Parametric Analysis of the Bridge Architecture

Kashyap, Shruthi; Rao, Vijay; Venkatesha Prasad, Ranga Rao; Staring, Toine

DOI
10.1007/978-3-030-85836-0_7

Publication date
2021

Document Version
Final published version

Published in
SpringerBriefs in Applied Sciences and Technology

Chapter 7
Parametric Analysis of the Bridge Architecture

Chapter 5 discussed how the TCP RTT and RTO parameters can be adapted to the cordless kitchen system. The results after the adaptation were shown in Chap. 6. This chapter focuses on analyzing the TCP MSS and CWND parameters, and also other factors that affect the latency of the system. Simulations and theoretical calculations have been made to analyze the effects of parameters like the NFC bit error rate, communication time-slot size, frequency of non-TCP/IP messages over the NFC channel, etc.

7.1 Effect of TCP CWND Size and Slow Start Process on the System Latency

The TCP congestion control mechanism controls the maximum amount of data a sender can transmit before receiving an ACK from the receiver. The sender maintains a Contention Window (CWND) to keep track of this. The TCP congestion control consists of the slow start and congestion avoidance mechanisms, as shown in Fig. 7.1. The TCP slow start mechanism starts with an initial minimum CWND size and increases the window size by 1 MSS for every ACK received, until the slow start threshold (ssthresh) is reached. The congestion avoidance mechanism then takes over and gradually increases the window size until the network's capacity is reached or until a packet loss occurs. If it encounters a packet loss, the slow start process starts over with the minimum CWND size and with ssthresh set to half of the current CWND, as shown in Fig. 7.1 (Note: Refer to [1] for a detailed explanation on the working of the TCP/IP protocol).

It can be hypothesized that if the initial CWND is very small, then the latency of the TCP session would increase as the slow start process would take longer to

S. Kashyap et al., *Cook Over IP*,
SpringerBriefs in Applied Sciences and Technology,
https://doi.org/10.1007/978-3-030-85836-0_7

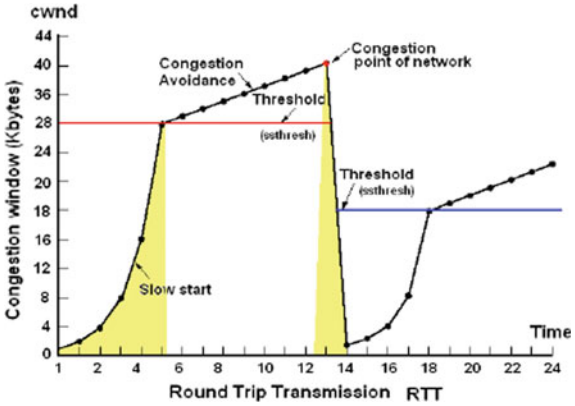

Fig. 7.1 TCP slow start and congestion avoidance mechanisms (*Source* [2])

Table 7.1 LwIP configuration for the TCP CWND experiments

TCP MSS	1024 bytes
Initial CWND size	4096 bytes
Maximum CWND size	8192 bytes
Sender buffer size	8192 bytes
NFC bit rate in slotted mode	11.2 kbps

reach the maximum window limit, making the channel idle for a significant amount of time. This hypothesis is tested with the LwIP configuration given in Table 7.1, and with a data transfer of 50 kB from the end-user device to the appliance. In the slow start process, for every ACK received, the CWND increases by 1 MSS. In case of packet loss, the ssthresh is set to half of the current CWND and the slow start process begins with initial CWND size of 1 MSS. This experiment is performed to find the optimum initial CWND size for the system such that there is minimum latency considering the congestion on the Ethernet/Wi-Fi channels.

Figure 7.2 shows the result of transferring 50 kB of data for different initial CWND sizes of 2048 bytes, 4096 bytes and 8192 bytes. It can be noticed that the difference in latency between the two extreme sizes is only about 76.07 ms. This implies that choosing a higher initial CWND size will not give a significant improvement in performance. The reasons for this behavior are explained below.

1. The NFC channel is half-duplex and has a low bandwidth in the time-slotted mode. So the bandwidth utilization of the NFC channel will be already high considering the small delay on the Ethernet channel and the high speed of the TCP/IP stacks. The data packets are almost always available to the NFC module unless the initial CWND is less than 2*MSS. Therefore, the reduction in the latency obtained by opting for a high initial CWND size will be very insignificant, as the bandwidth of the NFC channel cannot be improved further by a large factor.

