An Experimental Analysis of Checkpoint Timer of Licklider Transmission Protocol for Deep-Space Communications

Lei Yang[1], Ruhai Wang[2], Xingya Liu[3], Yu Zhou[1], Jie Liang[2], and Kanglian Zhao[4]
[1]School of Electronics and Information Engineering, Soochow University, Suzhou, Jiangsu, P. R. China
[2]Phillip M. Drayer Department of Electrical Engineering, Lamar University, Beaumont, TX, USA
[3]Department of Computer Science, Lamar University, Beaumont, TX, USA
[4]School of Electronic Science and Engineering, Nanjing University, Nanjing, Jiangsu, P. R. China

***Abstract*—As the main data transport protocol of delay-disruption-tolerant networking (DTN) targeting deep-space communications, Licklider transmission protocol (LTP) is developed to provide reliable and highly efficient data delivery over unreliable communication channels that are characterized by very long delay and/or frequent and random link interruptions. A checkpoint (CP) segment of LTP is sent, with a timer set (or simply, CP timer) to check the arrival status of the entire data block at the receiver, and it is retransmitted upon the expiration of the CP timer prior to reception of an acknowledgment. In the previous work, an analytical model is built for the CP timer setting in such a way that reliable file delivery in deep space is ensured with a minimum number of retransmission attempts taken for energy-efficient deep-space communications. However, the model was not validated. In this paper, a packet-level analysis based on realistic file transfer experiments using a PC-based testbed is presented to validate the built model for LTP in deep-space communications characterized by a long latency and highly asymmetric channel rate.**

***Index Terms*—DTN, bundle protocol, space networks, satellite and space communications.**

I. 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]. Different data transport protocols are expected to be operable on the DTN stack to provide specific services depending on individual need from the applications [3]. As one of the main transport protocols of DTN, the Licklider transmission protocol (LTP) [4, 5] is intended to provide reliable data delivery service in interplanetary solar system Internet (SSI) [6].

For the operation of LTP, each data *block* is fragmented into data *segments* for transmission, with the last segment flagged as a checkpoint (CP) to check the arrival status of the entire block at the receiver. The CP segment is transmitted with a timer set (i.e., *CP Timer* [5]) and an acknowledgment (ACK) is expected, termed as a report segment (RS). The CP segment is necessarily retransmitted after the CP timer expires. The setting of the CP timer is very important for data delivery efficiency and transmission performance of LTP for space communications because it determines when the lost segments should be retransmitted, especially with a presence of channel-rate asymmetry which introduces additional delay to the transmission of RSs. This also leads to an issue of the total number of transmission attempts (or retransmission attempts) taken for successful file delivery.

A series of studies have been done jointly by NASA's Jet Propulsion Laboratory (JPL), California Institute of Technology and other research groups for LTP [7-20], other reliable data transport protocols [21-24] and DTN's main protocol, bundle protocol (BP) [25-32]. Most of these studies focus on protocol design and performance evaluation of LTP and BP for DTN-based space networks and deep-space communications.

Some of the series of studies have been done on analytical performance modeling of LTP with the focus on memory variation dynamics, RTT modeling, and analytical analysis. However, no work has been done in analyzing its CP timer setting for energy-efficient file delivery in deep-space networks that are characterized by asymmetric channel rates. In [19], an analytical model is built for the CP timer setting in such a way that reliable file delivery of LTP is ensured with a minimum number of retransmission attempts in deep-space communications characterized by a long latency and highly asymmetric channel rate. By this, the setting may avoid unnecessary retransmissions of the blocks and lead to a reduction of the energy consumption of LTP for successful delivery of the entire file over a deep-space channel.

In this paper, a packet-level analysis based on realistic file transfer experiments using a PC-based testbed is presented to validate the model for the CP timer setting built in [19]. The experimental analysis presented in this paper is expected to be useful in understanding the importance and impact of the CP timer setting on the efficiency of LTP in space communications, especially for energy-efficient data delivery in space missions.

II. Analytical Modeling

In [19], an analytical model is built for the CP timer setting, defined as $T_{CP_{timer}}$, to determine the minimum number of retransmission attempts of CP segments needed for successful file delivery of LTP in deep-space communications characterized by highly asymmetric channel rates. The model is reiterated as

