, data loss caused by channel error is one of the key features that characterize space communications, and its effect on LTP transmission cannot be ignored.

Let $\overline{T_{RS}}$ be the average transmission time of an RS out of the total number of RSs. As discussed, an RS can only be triggered by a successfully delivered CP segment. In other words, the CP segments that are lost or corrupted when received at the receiver do not generate the RSs. Therefore, $\overline{T_{RS}}$ can be formulated as

$$\overline{T_{RS}} = \frac{(1-p)^{8\times L_{CP}} \times L_{RS}}{R_{RS}} \tag{3}$$

in which

p denotes an "effective net channel bit-error-rate (BER)" of the data channel which represents the net overall data loss rate (i.e., the transmission quality of a channel),

L_{CP} is the length of the CP segment in bytes,

L_{RS} is the length of an encapsulated RS at the link layer, and

R_{RS} is the uplink channel rate available for RS transmission.

Denote $\overline{T_{Block}}$ as the average transmission time of a single block out of the total number of blocks for the entire file transmission. Let R_{Data} be the data channel rate available for downloading data from a deep-space craft (i.e., Moon lander or Mars lander). Then, $\overline{T_{Block}}$ can be formulated as

$$\overline{T_{Block}} = \frac{L_{Block\_Link}}{R_{Data}} \tag{4}$$

Because of the "one RS per block" acknowledgment policy of LTP, the total number of RSs transmitted by the receiver is actually equal to the total number of LTP data blocks that the entire file is divided among to convey all the data bytes. In order to avoid RS transmission delay due to channel-rate asymmetry, it is required that during the entire file delivery process, the average transmission time of a block must be greater than or equal to the average transmission time of an RS sent by the data receiver, with each segment serving as a positive report of the cumulative reception for an individual block (i.e., a positive ACK), i.e., $\overline{T_{Block}} \geq \overline{T_{RS}}$., Their relationship can be formulated using different factors as

$$\frac{L_{Block\_Link}}{R_{Data}} \geq \frac{(1-p)^{8\times L_{CP}} \times L_{RS}}{R_{RS}} \tag{5}$$

With (5), L_{Block_Link} can be formulated as

$$L_{Block\_Link} \geq \frac{(1-p)^{8\times L_{CP}} \times L_{RS} \times R_{Data}}{R_{RS}} \tag{6}$$

L_{Block_Link} in (6) actually means that the total length of an encapsulated LTP block (when passed to the link layer) for transmission over the data channel must be large enough so that the average block transmission time is longer than the average RS transmission time. By this, the RS transmission delay due to the low ACK channel rate can be avoided.

Plugging the formula of L_{Block_Link} in (1) into (6), N_{Bundle} can be written as a function of other factors, and it is shown as

$$N_{Bundle} \geq \frac{(1-p)^{8\times L_{CP}} \times L_{RS}}{R_{RS}} \times \frac{R_{Data} \times L_{Ltp\_Seg}}{(L_{Bundle} + L_{Bundle\_Head}) \times (L_{Ltp\_Seg} + L_{Frame\_Head})} \tag{7}$$

The formula of N_{Bundle} in (7) defines the number of bundles to be aggregated within an LTP block so that the delay of RS transmission due to the effect of channel-rate asymmetry can be avoided. Therefore, the threshold numerical value of N_{Bundle} is actually a minimum number of aggregated bundles to be able to avoid the delay of RS transmission. Denote this minimum number of bundles as N_{Bundle_Min}. With the effects of channel-rate asymmetry, channel data loss rate, data fragmentations, and the total overhead taken into consideration, it can be formulated as

$$N_{Bundle\_Min} = \left\lceil \frac{(1-p)^{8\times L_{CP}} \times L_{RS}}{R_{RS}} \times \frac{R_{Data} \times L_{Ltp\_Seg}}{(L_{Bundle} + L_{Bundle\_Head}) \times (L_{Ltp\_Seg} + \sum_i L_{Head\_Layer\,(i)})} \right\rceil \tag{8}$$

5 Numerical Experimental Results and Model Validation

In this section, the numerical results of the file transfer experiments over the testbed are presented to validate the predictions of the analytical model for calculating the recommended minimum number of bundles aggregated within a block. The discussion focuses on a study of whether or not the delay of RS transmission caused by channel-rate asymmetry is avoided; the magnitude of this delay is measured by comparing the measured RTT lengths of the first and last block transmission rounds.

5.1 Overview of Experimental Infrastructure and Configurations

This analysis and the resulting value for the minimum number of bundles aggregated within a block to avoid the delay effect of RS transmission derived in Section IV are validated through file transfer experiments using an experimental infrastructure that emulates communication in a cislunar operational environment. The experimental infrastructure adopted for the proposed file transfer is the PC-based space communication and networking testbed (SCNT) [11]. The SCNT infrastructure was validated through a series of our previous studies in performance evaluation of a protocol suite proposed for space networks and deep-space communications [11-13]. For a detailed description of it, refer to [11].