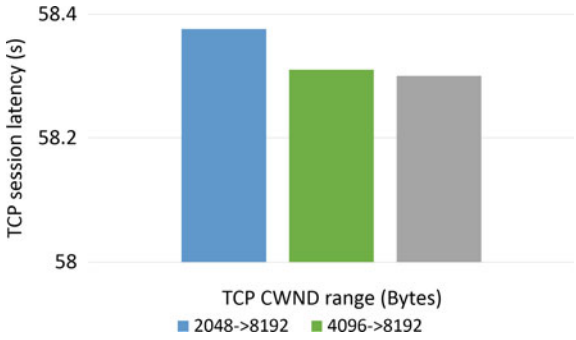

Fig. 7.2 TCP session latency for different initial CWND sizes

2. If the delay on the Ethernet/Wi-Fi channel is higher than or comparable to that of the NFC, the effects of a small initial CWND can be noticed. This is because the NFC channel may sometimes be idle when the packet is slowly traveling over the Ethernet/Wi-Fi channel. In this case, if it is made sure that there is at least one packet available at the NFC module at any point in time, it is possible to achieve maximum bandwidth utilization. So it is not necessary to always go for the maximum initial CWND size.

The experiment is repeated by varying the delays on the Ethernet channel to check if smaller initial CWND increases the latency by a significant amount. Figure 7.3 shows the results for Ethernet delays of <1 ms, 250 ms, 500 ms and 1 s, for different initial CWND sizes. It can be noticed that as the delay on the Ethernet increases, the difference in the latencies between the maximum and minimum initial CWND sizes increases. At an Ethernet delay of 1 s, there is a 1.08 s difference in the overall latency. This is not a very high gain in the performance though. Furthermore, the latencies with initial CWND sizes of 4096 bytes and 8192 bytes are almost the same. This implies that the NFC bandwidth utilization reaches the maximum with the initial CWND size of 4096 bytes. Any further increase will not result in any improvement.

Figure 7.4 shows the goodput graphs of the TCP sessions for different initial CWND sizes for a data transfer of 50 kB. It is interesting to see that no matter what the initial CWND size is, the goodput eventually comes to be 1 kbps or 8 kbps, which is the maximum achievable goodput on the NFC channel at 11.2 kbps, considering the NFC chunk size of 14 bytes with 10 bytes of usable payload size. This result supports the fact that the size of the TCP CWND does not have much effect on the throughput of the cordless kitchen system.

The TCP slow start process takes place only at the beginning of the TCP session if no packet loss is observed. So as long as there are no retransmissions, the effect of a small CWND may not be noticed in the system. This may not be true when retransmissions are taken into account, because every retransmission triggers the slow start process, making the TCP start over with a small initial CWND size. This could affect the overall latency. To verify this hypothesis, an experiment is designed where the end-user transfers 100 kB of data to the appliance, and the channel is lossy, where one out of 25 packets is lost. Ethernet delay of <1 ms is considered

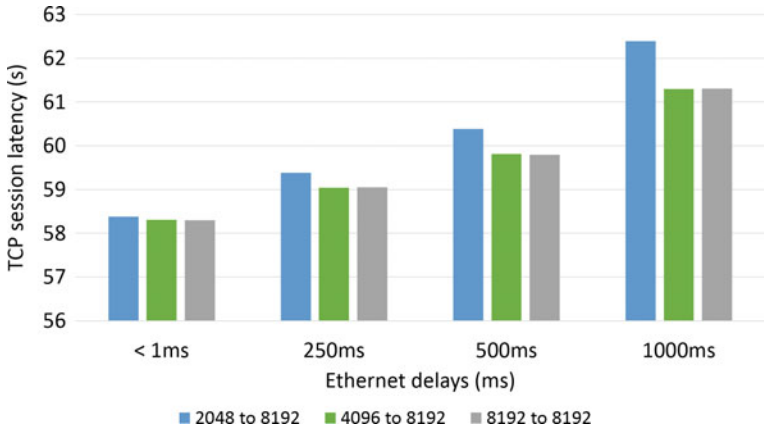

Fig. 7.3 TCP session latency for different initial CWND sizes and Ethernet delays