978-1-7281-8560-6/21/$31.00 ©2021 IEEE
DOI 10.1109/SMC-IT51442.2021.00020

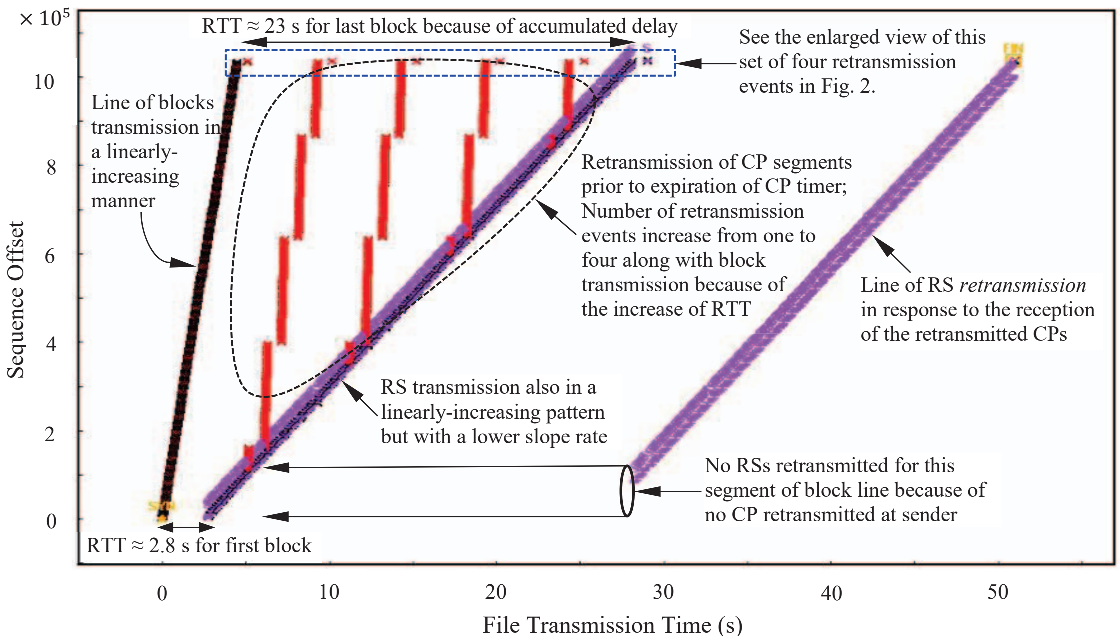

Fig. 1. A TSG at the sending node illustrating the LTP transmission at packet (or segment) level for delivery of a file of 1 Mbyte over an emulated cislunar communication channel with highly asymmetric channel rates (CR of 500/1) and a CP timer setting, T_{CP_timer}, configured to 4 s.

$$T_{CP\_timer} = \left[\frac{L_{Block\_Link}}{R_{Data}} + 2 \times T_{OWLT} + \frac{L_{RS}}{R_{RS}} + \left(\left\lceil \frac{L_{File}}{N_{Bundle} \times L_{Bundle}} \right\rceil - 1\right) \times \left(\frac{L_{RS}}{R_{RS}} - \frac{L_{Block\_Link}}{R_{Data}}\right)\right] \quad (1)$$

in which

L_{Block_Link} is the total length of an LTP data block when it reaches the link layer for transmission over the data channel,

R_{Data} is the downlink channel rate available for data transmission,

T_{OWLT} is the one-way-light-time, i.e., the one-way signal propagation time,

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

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

L_{File} is the size of a file in bytes that need to be delivered at the sender,

N_{Bundle} is the number of bundles to be aggregated within an LTP block, and

L_{Bundle} is the length of a bundle in bytes.

The total length of an LTP data block when it reaches the link layer, L_{Block_Link}, was derived in [19] as

$$L_{Block\_Link} = \frac{N_{Bundle} \times \left(L_{Bundle} + L_{Bundle\_Head}\right) \times \left(L_{Ltp\_Seg} + L_{Frame\_Head}\right)}{L_{Ltp\_Seg}} \quad (2)$$

in which

L_{Bundle_Head} is the header length of a bundle,

L_{Ltp_Seg} is the average length of a fragmented segment, and

L_{Frame_Head} is the total length of the overhead (starting from the LTP layer until the link layer) added to the segment to make it as a data frame.

For the detailed modeling process of both (1) and (2), refer to [19].

III. Overview of Experimental Infrastructure and Configurations

The model reiterated in Section II is validated through file transfer experiments using an experimental infrastructure, which is the PC-based space communication and networking testbed (SCNT) [7]. For a detailed description of the testbed, refer to [7]. The LTP implementation used for the experiments was adopted from the Interplanetary Overlay Network (ION) distribution v3.6.2 [36]. The ION is a software implementation of the DTN protocol suite for space networks and deep-space communications, developed by NASA's JPL. Each LTP segment is configured to be 1400 bytes, making it fit into an Ethernet frame MTU of 1500 bytes. As the performance of *reliable* data delivery service of LTP is concerned, all the data bytes of each LTP block are set as 100% red.

