

THE PERFORMANCE OF TCP IN A TACTICAL INTERNET*

Y. J. Liu

ITT Aerospace/Communications Division
100, Kingsland Road
Clifton, New Jersey 07014-1993

Abstract

Three issues concerning TCP [1,2] in a tactical internet are addressed in this paper. They are TCP connection procedure, TCP parameter optimization and its performance comparison with UDP. Extensive OPNET simulations with various net models and scenarios were performed. The results showed that initial retransmission timeout is a critical parameter and UDP with SAR (Segmentation and Reassembly) [3] outperforms TCP. A TCP connection procedure that is suitable to set up any number of connections based upon the need is also discussed.

1. Introduction

In a tactical internet, TCP is used as a possible way to send out messages from the user. It is a connection-oriented protocol with the capability of retransmissions and acknowledgments to enhance the reliability of transmitted messages. It can be used in the SINCGARS (Single Channel Ground-Airborne Radio Systems) net to send C2 messages. It can also be used in the EPLRS [4] (Enhanced Position Location Reporting System) network to set up connections among the BGP [5] peers.

In this paper three major issues concerning TCP are discussed. The first issue centers on the best way to set up the TCP connections. In a TCP protocol, two stations can communicate only if the connection is set up in advance. In a tactical internet, a station can receive messages from several sources. If a passive open is not set up in each station, messages are lost.

The second issue concerning TCP is its performance comparisons with UDP. UDP is a connectionless protocol. UDP by itself for reliable service does not seem to be a reasonable alternative to TCP. However, MIL-STD-2045-14502-1A specifies a Segmentation/Reassembly module to enhance the performance of UDP. This module does not just segment and reassemble the message, it also performs retransmissions and acknowledgments. A question is then which protocol, TCP or UDP, is better suited for the tactical internet.

The third issue concerning TCP is the determination of the best parameter values. Several parameters are used in TCP. They are initial retransmission timeout, R1, R2 [2] and other parameters to adaptively compute the retransmission timeout. Two questions need be answered. One is the determination of those parameters which are critical to TCP performance. The other is the determination of the optimum parameter values.

In order to answer these questions, OPNET simulations are performed with various net models and scenarios. At, and below, intranet layer, MIL-STD-188-220A is followed. Above Intranet layer, MIL-STD-2045-14502-1A is utilized. Both SINCGARS radio and EPLRS which are used in the tactical internet are simulated under the OPNET environment. In the following, each issue is discussed in detail together with the simulation results.

2. TCP Connections

A three-way hand shake is utilized in TCP to set up a connection. Under normal conditions, every station is initially set up with one passive open. A station wishing to transmit a message sends out an active open SYN to initiate the connection handshake. However, only one

* This work is supported under contract with PM TRACS

passive open can allow the station to receive one message from one source. In order to receive messages from several different sources, more than one passive open has to be set up. The question is then how a station knows the number of passive open connections to set up.

An exhaustive approach is for every station to set up all possible connections in a tactical internet. If there are n stations, a total of $n(n-1)/2$ passive open states have to be set up. There are two drawbacks to this approach. First, it is difficult for each user to know in advance the exact total number of stations. Second, it is a waste of memory to set up so many connections.

An alternative approach is to set up extra connections only if there is a need to do so. That is, initially, every station sets up only one passive open state. Therefore, a station has no problem in setting up the connection with the first source. Suppose station A sets up a connection with station B and receives another active open message from station C. Station B finds out it has connection with station A but no connection with station C. Station B then automatically adds a passive open with station C. The connection handshake then continues until the connection is complete. The advantage of this approach is that it is dynamic and connections can be set up with any number of stations limited only by the amount of physical memory.

3. TCP Optimization

The Optimum values of TCP parameters depend upon message traffic and net size. One of the major applications of TCP is to set up connections among BGP speakers using the EPLRS network. An area of a Brigade which has the most complicated EPLRS network is selected for parameter optimization.

EPLRS is an UHF radio network operating in a TDMA structure [4]. In the context of BGP petal, a host is modeled by a combination of INC (Internet Controller) and EPUU. The INC has three interfaces, two for the SINGARS radio and one for EPLRS. An INC acting as DTE utilizes the X.25 protocol to interface with EPUU which acts as DCE. Data packets are generated in the INC. They pass through either the TCP or UDP and then the IP layer before sending them through the X.25 interface to the EPUU. An EPUU communicates with another EPUU using the EPLRS network.

A total of 20 INCs is considered in the area 0 of the Brigade. Every EPUU is connected with multi-channel

needlines with neighboring EPUUs. There are different operating modes for each needline. The full-duplex mode is considered in the TCP optimization. Each EPLRS needline is modeled as a point-to-point link for OSPF [6]. If there are n needlines, then there are n point-to-point links. In the modeling, each point-to-point link is identified by a unique channel number. All the OSPF packets sent to these point-to-point links pass through the X.25 interface. Through the channel number, an EPUU can identify a point-to-point link after receiving the message from the INC.