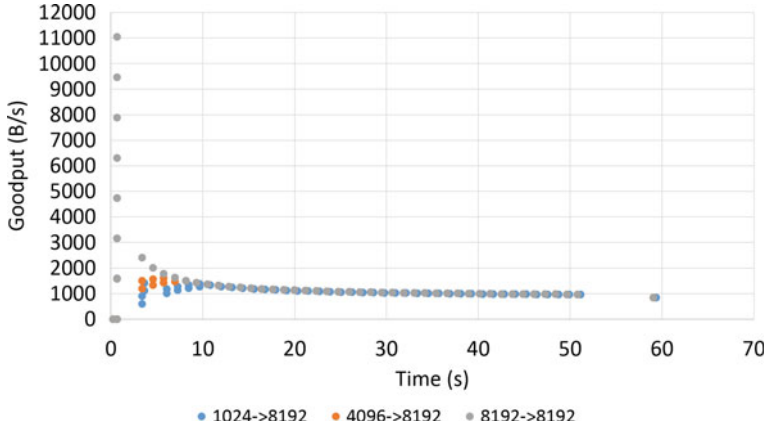

Fig. 7.4 Goodput of the TCP session for different initial CWND sizes for 50 kB data transfer

for this experiment. Figure 7.5 shows the latency for different initial CWND sizes. A reduction of up to 7.5% can be achieved when a bigger CWND size is chosen. Therefore, it can be concluded that the size of the initial CWND does not have a significant effect on the latency of the cordless kitchen system. This is because the NFC channel has a very small bandwidth and has almost maximum utilization even with small window sizes. So larger CWND does not help in further increasing the utilization of the channel. It should be noted that the intention of the TCP slow start process is to avoid congestion in the channel. It is recommended not to choose very high CWND as it may aggravate the latency in lossy congested channels.

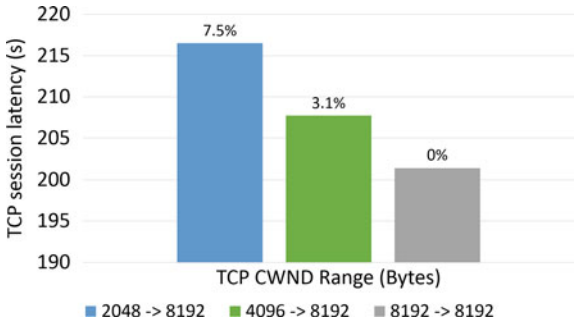

Fig. 7.5 TCP session latency for different initial CWND sizes over a lossy channel

7.2 Effect of TCP MSS Size on the System Latency

The TCP segments carry the actual data that is being transmitted. Choosing the right maximum segment size is very important to achieve minimum latency. If the MSS is very large, the size of the IP datagram will increase which may cause IP fragmentation reducing the efficiency of transmission. It may also increase the chance of the TCP segment getting lost. If the MSS size is too small, it would create more number of packets with very small data in each. In this case, the TCP/IP header overhead would become very prominent resulting in inefficient use of the channel bandwidth, thus increasing the latency.

Table 7.2 summarizes the average results of transferring 5 kB of data from the end-user device to the appliance using different TCP MSS sizes at 11.2 kbps. The results show that unless a very small MSS (<512 bytes) is chosen, the latency will not increase by a large number. A small MSS of 256 bytes increases the latency by 26.28%, however, choosing a size greater than or equal to 512 bytes increases the latency only by <10%. So an MSS value of 1024 bytes or greater would give a very high performance with minimal latency. It is important to note that in the case of an erroneous NFC/Wi-Fi channel with a high bit error rate (BER), large packets would be more susceptible to errors compared to smaller packets. So the TCP MSS should be chosen depending on the conditions of the channel in order to avoid retransmissions caused by packet errors.

Table 7.2 TCP session latency for different TCP MSS values

TCP MSS (Bytes)	1460	1024	512	256
TCP session latency (s)	6.24	6.29	6.78	7.88

7.3 Effect of NFC BER on the System Latency

The bit errors in the NFC channel would introduce errors in the TCP/IP packets being tunneled through the NFC channel causing the packets to be dropped by the TCP/IP stacks due to failing checksum. In the given setup, there is no error detection or correction mechanisms implemented in the NFC layer. So even a single bit error in the packet would lead to retransmission as the packet will be dropped. Therefore, the presence of bit errors in the NFC channel will have a huge impact on the system latency.

Bit errors in the NFC channel can be random or bursty. Random errors would lead to more number of packet drops as the errors are randomly distributed, which can affect any packet in the TCP session. On the contrary, the burst errors come as a block, so the errors would be confined to a single or a couple of packets depending on the size of the block and the time of occurrence. So the burst error would have less impact on the TCP session latency compared to random error. An experiment is designed to verify this hypothesis where the appliance and end-user device exchange 100 packets of 500 bytes each. Random and burst errors of 10^{-4}, 10^{-5} and 10^{-6} are introduced in the NFC channel to test the latency of the TCP session.