A one-way propagation delay of 1.35 s was introduced to each of the data and ACK channels. The effect of the channel-rate asymmetry on file transmission was implemented by configuring a high channel ratio (CR) of 500/1, having a downlink channel rate of 2 Mbps and uplink channel rate of 4 Kbps. In other words, the transmission rate of LTP data blocks is 2 Mbps, while the transmission rate of RS segments in the

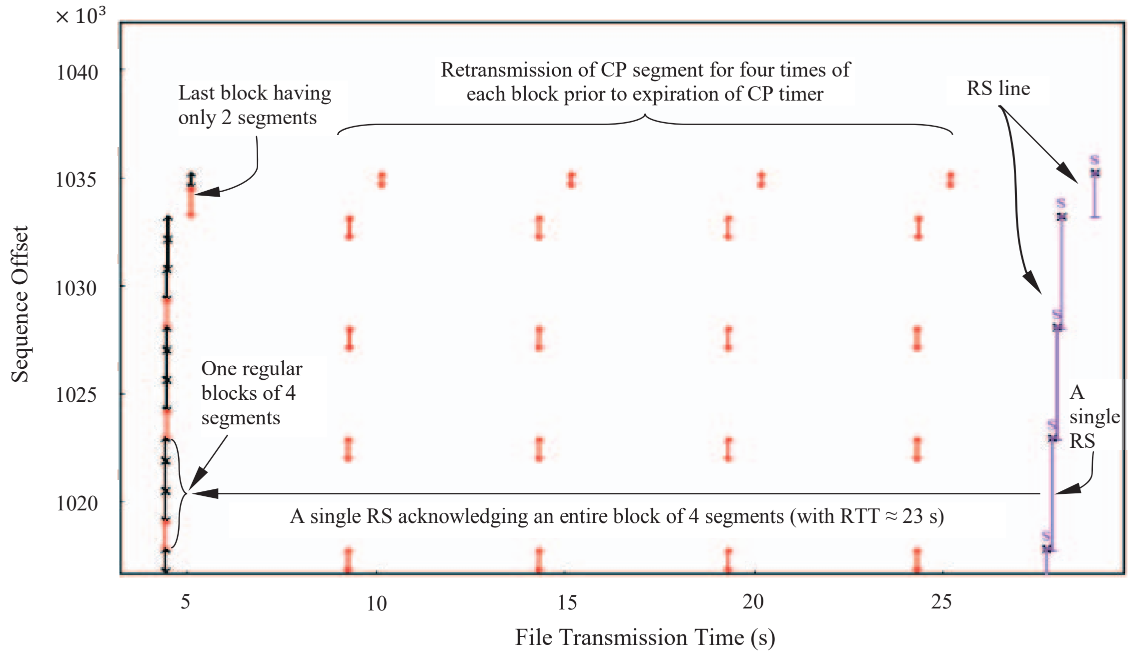

Fig. 2. An enlarged view of the TSG for transmission of the last five blocks of the file, illustrating four retransmission events of CP segments prior to expiration of CP timer.

opposite direction is 4 Kbps. The performance and packet-level traffic of LTP are collected from transmitting a text file of 1 Mbyte by running the protocol over the testbed.

IV. Analysis of Experimetnal Results and Model Validation

In this section, the LTP transmissions with different CP timer setting are first analyzed at the packet (or segment) level using the time sequence graph (TSG) [37]. Then, a comparison of the goodput performance among multiple CP timer settings and different channel loss rates is presented.

A. Analysis of LTP Transmissions at Packet Level

In Fig. 1, a TSG is presented to illustrate the LTP transmission at the packet (or segment) level at the sender for delivery of a file of 1 Mbyte over an emulated cislunar communication channel with highly asymmetric channel rates (CR of 500/1) and BER of 0 experienced. Five bundles with each of 1000 bytes are aggregated within a block. Provided that a one-way link delay of 1.35 s is configured, T_{CP_timer} is set to 4 s which is greater than the estimated RTT.

It is observed from Fig. 1 that the entire file of 1 Mbyte is transmitted in a linearly increasing time sequence of data segments with continuous LTP block transmission. The corresponding RSs from the data receiver are also transmitted in a linearly-increased pattern. However, the slope rate of the RS transmission line is obviously much lower. This happens because the transmission of the RS segments experiences delay due to the longer transmission time of an RS for each block introduced by the much lower ACK channel rate. The delay is actually accumulated with the transmission of the blocks, leading to varying lengths of RTT.

In fact, after the transmission of about the first 100 KB of the file is completed, the RTT experienced for the blocks starts to exceed the configured length of T_{CP_timer}, 4 s. This means that for those blocks transmitted following the first 100 KB, their RS segments are not received by the sender prior to the expiration of their CP timer, leading to retransmissions. Because the RTT experienced for the blocks is getting longer, the retransmission events of the CP segments increase along with transmission time, as shown in Fig. 1. While it is only retransmitted one time for the sets of blocks following the first 100 KB, the CP segments are retransmitted four times for the last set of blocks. These retransmission events are shown as vertical line segments in parallel between the data block line and the RS line.