The optimization concentrates on all message traffic inside the EPLRS needlines only. The distance between any two INCs ranges from one hop to 7 hops and the EPLRS needline has a transmission rate of 480 bps.

3.1 Message Traffic and Size

Basically, there are three different types of messages in the EPLRS needlines. There are C2, OSPF [6] and BGP messages. The C2 messages are sent through UDP and BGP messages are sent through TCP. The OSPF protocol is implemented in OPNET and can generate all the required traffic such as HELLO, link state request, update and acknowledgments [6]. BGP protocol is currently not implemented. However, the equivalent traffic is generated and passes through TCP.

There are three different types of BGP messages from the BGP speakers : OPEN, KEEPALIVE and UPDATE [5]. OPEN message is sent during startup to set up a BGP connection. Upon receiving OPEN message, a KEEPALIVE message is sent from the receiver. During steady state, it is assumed each BGP speaker sends out one KEEPALIVE message every 8 minutes and 3 UPDATE messages every hour. The C2 messages are sent randomly from any INC to a randomly selected destination. It is assumed there are a total of 240 C2 messages per hour.

Without including TCP and IP headers, the message size for KEEPALIVE, OPEN and UPDATE is 160 bits, 400 bits and 880 bits, respectively. For C2, the raw data size without including the protocol header is 3968 bits.

3.2 Optimization Parameters

There are seven TCP parameters selected for optimization. They are initial retransmission period, retransmission gain, retransmission deviation gain, retransmission deviation

coefficient, ACK delay, R1 and R2. Each parameter is briefly defined here.

Assume T and σ represent the round trip delay and its standard deviation. They are updated according to the following formula

$$T_{n+1} = (1-a) T_n + a T_{new}$$

$$\sigma_{n+1} = (1-b) \sigma_n + b \sigma_{new}$$

$$RTO = T_{n+1} + c \sigma_{n+1}$$

In the above formula, a is the retransmission gain used to smooth the round trip delay, b is the retransmission deviation gain used to smooth the standard deviation of the round trip delay, c is the retransmission deviation coefficient used to combine the round trip delay and its standard deviation to compute the retransmission period. The ACK delay is the time that must elapse before sending the ACK after receiving a message.

There are also two more parameters, R1 and R2. Parameter R1 is the maximum number of retransmissions before the application is warned. Parameter R2 is the maximum duration after R1 retransmissions. If R2 is reached, the connection is aborted. RFC 1122 requires R1 is set at 3 retransmissions and R2 is set at 4 minutes.

3.3 Optimization Criterion

The goal of optimization is to maximize the completion rate while minimizing the message delay. However, to reach this goal it is also necessary to minimize the bandwidth consumption due to the TCP traffic. The purpose is to allow more bandwidth to be used to transmit C2 data messages. Therefore, the optimization metric is:

$$m = c/(d*t)$$

where c is the completion rate, d is the delay and t is TCP data rate.

The TCP data rate represents TCP traffic in all EPLRS needlines. They include the original data, retransmissions and also ACKs. The sensitivity of the optimum parameter values to error rate is also considered. As error rate increases, the numbers of retransmissions increase and bandwidth utilization jumps.

3.4 Optimization Procedure and Results

Initially, the optimization is done in an error free channel in the sequence of initial retransmission period,

retransmission gain, retransmission deviation gain, retransmission deviation coefficient and the ACK delay. During each parameter optimization, a range of values is selected. The parameter that maximizes the metric m is selected. The selected value is then used for optimization of the subsequent parameter.

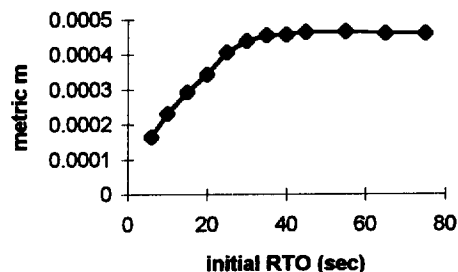


Figure 1. Sensitivity of initial retransmission period

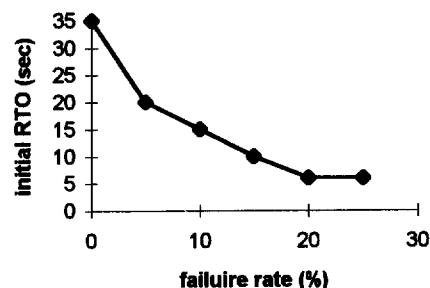


Figure 2. Sensitivity of initial RTO to failure rate

The sensitivity can be measured by the ratio of maximum metric to minimum metric. Figure 1 is a plot of metric m versus initial RTO. The ratio of maximum to minimum is about 4. For the other parameters, the ratio is around 1. Therefore, initial retransmission period is a critical parameter. The optimum value from Figure 1 is chosen to be 35. Because initial RTO is a more important parameter, more tests are performed to study its sensitivity to failure rate. Figure 2 is a plot of optimum initial RTO versus failure rate. As failure rate increases, the optimum value tends to drop. This is understandable, because a smaller retransmission period represents higher chances of retransmissions and the probability of recovering the lost packet increases.