The protocol implementation, BP/LTP, used for the experiments was adapted from the Interplanetary Overlay Network (ION) distribution v3.6.2. ION is a software implementation of the DTN protocol suite for space networks and deep-space communications developed by NASA's JPL [21]. IP and Ethernet were adopted to serve at the underlying network layer and data link layer, respectively. As the frame length (MTU size) of Ethernet is 1500 bytes, each LTP segment is configured to be 1400 bytes, making it fit within the frame MTU. Bundle size is arbitrarily fixed at 1000 bytes.

A one-way delay of 1.35 s, which is common over a cislunar channel, was introduced to each of the data and ACK channels to emulate the signal propagation delay in deep space. The effect of the channel-rate asymmetry on file transmission was implemented by configuring a high channel speed ratio (CR), 500/1, resulting from a downlink channel rate of 2 Mbps and an uplink channel rate of 4 Kbps. A text file of 1 Mbyte is transmitted from the sender to the receiver by running LTP together with associated protocols to measure the performance of the protocol.

As this study focuses on the effect of data loss on bundle aggregation in LTP, different levels of data loss are imposed as channel loss rate or channel quality for LTP file transmission. Four different effective net BERs (i.e., p in (3)), including 10^{-8}, 10^{-7}, 10^{-6} and 10^{-5}, are configured in our experiments. While the BERs of 10^{-8} and 10^{-7} are introduced to emulate a *less lossy* channel, the BERs of 10^{-6} and 10^{-5} are introduced to emulate a *lossy* channel and a *highly lossy* channel, respectively. The study is mainly concerned with the effect of data loss caused by the channel error on reliable data delivery of LTP. For this reason, each LTP block is declared 100% *red* in our experiments.

Although the size of an LTP segment is fixed at 1400 bytes in this study, the LTP block sizes adopted for the experiments vary drastically. The number of bundles aggregated within a block varies over a wide range. By varying block size in this way, we study how the transmission performance of LTP varies with variations of block size measured in the number of bundles.

5.2 Effect of RS Delay on File Delivery over Lossy Channels

Fig. 2 provides a comparison of the RTT lengths measured for the first and the last block transmission rounds in transmitting a 1-Mbytes file over an emulated cislunar space communication infrastructure having different channel qualities (i.e., with the BERs of 10^{-8}, 10^{-7}, 10^{-6}, and 10^{-5}). The RTT length for the first block transmission round is denoted as RTT_{1st} while it is denoted as RTT_{Last} for the last round. The communication infrastructure is configured with a CR of 500/1 and 5 bundles/block. It is observed that the RTT_{1st} is consistently around 2.9 sec at all three channel BERs. This is reasonable provided that the one-way link delay configured for the experiments is 1.35 sec to emulate a cislunar communication scenario.

In comparison, RTT_{Last} is much higher than RTT_{1st} at all three channel BERs considered, varying in the range of 22~24 s. This leads to a significant difference between RTT_{1st} and RTT_{Last}, around 20 s, regardless of the channel loss rate. Given that RTT_{1st} is shorter than three seconds, the RTT length difference of more than twenty seconds is a very strong indication that RTT_{Last} has the delay effect of RS transmission over the forward uplink ACK channel and/or delay effect of block transmission over the return downlink data channel. Because block transmission is initiated by the sender without waiting for any signal from the receiver and transmission is therefore continuous, no delay should have been experienced. Therefore, the only delay effect contributing to RTT_{Last} is the delay in RS transmission, which is caused by the channel-rate asymmetry.

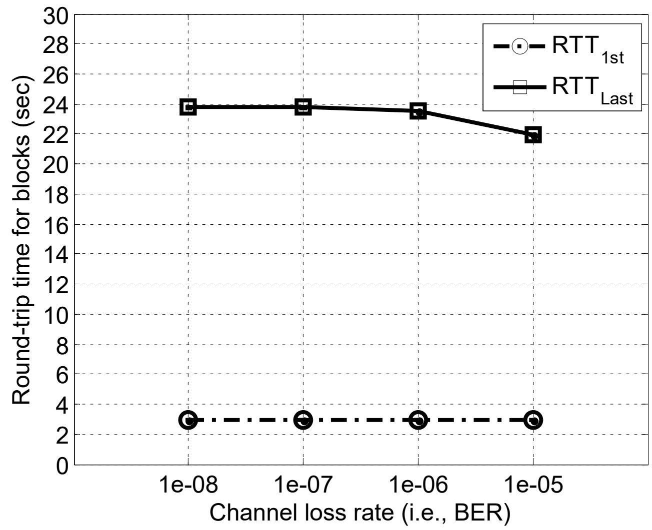

Fig. 2. Comparison of the RTT lengths measured for the first and the last block transmission rounds of LTP in transmitting a 1-Mbyte file over an emulated cislunar space communication infrastructure with highly asymmetric channel rates (with the CR of 500/1) and different channel qualities (i.e., with different loss rates), and LTP configured with 5 bundles/block.