The details of the retransmission events at the packet level can be seen from Fig. 2, which shows an enlarged view of the TSG for transmission of the last four blocks of the file. The block's aggregation of bundles, together with the acknowledgment by a single RS, can also be observed in Fig. 2. While the first three blocks have the same length, the last block (with a sequence offset around 1,034,000) is much shorter, as it conveys the last piece of the entire file. This can be easily explained. Given a bundle size of 1000 bytes, the size of each aggregated block is 5000 bytes because the experiment is configured to have five bundles aggregated within each block. With each segment configured to be around 1400 bytes , an individual block is divided into four segments for transmission according to $\left\lceil \frac{5000}{1400} \right\rceil$, with the last segment conveying fewer data bytes. The last block is divided into only two segments (with the second much shorter) because

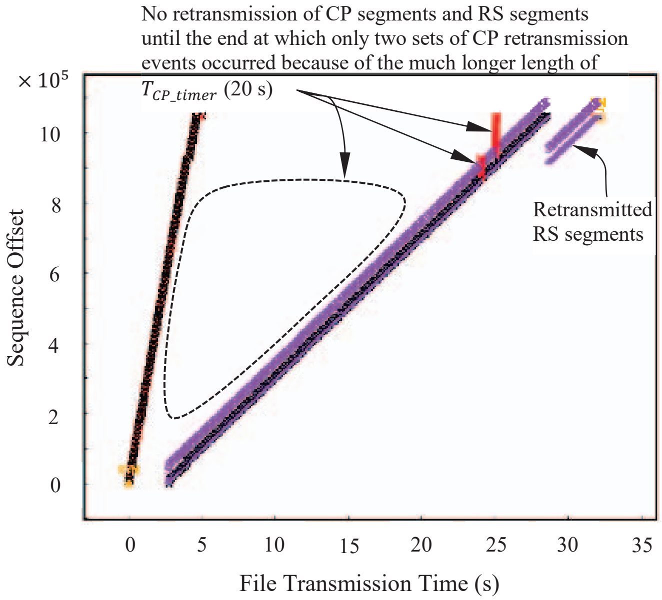

Fig. 3. A TSG at the sending node illustrating the LTP transmission at the packet (or segment) level for delivery of a file of 1 Mbyte over a cislunar communication channel with highly asymmetric channel rates (CR of 500/1) and the CP timer setting, T_{CP_timer}, configured to 20 s.

of its shorter length. For each of all four blocks, it is acknowledged by a single RS segment.

It is observed in Fig. 2 that the time gap (over the *x*-axis) between each block and its corresponding RS segment is around 23 s. This is actually the measured RTT interval for these blocks. This RTT interval is extremely long with respect to the configured one-way propagation length of 1.35 s for which the CP segment is retransmitted four times, as observed. The extremely long RTT interval occurs because of the linearly growing delay experienced by the RS segments, and it is accumulated for the last set of blocks, as previously discussed.

As an RS is required in response to reception of each CP segment, the first retransmitted CP segments result in the retransmission of an RS from the receiver for each block having a RTT longer than the CP timer setting. A series of these retransmitted RS segments forms another RS transmission line parallel to the original RS line, as shown in Fig. 1. These RSs are only retransmitted after the transmission of all the original RS segments is completed because they all have to wait to be sent out at the receiver. That is why retransmission of the RS segments starts at around 28 s when their original transmission ends. The retransmission of a large number of RS segments over the already overloaded, constrained ACK channel makes the situation worse, leading to even lower transmission efficiency of LTP.

Note that the RS retransmission line is not formed for the blocks conveying the first 100 KB of data (i.e., those having a sequence offset less than 100,000) in Fig. 1. This is because these blocks have no CPs retransmitted, as their RS segments are received prior to the expiration of the CP timer, as previously discussed.

If the length of T_{CP_timer} is configured to be much longer, the file transmission is quite different. Fig. 3 illustrates a TSG reiterated for the same LTP transmission in Fig. 1, but with T_{CP_timer} configured to be 20 s. In comparison, the RS transmission line and trend are similar to those for the original RS transmissions in Fig. 1. However, because of the much longer T_{CP_timer}, there is no retransmission of CP segments observed until the end, at which time only two sets of CP retransmission events occurred. While the entire file delivery time is slightly over 30 s, these two sets of retransmission events occurred around 25 s.

The retransmissions of the CP segments illustrated in Fig. 3 are for the data blocks corresponding to the time sequence (offset) numbers of around 900,000 and beyond. The other blocks of the file (i.e., the data bytes with the time sequence (offset) numbers smaller than 900,000) have no CP segments retransmitted, even though the transmission of the RS segments experiences delay, which increases along with the transmission of the blocks. This happens because their RS segments arrive at the sender prior to the expiration of the CP timer. In other words, even though with the RS delay involved, the CP timer setting is longer than the resulting length of the RTT for those blocks.