7.3.1 Random Errors

The random bit errors are modeled using the following formula:

$$P(0) = 1 - P(1) \tag{7.1}$$

where P(0) and P(1) denote the probabilities of transmitting bits 0 and 1 without errors.

7.3.2 Burst Errors

The burst errors are introduced based on the Gilbert–Elliott model as shown in Fig. 7.6. The states Good and Bad represent the bit error conditions. p and q are the transition probabilities between these states. The average burst length is taken as 4 bits in this experiment. An example below shows the steady state and transition probabilities of the Good and Bad states for an NFC BER of 10^{-4}.

$$\pi_{Good} = \frac{q}{q+p} \tag{7.2}$$

$$\pi_{Bad} = \frac{p}{p+q} \tag{7.3}$$

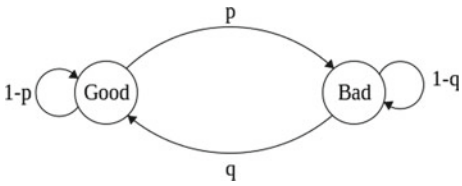

Fig. 7.6 Gilbert–Elliot error model for simulating burst errors

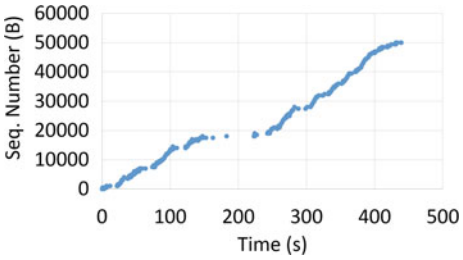

Fig. 7.7 TCP session with a random BER of 10^{-4}

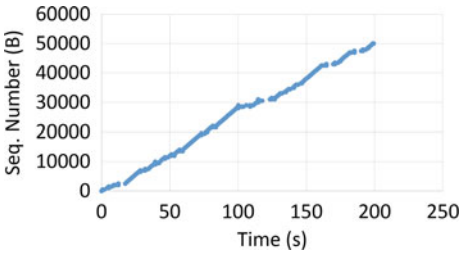

Fig. 7.8 TCP session with a burst BER of 10^{-4}

where $\pi_{Good} = 1 - 10^{-4}$ (steady state probability of state Good) and $\pi_{Bad} = 10^{-4}$ (steady state probability of state Bad). The transition probabilities p and q are calculated by taking Average Burst Length (ABL) as 4 bits. So q becomes 0.25 and p becomes $25 * 10^{-6}$.

Figures 7.7, 7.9 and 7.11 show the output of the TCP session with random NFC BERs of 10^{-4}, 10^{-5} and 10^{-6}, respectively, at 11.2 kbps. It can be seen that as the BER reduces, the number of retransmissions decreases and hence the TCP session latencies. The same behavior is observed with burst NFC BER as shown in Figs. 7.8, 7.10 and 7.12 at 11.2 kbps. As per the hypothesis, at a given NFC BER, fewer retransmissions are observed with burst errors compared to that with random errors. This proves that burst errors have less impact on the system latency as the errors come in bursts which affect fewer TCP/IP packets.

Table 7.3 summarizes the latencies of the TCP sessions with random and burst errors at different NFC BERs. At a BER of 10^{-4}, the latency with the burst error is around 54.56% less than that with random error. However, as the BER reduces, the difference in latency between the two types of errors reduces. At a BER of 10^{-6}, there is only about 1.8% difference in the session latencies. Therefore, it can be concluded that at lower BERs the type of error will not matter much but at higher BERs burst errors will have lesser impact on the overall latency.