The drastic difference of the RTT lengths can be easily explained by considering the number of bundles aggregated. The testbed is configured to have five bundles aggregated within a single block for this set of transmissions. Five bundles (aggregated within a block) are too few to resolve the delay effect of the RS transmission caused by the very low ACK channel rate. The time difference between RTT_{Last} and RTT_{1st} is so large because the RS delay resulted from each block transmission is accumulated over the transmission rounds of all the blocks. Furthermore, all the delay effects experienced by the previously transmitted blocks add up to the RTT length for the last round, which makes RTT_{Last} extremely long compared to RTT_{1st} (without any delay effect of RS transmission).

It is also observed that along with the channel BER increase, there is a slight decrease of RTT_{Last}. While the decrease is minor for the BER increase from 10^{-8} to 10^{-6}, it is obvious from 10^{-6} to 10^{-5}, around 2 sec. This is reasonable according to the discussion in Section IV. Formulated as $1-(1-p)^{8\times L_{CP}}$, the probability that a CP segment of a block is corrupted during file transmission is higher when a higher channel loss rate (p) is experienced. Given that the size of a file for transmission is fixed, the number of blocks is fixed, implying that the number of CP segments is also fixed according to the "one CP segment per block" policy of LTP. With a higher loss probability for a CP segment, the number of CP segments that are successfully received by the receiver decreases. Following the "one RS per block" acknowledgment policy of LTP [5], the number of RSs generated at the receiver for transmission in response to the arrived CP segments must likewise decrease. With fewer RSs transmitted over the constrained ACK channel, the number of RS in waiting for transmission is reduced, leading to a shorter queue for the RSs. This implies that the RSs have shorter waiting time for their transmission. The shorter waiting time of RSs at the receiver means that the RSs generated for the file transmission can be received by the data sender sooner. This leads to a shorter RTT length for all the block transmission rounds and therefore, a shorter RTT for the last round. This clarifies the drop of RTT_{Last} along with the increase of the channel BER from 10^{-6} to 10^{-5} in Fig. 2.

Fig. 3 illustrates the time sequence graph (TSG) [22, 23] for a transmission scenario over a highly lossy channel with a BER of 10^{-5}, i.e., for the transmission which demonstrated a slight decrease of RTT_{Last} in Fig. 2. The original data transmission line and the corresponding initial RS transmission line vary in a linearly increasing pattern but with different slope rates. This leads to significantly different lengths of RTT—RTT_{1st} for the first block has a very short length around 2.8 sec, and RTT_{Last} for the last block is very long, around 22 sec. The TSG shows various loss and retransmission events starting from the RS transmission line. It is found that around eighty corruption/retransmission events are observed during the entire file delivery. As the transmission is configured with

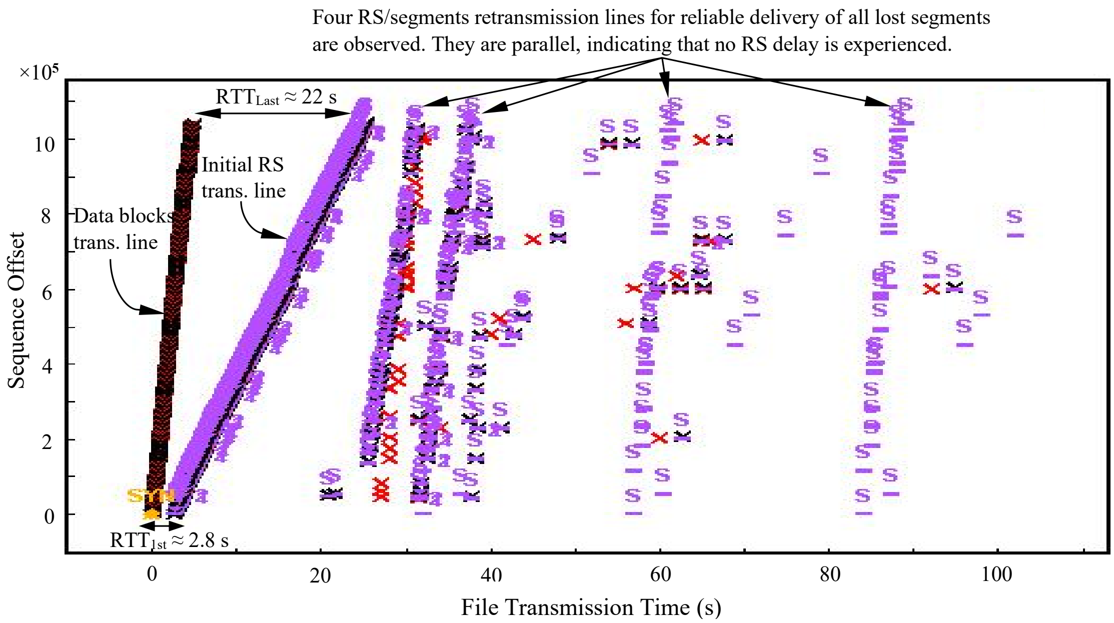

Fig. 3. TSG illustrating the LTP transmission at packet level at the sending node for delivery of a 1-Mbyte file over a highly lossy cislunar channel (with a BER of 10^{-5}) with highly asymmetric channel rates (with a CR of 500/1) and LTP configured with 5 bundles/block.