When the transmission approaches the end of the file, the accumulated RS delay is very long, which results in an RTT length longer than the CP timer setting. As a result, the CP timer expires before the RS is received, and therefore, the CP segment is retransmitted. Similar to those observed in Fig. 1, the retransmitted CP segments result in acknowledgment in the form of retransmitted RSs from the receiver. Those retransmitted RSs are illustrated in Fig. 3 as a short RS transmission line at the end of the file transmission (around 30 s). As in Fig. 1, it is shown as a separated RS line because those RSs have to wait to be retransmitted until the transmission of all the original RSs is complete.

With the file size of 1 Mbyte and aggregated block size of 5000 bytes configured for the experiments, the number of LTP blocks that the file is divided into for transmission is 200. The size of the encapsulated block at the link layer is 5309 bytes according to (2). With these two important numerical values and other numerical settings plugged into (1), it is calculated that an optimal length of the CP timer setting for the file transfer is 27 s. In other words, the CP timer length of 27 s is expected to avoid retransmissions of the CP segments for each block that experiences excessive delay for the RS due to the channel-rate asymmetry.

The file transfer experiment is also run with T_{CP_timer} configured to be 27 s and higher. The corresponding TSG indicates that the data transmission line and RS transmission line and their variation trends are almost the same as those shown in Fig. 3. The only difference is that there is no retransmission of CP segments (or RSs) observed over the entire course of file transmission of 1 Mbyte. Therefore, its TSG is not presented here. The difference occurs because for all blocks of the file, the corresponding RSs are received by the sender before their CP timers expire, which is credited to the longer setting of T_{CP_timer}, 27 s. The packet-level analysis based on the experimental file transfer conducted with three CP timer settings indicates that the analytical model (i.e., (1) in Section II) is validated.

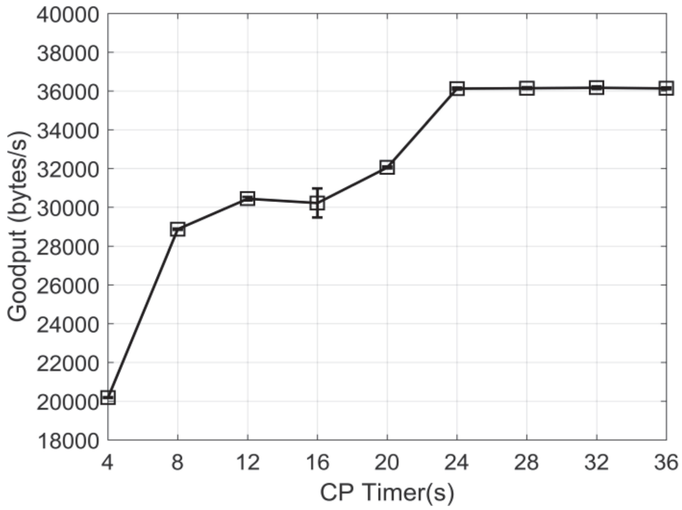

Fig. 4. A comparison of the experimented goodput performance of LTP in reliable delivering a 1-Mbyte file over a space communication channel with highly asymmetric channel rates (CR of 500/1) and various settings of the CP timer.

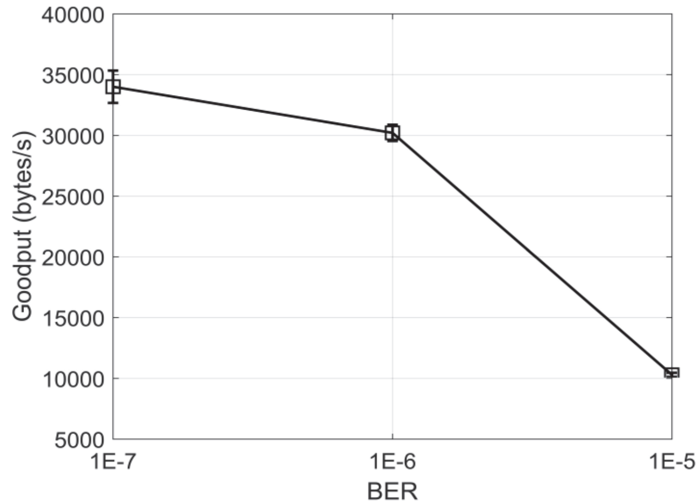

Fig. 5. Experimental goodput performance of LTP in transmitting a 1 Mbyte file over a cislunar communication channel with highly asymmetric channel rates (CR of 500/1) and the optimal CP timer setting T_{CP_timer}, 27 s, with different channel qualities that are equivalent to BERs of 10^{-7}, 10^{-6}, and 10^{-5}.