During previous tests, parameter R1 is fixed to be 3 and parameter R2 is fixed to be 240 seconds. It is reasonable to assume a 5% error rate and initial retransmission period is chosen to be 20 seconds to optimize R1 and R2. In the

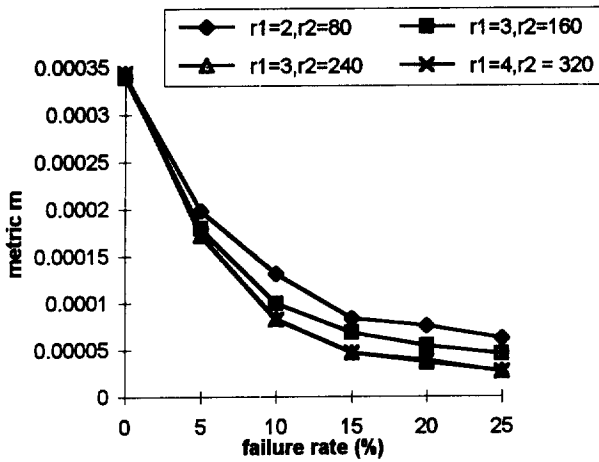


Figure 3. Sensitivity of R1 and R2 to failures

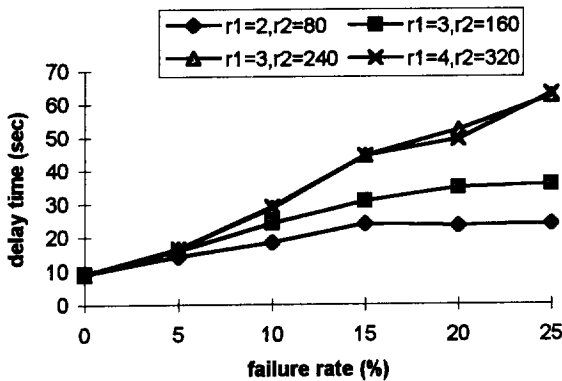


Figure 4. The performance of delay versus failures

TCP protocol, the retransmission period is doubled for failure to receive an ACK. Therefore, four sets of R1 and R2 values are chosen for evaluation. They are (R1=2, R2=80 seconds), (R1=3, R2=160 seconds), (R1=3, R2=240 seconds) and (R1=4, R2=320 seconds). Figure 3 is a plot of the metric m versus the failure rate. The plot shows (R1=2 and R2=80) is the best combination. The reason a smaller R1 is chosen suggests higher R1 tends to increase the net traffic which results in worse completion and delay. Figure 4 shows the delay performance and set (R1=2, R2=80) clearly has an advantage over the others. For a failure rate below 5%, the performance difference among all sets is very small.

4. Performance Comparison Between UDP and TCP

TCP and UDP are different in that the first is connection oriented while the second is connectionless. TCP provides an elaborate scheme to make the transmission more reliable while UDP has none. It does not seem justified to compare directly the two protocols. But, with the inclusion of Segmentation/Reassembly for UDP as was mentioned

previously, both protocols have the capability of message segmentation, acknowledgments and retransmissions. The pertinent question is which protocol has a better performance in a Tactical Internet.

There are still two major areas which are not addressed in SAR. One is flow control and the other is adaptive adjustments of retransmission period. For TCP, the initial retransmission period is a critical parameter that can affect network performance. For SAR, there are two acknowledgment schemes. One is complete acknowledgment and the other is partial acknowledgment. Complete acknowledgment is when the receiver informs the source concerning the receipt of a complete message. Partial acknowledgment is when the receiver informs the source concerning which segments received thus far. If a source cannot receive complete acknowledgment from the receiver within the timeout period, the original message is retransmitted. If the receiver cannot assemble all the segments in an assembly timeout period, a partial acknowledgment is transmitted. Both timeout values are not adaptively adjusted.

4.1 Network Model and Message Traffic

To compare the performances between TCP and UDP, a company sized SINGARS net is utilized. We assume there are 16 members in the company. Every member sends out traffic with an exponentially distributed interdeparture time at a data rate of 16 kbps.