a BER of 10^{-5} and a given file size of 1 Mbyte, this is correct according to the calculation of $\lceil 10^6 \times 8 \times 10^{-5} \rceil = 80$ which is statistically an average number of the data corruption (and retransmission) events.

These corruption events cause either losses of regular data segments or losses of CP segments. The loss events result in retransmission of the segments and even re-retransmission(s) of them. That is why in addition to the initial RS line, four more RS/segment retransmission lines are observed, indicating segments that are retransmitted for four times. Some segments are even retransmitted more than four times. In contrast to the original RS transmission line, these four retransmission lines are parallel, which indicate that there is no RS delay experienced in transmitting them. This is because of the decrease of the number of RSs for transmission along with the increase of the transmission attempts. Each attempt causes additional blocks to be successfully delivered and leaves fewer blocks to be retransmitted in the next attempt. Fewer blocks result in fewer CPs, resulting in fewer RSs sent by the receiver. This can also be observed from the fact that the RS lines are becoming sparser along with the transmission because of the decrease in the number of RSs sent.

The details of the loss of regular segments and the absence of some RSs for transmission over the ACK channel because of the failure of delivery of the corresponding CP segments can be observed from a TSG at LTP segment level. Fig. 4 shows a zoom-in view of a selected portion of the TSG for the transmission of two LTP data blocks. The two blocks, numbered as $Block_1$ and $Block_2$, are transmitted in a continuous manner around 2 sec. It is observed that the RS for $Block_2$ is received around 10 sec while it is unexpectedly not received for $Block_1$. This is an indication that $Block_2$ experiences regular data corruption/loss (not CP loss) due to the high channel error rate of 10^{-5}, and the situation for $Block_1$ is different.

For the transmission of $Block_2$, it is transmitted in four segments in order, and the last segment should be sent as its CP segment. After one round trip, the RS for the block is received around 10 sec, showing the RS line corresponding to the sequence numbers for the first, third and fourth (CP) segments. This indicates that these three segments are successfully delivered. But the RS line corresponding to the sequence numbers for the second segment is missing, and it is shown as a gap in the RS line to be filled. This means that the second segment is not successfully delivered because of corruption, and the receiver requests for retransmission of it.

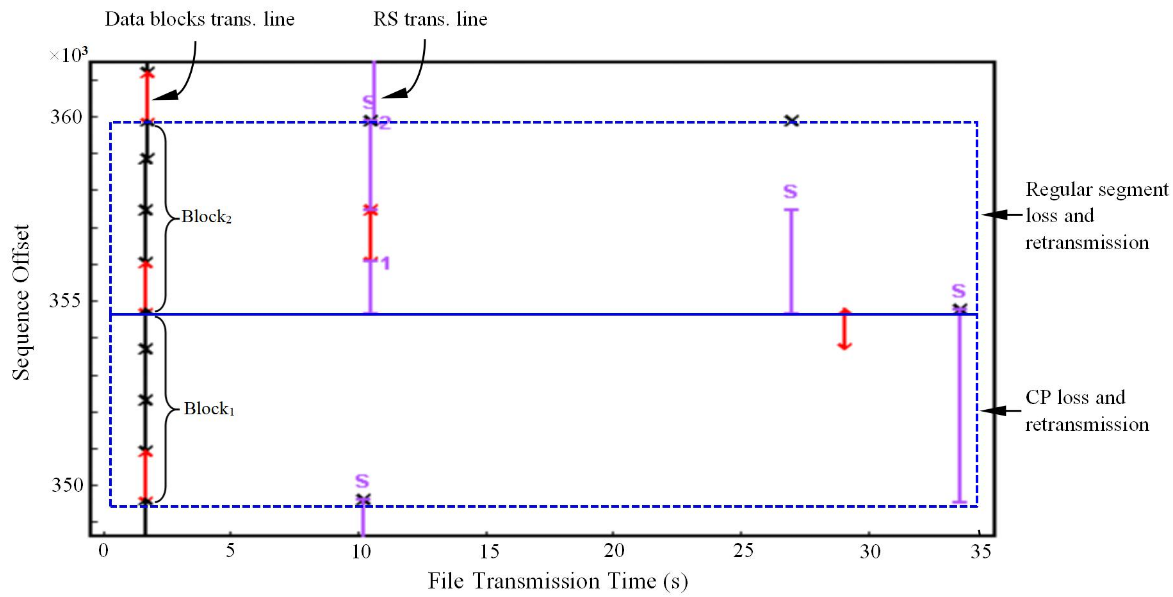

Fig. 4. A zoomed-in view of a selected portion of the TSG in Fig. 3, illustrating transmission of two LTP data blocks of a file—Block$_1$ having a CP loss and Block$_2$ having a regular segment loss.