B. *Goodput Performance of LTP with Respect to Variations of CP Timer Setting and Channel Quality*

Fig. 4 presents a comparison of the goodput performance of LTP measured from the experiments in transmitting the same 1 Mbyte file with different setting of the CP timer over a broad range of 4-36 s. It is observed that, overall, the goodput of the protocol increases with an increase of the CP timer setting until it approaches the calculated optimal CP timer setting, 27 s. The performance improvement is significant in the region of short CP timer setting (4-24 s). However, there is no obvious performance variation observed for transmissions with a longer timer. This is reasonable, as discussed, when configured with a short CP timer setting, the resulting frequent retransmission events for the CP segments lead to an increase of the file delivery time. With the increase of the CP timer setting, the number of retransmission events drops, which leads to a decrease in the file delivery time and therefore, improvement of the goodput performance.

For transmissions with a CP timer setting around 27 s and longer, the performance is almost the same. This is because the CP timer setting already reaches an optimal setting of 27 s, and is already large enough to avoid any possible retransmission of the CP segments for each block within the individual RTT interval. Any increase of the CP timer setting does not further reduce retransmission events of the CP segments. In this case, any increase of the CP timer setting does not shorten the file delivery time and therefore, does not bring additional advantages with respect to the goopdut for file delivery. In comparison, the performance advantage with the optimal CP timer setting over the experimental shortest timer setting of 4 s is around 16 Kbytes/s which is indeed significant.

The comparison of the goodput performance for LTP presented in Fig. 4 is for transmissions over an emulated clean deep-space channel, i.e., with a BER of 0. It is important to have an understanding of how the channel quality (loss rate or simply, noise) affects the performance of LTP given that the CP timer setting is configured to be optimal. Fig. 5 illustrates the experimented goodput performance of LTP in transmitting the file with T_{CP_timer} of 27 s at different channel qualities that are equivalent to BERs of 10^{-7}, 10^{-6}, and 10^{-5}.

It is observed in Fig. 5 that the goodput performance of LTP degrades with an increase of the channel BER, i.e., with a reduction of the channel quality. The goodput shows a significant drop from 30 Kbytes/s to 10 Kbytes/s as the channel BER increases from 10^{-7} to 10^{-6}. It is obvious that the rate of goodput change of LTP increases as the BER varies in a higher range (i.e., over lossier channels).

The performance variations along with the change of the channel quality observed in Fig. 5 are reasonable. With a channel BER configured for the file transmission, a certain number of data bytes of the file are corrupted during transmission. Since all the data bytes are organized as LTP segments (and then data frames) for transmission, an individual segment that conveys any of these corrupted bytes are corrupted (or simply, lost). This leads to a delivery failure for the entire segment. Depending on the BER, the number of data corruption events (and therefore the number of corrupted segments) experienced is significantly different even with a given file size for transmission.

The corrupted segments must be retransmitted for successful delivery of the entire file. Therefore, for transmission with a high BER, which results in more corruption events, more retransmission events are required than with a low BER. More retransmission events—each in itself subject to corruption—lead to even more transmission rounds. As discussed above, corrupted segments are only retransmitted upon receipt of the RS for the block containing the corrupted segment. In other words, it takes an entire RTT interval to receive the acknowledging RS segment at the data sender. Therefore, given a file size, transmission with a high BER leads to longer file delivery time and thus, lower goodput. This explains the significant performance degradation with an increase of BER from 10^{-7} to 10^{-5} in Fig. 5.

The situation is worse if the segment conveying the CP flag is corrupted. In that case, it takes the sender much longer to resend a corrupted CP segment. This is because the sender has to wait until expiration of the CP timer to learn the loss of a segment. After the CP segment is resent, the sender has to wait again for the corresponding RS segment to arrive as a confirmation of successful delivery of the resent CP segment. This takes much longer, which is counted toward total file delivery time and therefore, causes a big drop in the goodput performance.

V. CONCLUSIONS

The analytical model built in the previous work for the optimal setting of the CP timer is validated by running realistic file transfer experiments over a PC-based testbed and analyzed at the data-packet level. It was found that if file transmission is configured with a CP timer setting shorter than the derived optimal one, the significantly reduced ACK channel rate introduces additional delay to the transmission of the RS segments. The delay grows linearly because it is accumulated along with the transmission of the blocks, leading to significantly increased block RTT.

With an optimal CP timer setting or greater configured for transmission, the RS segments are received from the sender before their CP timer expires. This leads to removal of all the retransmitted CP and RS segments (occurred with a shorter CP timer) and significantly reduces unnecessary traffic over both data and ACK channels. This is especially important for efficient usage of constrained (asymmetric) channel-bandwidth resources in a scenario characterized by highly asymmetric channel rates. With respect to the goodput performance of LTP transmission, it increases with an increase of the CP timer setting until the derived optimal length. It is concluded that the derived optimal CP timer setting is optimal with respect to various aspects of transmission performance of LTP, including file delivery time, goodput, and therefore, energy efficiency.