Even though TCP has an adaptive scheme to update the retransmission period, an initial value still has to be specified. A small value can quickly congest the net while a large value can easily increase the delay. A similar phenomenon appears when a fixed value is used in SAR. For a fair comparison, some initial tests are performed to find the appropriate timeout values.

The intranet and link layers follow the 188-220A specification. The radio embedded network access scheme is utilized to schedule network access opportunities. The CSMA scheme is utilized for channel access. Even though all stations are assumed to be fully connected, topology update messages are active in all the tests.

4.2 Test Results

In the first test, the message size is 1024 bits and the message segment size is 3968 bits. Therefore there is only one segment. For SAR, this means only complete

acknowledgment is active because message assembly is not necessary. Figure 5 is a plot of completion rates against the input rate which is a measure of the total net traffic. If

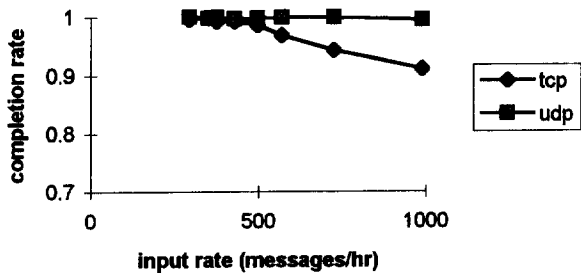


Figure 5. Comparisons of completion rate for a company net

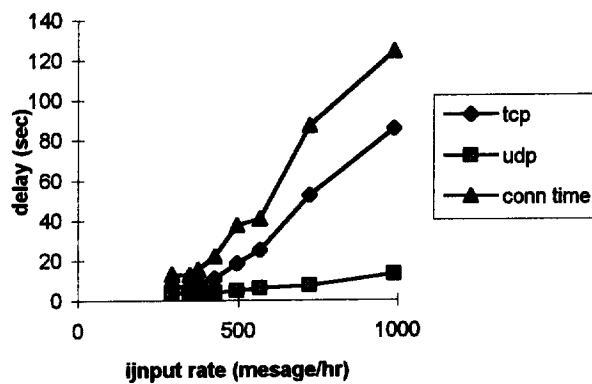


Figure 6. Comparisons of delay for a company net

the traffic is light, the completion rate difference is small. However, as the traffic increases, UDP clearly demonstrates an advantage. Figure 6 is a plot comparing the delay between UDP and TCP. The superior performance of UDP is quite obvious especially with increased net traffic. Also shown in the figure is the average TCP connection setup time which increases with net traffic.

Tests were also performed for message sizes greater than 3968 bits. In that case, there will be message segmentation and reassembly. For SAR, the accumulation timer becomes active now and partial acknowledgments can be sent out. However, the results still show UDP outperforms TCP in both completion rate and delay. Actually, the performance difference even widens. Tests performed for the platoon net show no changes in the conclusions.

5. Conclusion and Discussion

In this paper, we have discussed three issues related to TCP in a tactical internet. They are TCP connections, TCP parameter optimization and performance comparisons between UDP and TCP. The revised TCP connection procedure was used in all the tests. Therefore, a TCP station can set up connections with any number of stations without having to know in advance the number of stations with which it may communicate.

Comparison of TCP and UDP in the context of this paper demonstrates the superiority of UDP even with the inclusion of SAR. The results show the potential of replacing TCP with the UDP and SAR combination. TCP message number is measured by bytes. The information message from different packets must be received in the correct byte sequence. For SAR, each message is assigned a unique sequence number. The complete message identifiable by the sequence number can be received out of order. That is one of the reasons UDP outperforms TCP. However, before UDP can replace TCP, the SAR must also be enhanced. One enhancement is to design an adaptive scheme for the retransmission timeout. The other is to put in flow control. The SAR can be further enhanced to perform the bookkeeping operation for the selected directed broadcast [3,7]. The bookkeeping operation involves the tracking of whether a message has been received by all the destinations.

The optimization of TCP parameters shows that the initial retransmission timeout is a critical parameter. A suitable choice of this parameter is necessary in order to optimize network performance.

REFERENCES

- 1 "Transmission Control Protocol," RFC793, Sept. 1981
- 2 R. Braden, "Requirements for Internet Hosts, Communication Layers," RFC1122, October, 1989
- 3 MIL-STD-2045-14502-1A, May 8, 1995
- 4 Enhanced Position Location Reporting System, Hughes Aircraft Company, 1993
- 5 Rekhter and T. Li, "A Border Gateway Protocol (BGP-4)," RFC1654, July 1994
- 6 J. Moy, "OSPF Version 2," RFC1583, March 1994
- 7 Y. J. Liu and Eric Whitehill, "On selected Directed Broadcast of MIL-STD-2045-14502-1A," MILCOM'96