The TSG shows that the second segment is then quickly retransmitted in *red* (for transmission reliability) as a separate block, and it is eventually successfully delivered. The acknowledging RS for the successful delivery of the second segment (up to the beginning of the block) arrives at the sender around 27 s, signaling the successful delivery of all four segments of the entire block. The mandatory RA with the sequence number aligned with the end of the block is sent immediately by the sender in response to the receipt of the RS. The RA is shown on top of the RS.

For transmission of Block$_1$ illustrated in Fig. 4, the missing of its RS is shown as a gap in the RS line to be filled. It can be inferred that the loss of the RS is caused by the corruption/loss of the CP segment (i.e., the last segment) of Block$_1$ to the receiver. Otherwise, if the CP segment is not corrupted, an RS should be sent to the sender reporting the delivery status, but this does not occur. For all four segments of the block, only the last segment (i.e., the CP) is retransmitted. This is a strong indication that the CP segment of Block$_1$ was corrupted by channel error.

The sender resends the lost CP segment for Block$_1$ upon the expiration of the CP timer around 29 s. As soon as the resent CP is successfully received, the receiver checks all four segments of the block (including the previously received three segments and the resent CP). It is found that all four segments are successfully received, implying a successful delivery of the entire block. Therefore, the RS for the block is sent in response immediately. As shown in the enlarged view, the RS arrives at the sender around 34 s, covering all four segments of Block$_1$. By this, Block$_1$ is successfully delivered and acknowledged.

The numerical value for a minimum number of bundles to be aggregated within a block to avoid the RS delay (i.e., N_{Bundle_Min}) can be calculated from the model using the pre-defined transmission conditions and protocol setting. As mentioned, the encapsulated segment size for transmission at the link layer is around 1440 bytes. So, for transmission over less lossy channels (e.g., with a BER of 10^{-7}), it is calculated that

$$(1-p)^{8\times L_{CP}} = (1-10^{-7})^{8\times 1440} \approx 99.89\%$$

and according to (8), $N_{Bundle\_Min} = 33$. The same value is given at the BER of 10^{-8}.

Similarly, for a transmission over a lossy channel with a BER of 10^{-6}, it is calculated that

$$(1-p)^{8\times L_{CP}} \approx 98.86\% \text{ and } N_{Bundle\_Min} = 33$$

and over a highly lossy channel at the BER of 10^{-5},

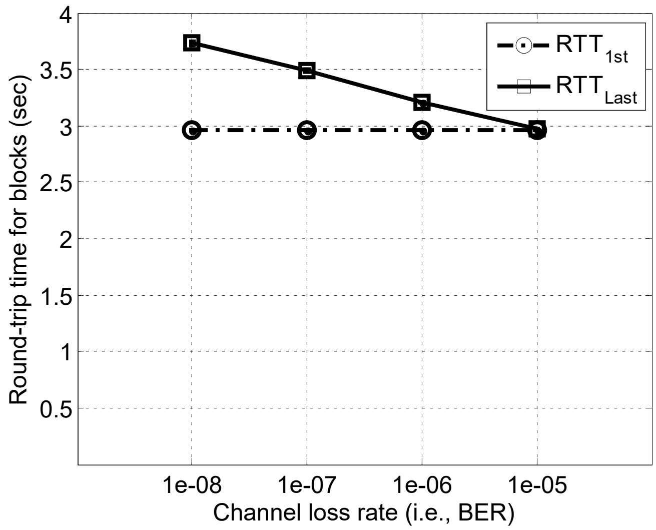

Fig. 5. Comparison of the RTT lengths measured for the first and the last block transmission rounds of LTP configured with 30 bundles/block.

$$(1-p)^{8\times L_{CP}} \approx 89.12\% \text{ and } N_{Bundle\_Min} = 30$$

The calculation result indicates that over less lossy and lossy channels (i.e., with the BERs of 10^{-8}-10^{-6}), the minimum number of bundles to be aggregated within a block to avoid RS delay is equal, 33 bundles/block in each case. The calculated minimum number of bundles to be aggregated is thirty (i.e., 30 bundles/block) for transmission over a highly lossy channel with a channel BER of 10^{-5}.