VI. FUTURE WORK

It is widely recognized that link disruption is inevitable in space communication networks, especial in deep-space communications that more likely experience link outage because of planet rotations, spacecraft rotations, and even solar storm. Some studies on analytical understanding of the effect of link disruption on DTN's BP have been done [33-35]. Although a series of studies have been done for LTP in space networks, how the evitable link disruption in space affects the performance for reliable data delivery of LTP is missing. In other words, there is a lack of a solid (analytical) understanding of the effect of link disruption on LTP, especially in deep-space communications. A detailed study of this effect is left as the major future work.

REFERENCES

[1] S. Burleigh, A. Hooke, L. Torgerson, K. Fall, V. Cerf, R. Durst, K. Scott, and H. Weiss, "Delay-tolerant networking: An approach to inter-planetary Internet," *IEEE Communications Magazine*, vol. 41, No. 6, pp. 128–136, Jun. 2003.

[2] The Space Internetworking Strategy Group (SISG), "Recommendations on a strategy for space internetworking," IOAG.T.RC.002.V1, Report of the Interagency Operations Advisory Group, NASA Headquarters, Washington, DC 20546-0001, USA, August 1, 2010.

[3] 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.

[4] S. Burleigh, M. Ramadas, and S. Farrell, "Licklider Transmission Protocol-Motivation," IRTF Internet Draft, Oct. 2007.

[5] M. Ramadas, S. Burleigh, and S. Farrell, "Licklider Transmission Protocol Specification," Internet RFC 5326, Sept. 2008.

[6] Consultative Committee for Space Data Systems, "Solar system internetwork (SSI) architecture," CCSDS 730.1-G-1. Green Book. Issue 1. Washington, DC, USA: CCSDS, July 2014.

[7] R. Wang, S. Burleigh, P. Parik, C-J Lin, and B. Sun, "Licklider Transmission Protocol (LTP)-based DTN for cislunar communications," *IEEE/ACM Transactions on Networking*, vol. 19, No. 2, April 2011, pp. 359-368.

[8] R. Wang, Z. Wei, V. Dave, B. Ren, Q. Zhang, J. Hou, and L. Zhou, "Which DTN CLP is best for long-delay cislunar communications with channel-rate asymmetry?", *IEEE Wireless Communications*, vol. 18, No. 6, December 2011, pp. 10-16.

[9] X. Sun, Q. Yu, R. Wang, Q. Zhang, Z. Wei, J. Hu, and A. Vasilakos, "Performance of DTN Protocols in Space Communications", *ACM/Springer Wireless Networks* (WINET), vol. 19, No. 8, November 2013, pp. 2029-2047

[10] J. Hu, R. Wang, X. Sun, Q. Yu, Z. Yang, and Q. Zhang, "Memory dynamics for DTN protocol in deep-space communications", *IEEE Aerospace and Electronic Systems Magazine*, vol. 29, No. 2, February 2014, pp.22-30

[11] K. Zhao, R. Wang, S. Burleigh, M. Qiu, A. Sabbagh, and J. Hu, "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.

[12] Q. Yu, S. Burleigh, R. Wang, and K. Zhao, "Performance Modeling of LTP in Deep-Space Communications," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, No. 3, July 2015, pp. 1609-1620.

[13] R. Wang, X. Wu, T. Wang, X. Liu, and L. Zhou, "TCP Convergence Layer-based operation of DTN for long-delay cislunar communications," *IEEE Systems Journal*, vol. 4, No. 3, September 2010, pp. 385-395.

[14] Q. Yu, R. Wang, K. Zhao, W. Li, X. Sun, J. Hu, and X. Ji, "Modeling RTT for DTN Protocols over Asymmetric Cislunar Space Channels," *IEEE Systems Journal*, vol. 10, No. 2, June 2016, pp. 556-567.

[15] L. Shi, J. Jiao, A. Sabbagh, R. Wang, Q. Yu, J. Hu, H. Wang, S. Burleigh, and K. Zhao, "Integration of Reed-Solomon Codes to Licklider Transmission Protocol (LTP) for Space DTN," *IEEE Aerospace and Electronic Systems Magazine*, vol. 32, No. 4, April 2017, pp. 48-55.

[16] G. Yang, R. Wang, S. Burleigh, and K. Zhao, "Analysis of Licklider Transmission Protocol (LTP) for Reliable File Delivery in Space Vehicle Communications with Random Link Interruptions," *IEEE Transactions on Vehicular Technology*, vol. 68, No. 4, April 2019, pp. 3919-3932.