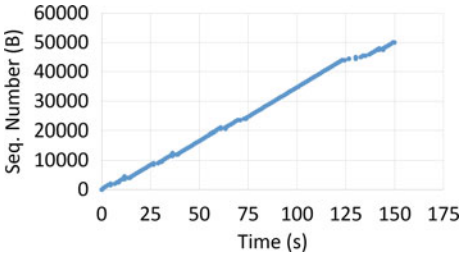

Fig. 7.9 TCP session with a random BER of 10^{-5}

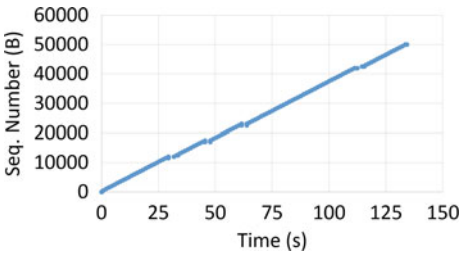

Fig. 7.10 TCP session with a burst BER of 10^{-5}

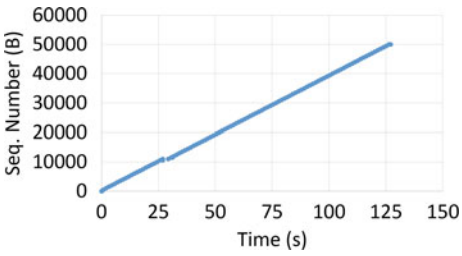

Fig. 7.11 TCP session with a random BER of 10^{-6}

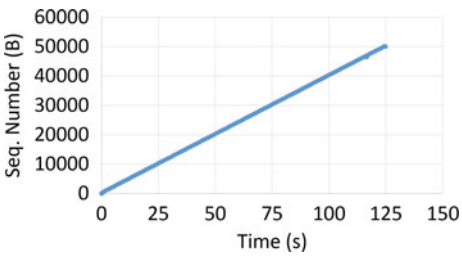

Fig. 7.12 TCP session with a burst BER of 10^{-6}

Table 7.3 TCP session latencies with random and burst errors at different NFC BERs

NFC BER	TCP session latency (s)	
	Random error	Burst error
10^{-4}	439.22	199.6
10^{-5}	150.07	134.46
10^{-6}	127.32	125.03

7.4 Effect of Varying the NFC Communication Time-Slot Duration on the System Latency

The cordless kitchen specification defines an NFC communication time-slot size of 1.5 ms in the time-slotted mode. If the size of the time slot is increased, the latency of the system can be reduced because more data can be transferred over bigger time slots. The decrease in latency with the increase in time-slot size will be non-linear because they are inversely proportional. An experiment is carried out to find out the optimum NFC time-slot size for the system such that minimum latency is maintained.

The standard payloads of NFC read and write commands, as defined in the cordless kitchen specification, are used in this experiment (refer to Sect. 2.5.2). A short TCP session exchanging 1 kB of data is considered, with NFC time-slot sizes varying from 1 to 2.5 ms. The latencies are theoretically calculated using RTT Eq. 5.1 as described in Sect. 5.3.2.1. Figures 7.13, 7.14 and 7.15 show the results of tunneling the TCP/IP packets over different time-slot sizes at 212 kbps, 424 kbps and 848 kbps, respectively. The graphs represent an inverse variation function. The rate at which the latency decreases with increasing slot size is steep at the beginning, and it gradually flattens out at higher slot sizes. This behavior is more noticeable at lower bit rates because the amount of data that can be sent over a time slot is small compared to that at higher bit rates. This implies that the choice of the correct time-slot size is more critical at lower bit rates for maintaining a reasonable latency.

In the cordless kitchen system, there is a trade-off between the efficiency of power transfer and communication. As the size of the time slot increases, the efficiency of the power transfer decreases. Moreover, bigger time-slot sizes would generate harmonics in the power signal leading to vibrations and heating in the PTx module. Therefore,

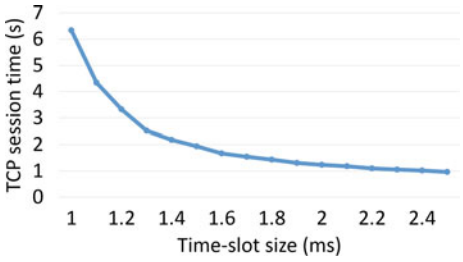

Fig. 7.13 Latencies of TCP sessions for different NFC time-slot sizes at 212 kbps

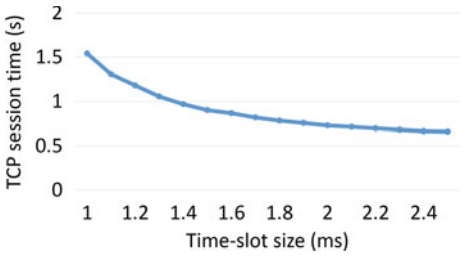

Fig. 7.14 Latencies of TCP sessions for different NFC time-slot sizes at 424 kbps