As a validation to the model, Fig. 5 presents a comparison of the RTT lengths measured for the first block transmission round and the last block transmission round of a 1-Mbyte file configured with thirty bundles aggregated within a single block, or simply, 30 bundles/block. All the transmission conditions are the same as in Fig. 2 except for the number of aggregated bundles. Comparing with the observations in Fig. 2, the similarities are: (1) RTT_{1st} remains unchanged and low, less than 3 s, at all four channel BERs; (2) RTT_{Last} is longer than RTT_{1st} at the BERs of 10^{-8}-10^{-6}; and (3) RTT_{Last} shows slight decrease with the increase of the channel BER. All these observations are clarified in the discussion of Fig. 2, and the explanations still apply to the results in Fig. 5 even though the number of aggregated bundles is six times greater.

There are two main differences between Fig. 2 and Fig. 5. First, RTT_{Last} with 30 bundles/block in Fig. 5 is much shorter than the RTT_{Last} with 5 bundles/block in Fig. 2. This leads to only minor differences between RTT_{Last} and RTT_{1st} for the transmissions with 30 bundles/block. The second difference is with the transmission at the channel BER of 10^{-5}—while RTT_{Last} is significantly longer than RTT_{1st} with 5 bundles/block, they are roughly equal with 30 bundles/block.

By way of explanation for the first observed difference, RTT_{Last} in Fig. 5 is significantly shorter because the transmissions are configured with a larger number of bundles (30 bundles) aggregated within a block. With so many bundles aggregated, given the fixed file size, the number of blocks sent by the sender to convey the entire file decreases significantly. Following the "one RS per block" policy, the number of RSs generated at the receiver for transmission also decreases. This leads to a much shorter waiting time for the RS transmission over the ACK channel and therefore, a much shorter RTT length for almost every block. As a result, the RTT length for the last block, RTT_{Last}, drops drastically.

The second difference observed, i.e., RTT_{Last} = RTT_{1st} at the BER of 10^{-5}, has a similar explanation: it happens because the transmission in Fig. 5 is configured with 30 bundles/block. According to the aforementioned calculation, with the BER of 10^{-5}, the minimum number of bundles to be aggregated within a block to avoid RS delay is thirty, i.e., N_{Bundle_Min}=30. In other words, the transmission configured with 30 bundles/block is able to resolve the delay effect caused by the CR of 500/1 and does not impose delay in RS transmission. Therefore, the RTT length for the

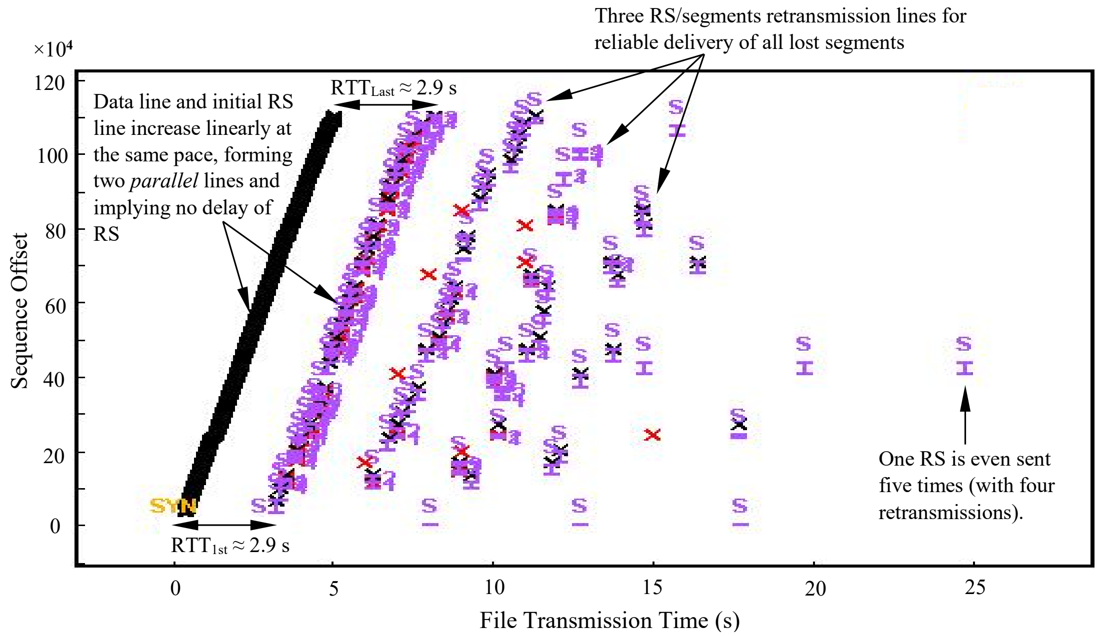

Fig. 6. TSG illustrating the transmission at packet level for delivery of a 1-Mbyte file over a highly lossy cislunar channel (with a BER of 10^{-5}) and LTP configured with 33 bundles/block.

last block transmission round remains unchanged compared to the ones measured for the first block and all other blocks.