[17] R. Wang, Z. Wei, Q. Zhang, and J. Hou, "LTP Aggregation of DTN Bundles in Space Communications," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, No. 3, July 2013, pp.1677-1691.

[18] Z. Yang, R. Wang, Q. Yu, X. Sun, M. Sanctis, Q. Zhang, J. Hu, and K. Zhao, "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.

[19] Y. Zhou, R. Wang, K. Zhao, and S. Burleigh, "A Study of Cross-Layer BP/LTP Data Block Size in Space Vehicle Communications over Lossy and Highly Asymmetric Channels," *IEEE Transactions on Vehicular Technology*, vol. 69, No. 12, December 2020, pp. 16126-16141.

[20] R. Lent, “Analysis of the block delivery time of the Licklider transmission protocol,” *IEEE Transactions on Communications*, vol. 67, No. 1, January 2019, pp. 518-526.
[21] R. Wang, B. Gutha, S. Horan, Y. Xiao, and Bo Sun, “Which transmission mechanism is best for space Internet: window-based, rate-based, or a hybrid of the two?” *IEEE Wireless Communications,* vol. 12, No. 6, December 2005, pp. 42-49.
[22] R. Wang and S. Horan, “Protocol testing of SCPS-TP over NASA’s ACTS asymmetric links,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 45, No. 2, April 2009, pp. 790-798.
[23] R. Wang and S. Horan, “The Impact of Van Jacobson Header Compression on TCP/IP throughput performance over lossy space channels,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, No. 2, April 2005, pp. 681-692.
[24] R. Wang, B. Gutha, and Paradesh Kumar Rapet, “Window-based and rate-based transmission control mechanisms over space-Internet links,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 44, No. 1, January 2008, pp. 157-170.
[25] R. Wang, M. Qiu, K. Zhao, and Y. Qian, “Optimal RTO Timer for Best Transmission Efficiency of DTN Protocol in Deep-Space Vehicle Communications,” *IEEE Transactions on Vehicular Technology*, vol. 66, No. 3, March 2017, pp. 2536-2550.
[26] A. Sabbagh, R. Wang, K. Zhao and D. Bian, "Bundle Protocol Over Highly Asymmetric Deep-Space Channels," *IEEE Transactions on Wireless Communications*, vol. 16, no. 4, April 2017, pp. 2478-2489.
[27] C. Feng, R. Wang, Z. Bian, T. Doiron, and J. Hu, “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.
[28] K. Zhao, R. Wang, S. Burleigh, A. Sabbagh, W. Wu, and M. D. Sanctis, “Performance of Bundle Protocol for Deep-Space Communications,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 52, No. 5, October 2016, pp. 2347-2361.
[29] R. Wang, A. Sabbagh, S. Burleigh, K. Zhao, and Y. Qian, “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.
[30] G. Wang, S. Burleigh, R. Wang, L. Shi, and Y. Qian, “Scoping Contact Graph Routing Scalability,” *IEEE Vehicular Technology Magazine*, vol. 11, No. 4, December 2016, pp. 46-52.
[31] Q. Yu, X. Sun, R. Wang, Q. Zhang, J. Hu, and Z. Wei, “The effect of DTN custody transfer in deep-space communications”, *IEEE Wireless Communications*, vol. 20, No. 5, October 2013, pp.169-176.
[32] G. Yang, R. Wang, A. Sabbagh, K. Zhao, and X. Zhang, “Modeling Optimal Retransmission Timeout Interval for Bundle Protocol,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 54, No. 5, October 2018, pp. 2493-2508.
[33] A. Sabbagh, R. Wang, S. C. Burleigh and K. Zhao, "Analytical Framework for Effect of Link Disruption on Bundle Protocol in Deep-Space Communications," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 5, May 2018, pp. 1086-1096.
[34] R. Wang, A. Sabbagh, S. C. Burleigh, M. Javed, S. Gu, J. Jiao, Q. Zhang "Modeling Disruption Tolerance Mechanisms for a Heterogeneous 5G Network," *IEEE Access*, vol. 6, 2018, pp. 25836-25848.
[35] L. Yang, R. Wang, Y. Zhou, X. Liu, K. Zhao and S. C. Burleigh, “Hybrid Retransmissions of BP for Reliable Deep-Space Vehicle Communications in Presence of Link Disruption,” *IEEE Transactions on Vehicular Technology*, vol. 70, No. 5, May 2021, pp. 4968-4983.
[36] S. Burleigh, “Interplanetary overlay network design and operation v3.6.2,” JPL D-48259, Jet Propulsion Laboratory, California Institute of Technology, CA, March 2020, [Online]. Available: http://sourceforge.net/projects/ion-dtn/files/latest/download.
[37] Ltptrace Analysis Tool: LTPTRACE, [Online] http://ion.ocp.ohiou.edu/content/ltptrace