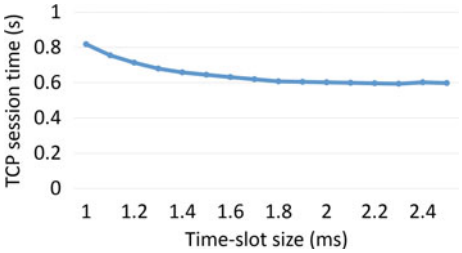

Fig. 7.15 Latencies of TCP sessions for different NFC time-slot sizes at 848 kbps

it is important to choose an optimum slot size such that both power transfer and data transfer are efficient. An optimum time-slot size needs to be chosen for the lowest NFC bit rate, which is 212 kbps in this case. This slot size would give a better performance at higher bit rates as well.

At 212 kbps, an efficiency of about 50% in data transmission can be achieved with a time-slot size of 1.5 ms. Higher efficiency with the same slot size can be achieved at higher bit rates. A slot size of 1.5 ms results in an efficiency of 75% at 424 kbps and about 95% at 848 kbps. An efficiency close to 99% can be achieved with a slot size of 1.9 ms at 848 kbps. Depending on the criticality of the Internet applications, appropriate slot size can be chosen such that desired efficiency is achieved at all data rates.

7.5 Considering Non-TCP/IP Messages over the NFC Channel

For ease of analysis, in all of the experiments the NFC channel was assumed to comprise only TCP/IP messages. However, in a real case scenario, the NFC channel would also carry other types of messages such as power control, negotiation, measurements and state transition. The frequency of these messages would depend on the type of application being used. An experiment is performed to analyze the performance of the TCP session in the presence of such messages at different frequencies of their occurrence.

The frequency of non-TCP/IP messages is taken as a fraction, for example, 2 slots every 10 slots used for other messages. This will impact the RTT of a TCP/IP packet over the NFC channel, i.e. the t_{NFC} in Eq. 5.4 will increase by the number of slots used by other messages while the packet is being transferred over the NFC channel. The modified version of Eq. 5.4 is as follows:

$$t_{NFC} = \frac{slots_{pckt}}{1 - freq_{ctrlmsgs}} * 10 \text{ (ms)} \tag{7.4}$$

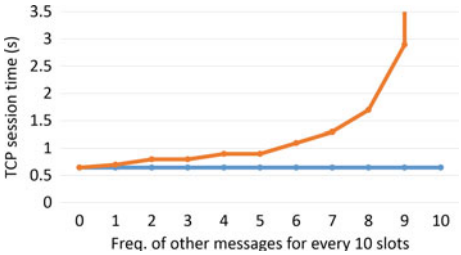

Fig. 7.16 Latencies of TCP sessions for different frequencies of non-TCP/IP messages at 848 kbps

where

$$freq_{ctrlmsgs} = \frac{a}{b} \tag{7.5}$$

a/b signifies 'a' slots every 'b' slots format. b is taken as section size and $b - a$ will be usable slots per section. The t_{NFC} has to be rounded up to the nearest section size b because the usable slots can occur anywhere in the section. As the frequency of other messages increases, the TCP session latency also increases. This increase will be non-linear because the number of usable slots varies inversely with latency. If $freq_{ctrlmsgs} = 0$, all slots will be available for TCP/IP packets and Eq. 7.4 will be the same as Eq. 5.4. If $freq_{ctrlmsgs} = 1$, i.e. all slots are used for other messages, then the t_{NFC} becomes ∞ which means that the TCP/IP messages cannot be transferred. A TCP session with 1 kB data exchange and a section size b of 10 slots is considered for the experiment. Theoretical calculations are made for TCP session latencies using Eq. 5.1 and for an NFC bit rate of 848 kbps.

The result of the experiment is depicted in Fig. 7.16. The graph shows an inverse variation function, so the rate of increase in latency will be steep as the frequency of non-TCP/IP messages increases. At lower frequencies, the latency varies slightly. It can be inferred from the results that efficiency of around 72% can be achieved in the transmission of TCP/IP messages at a frequency of 5/10. Appropriate frequencies can be chosen depending on the criticality of the Internet application.

References

1. *TCP/IP Illustrated* (3 Volume Set) by W.R. Stevens, G.R. Wright (2001) Hardcover (2021). Addison-Wesley Professional
2. G. Abed, M. Ismail, K. Jumari, A survey on performance of congestion control mechanisms for standard TCP versions. Aust. J. Basic Appl. Sci. **5**, 1345–1352 (2011)