In comparison, RTT_{Last} is slightly longer than RTT_{1st} at the BERs of 10^{-8}, 10^{-7} and 10^{-6}. In other words, there is a minor RS delay effect experienced for these three transmissions. In comparison, RTT_{Last} is slightly longer than RTT_{1st} at the BERs of 10^{-8}, 10^{-7} and 10^{-6}. In other words, there is minor effect of RS delay experienced for these three transmissions. According to the calculation derived above, the minimum number of bundles to be aggregated within a block to avoid the RS delay is thirty-three, i.e., $N_{Bundle\_Min} = 33$, for the transmissions at all three BERs. However, the file transmissions in the experiment were actually configured with 30 bundles/block. So, the aggregated numbers of bundles in the experiment were slightly fewer than the required minimum number of bundles. As a result, the delay effect is resolved significantly but not entirely. In other words, there is still some delay effect experienced for many blocks. The resulting minor delay effects are accumulated toward the last block transmission. This causes RTT_{Last} to be slightly longer than RTT_{1st} as shown in Fig. 5. Note that the delay effect is diminished with the increase of the BER because of the increment in CP loss events that leads to fewer RS for transmission and thus shorter delay of RS, as discussed with Fig. 2.

For an illustration of the LTP transmission at packet level without the effect of RS delay experienced, Fig. 6 presents the TSG for the same transmission of 1-Mbyte file as the one presented in Fig. 3 but configured with a much larger number of bundles aggregated within a block, 33 bundles/block. As calculated from the model, given that a bundle size of 1000 byte and a segment size of 1400 bytes are configured, thirty (30) bundles are the required minimum number of bundles to be aggregated to avoid the RS delay at the BER of 10^{-5}. Therefore, it is expected that a transmission with 33 bundles/block resolves thoroughly the delay of RS transmission and thus, leads to an equal length of the RTT for all the block transmission rounds.

Comparing this figure with the TSG with 5 bundles/block in Fig. 3, the main difference is that both the data line and the corresponding RS line in Fig. 6 increase linearly at the same pace, forming two parallel lines. This leads to nearly consistent length of RTT for all the blocks. Both RTT_{Last} and RTT_{1st} are around 2.9 s, as an indication that the configuration of 33 bundles/block resolves the delay effect of RS transmission. This means that the calculated numerical value of N_{Bundle_Min}, 30 bundles/block, is able to overcome the effect of RS delay even over a highly lossy channel. Because of the high data loss events (with a BER of 10^{-5}), three parallel RS/segments retransmission lines are obviously observed in Fig. 6, in addition to the initial RS line. One RS is even retransmitted five times with four retransmission attempts.

6 Conclusions

In this article, an analytical model is presented for the minimum number of BP bundles to be aggregated within an LTP block for transmission over a lossy channel so that the RS transmission delay imposed by highly asymmetric channel rates can be resolved. The model is validated by realistic file transfer experiments over a testbed infrastructure and packet-level analysis of the experimented results. It is found that regardless of the channel loss rate, the minimum number of aggregated bundles derived from the analytical model functions effectively with respect to resolving all the RS delay effect caused by asymmetric channels and achieves high goodput performance.

It is concluded that regardless of the number of bundles aggregated within a block (i.e., block size), along with the increase of channel loss rate, there is slight decrease of RTT. This is because with a high loss rate involved, both the number of CP segments that are successfully delivered to the receiver and the number of RS segments generated in response decrease. The decrease of the number of RS segments leads to a shorter waiting time of RS segments for their transmission and thus, a shorter RTT for all the block transmission rounds. On the other hand, regardless of channel loss rate, along with the increase of the number of aggregated bundles, the RTT for all the block transmission rounds also decreases because of the reduced number of data blocks/CP segments originally sent by the sender. In both cases of the RTT decrease, the transmission leads to shorter file delivery time and thus increase of goodput. The performance keeps improving along with the increase of block size until the number of aggregated bundles are as many as around a threshold number which is the minimum number of aggregated bundles calculated from the model.

REFERENCES

1. Burleigh, S., Hooke, A., Torgerson, L., Fall, K., Cerf, V., Durst, R., Scott, K., Weiss, H.: "Delay-tolerant networking: An approach to inter-planetary Internet," *IEEE Communications Magazine*, vol. 41, No. 6, pp. 128–136, Jun. 2003.
2. Consultative Committee for Space Data Systems, "Rationale, scenarios, and requirements for DTN in space," CCSDS 734.0-G-1. Green Book. Issue 1. Washington, DC, USA: CCSDS, August 2010.
3. Consultative Committee for Space Data Systems, "Licklider transmission protocol for CCSDS," CCSDS 734.1- B-1. Blue Book. Issue 1. Washington, DC, USA: CCSDS, May 2015.
4. Ramadas, M., Burleigh, S., Farrell, S.:"Licklider Transmission Protocol Specification," Internet RFC 5326, Sept. 2008.
5. Sabbagh, A., Wang, R., Zhao, K., Bian, D.: "Bundle Protocol Over Highly Asymmetric Deep-Space Channels," *IEEE Transactions on Wireless Communications*, vol. 16, no. 4, April 2017, pp. 2478-2489.
6. Scott, K., Burleigh, S.: "Bundle protocol specification," IETF Request for Comments RFC 5050, November 2007, [Online]. Available: http://www.ietf.org/rfc/rfc5050.txt.
7. Consultative Committee for Space Data Systems, "Bundle protocol specifications," CCSDS 734.2- B-1. Blue Book. Issue 1. Washington, DC, USA: CCSDS, September 2015.
8. Feng, C., Wang, R., Bian, Z., Doiron, T., Hu, J.: "Memory Dynamics and Transmission Performance of Bundle Protocol (BP) in Deep-Space Communications," *IEEE Transactions on Wireless Communications*, vol. 14, No. 5, May 2015, pp. 2802-2813.
9. Zhao, K., Wang, R., Burleigh, Scott C., Sabbagh, Alaa., Wu, Wenwei., Sanctis, Mauro De.:"Performance of Bundle Protocol for Deep-Space Communications," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 52, No. 5, October 2016, pp. 2347-2361.
10. Wang, R., Sabbagh, A., Burleigh, S., Zhao, K., Qian, Y.:"Proactive Retransmission in Delay-/Disruption-tolerant Networking for Reliable Deep-Space Vehicle Communications," *IEEE Transactions on Vehicular Technology*, vol. 67, No. 10, October 2018, pp. 9983-9994.
11. Wang, R., Burleigh, S., Parik, P., Lin, C-J., Sun, B.: "Licklider Transmission Protocol (LTP)-based DTN for cislunar communications," *IEEE/ACM Transactions on Networking*, vol. 19, No. 2, April 2011, pp. 359-368.

12. Zhao, K., Wang, R., Burleigh, S., Qiu, M., Sabbagh, A., Hu, J.: "Modeling Memory Variation Dynamics for the Licklider Transmission Protocol in Deep-Space Communications," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, No. 4, October 2015, pp. 2510-2524.
13. Yu, Q., Wang, R., Zhao, K., Li, W., Sun, X., Hu, J., Ji, X.:"Modeling RTT for DTN Protocols over Asymmetric Cislunar Space Channels," *IEEE Systems Journal*, vol. 10, No. 2, June 2016, pp. 556-567.
14. Lent, R.: "Regulating the Block Loss Ratio of the Licklider Transmission Protocol," In *Proceedings of 2018 IEEE 23th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, Barcelona, Spain, Sept. 17-19, 2018.
15. Lent, R.: "Analysis of the block delivery time of the Licklider transmission protocol," *IEEE Transactions on Communications*, vol. 67, No. 1, January 2019, pp. 518-526.
16. Alessi, N., Caini, C., de Cola, T., Raminella, M.: "Packet Layer Erasure Coding in Interplanetary Links: The LTP Erasure Coding Link Service Adapter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 56, No. 1, Feb. 2020, pp. 403-414.
17. Alessi, N., Burleigh, S., Caini, C., de Cola, T.: "Design and performance evaluation of LTP enhancements for lossy space channels," *International Journal of Satellite Communications and Networking* Special Issue: ASMS/SPSC 2016, vol. 37, Issue 1, January/February 2019, pp. 3-14.
18. Wang, R., Wei, Z., Zhang, Q., Hou, J.:"LTP Aggregation of DTN Bundles in Space Communications," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, No. 3, July 2013, pp.1677-1691.
19. Hu, J., Wang, R., Zhang, Q., Wei, Z., Hou, J.: "Aggregation of DTN Bundles for Space Internetworking System," *IEEE Systems Journal*, vol. 7, No. 4, December 2013, pp. 658-668.
20. Yang, Z., Wang, R., Yu, Q., Sun, X., Sanctis, M., Zhang, Q., Hu, J., Zhao, K.: "Analytical characterization of Licklider Transmission protocol (LTP) in cislunar communications," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 50, No. 3, July 2014, pp. 2019-2031.
21. Burleigh, S.: "Interplanetary overlay network design and operation v3.6.2," JPL D-48259, Jet Propulsion Laboratory, California Institute of Technology, CA, March 2019, [Online]. Available: http://sourceforge.net/projects/ion-dtn/files/latest/download.
22. TCP Connection Analysis Tool: TCPTRACE, [Online] https://linux.die.net/man/1/tcptrace.
23. Ltptrace Analysis Tool: LTPTRACE, [Online] http://ion.ocp.ohiou.edu/content/ltptrace