International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2020 IJSRCSEIT | Volume 6 | Issue 3 | ISSN : 2456-3307
DOI : https://doi.org/10.32628/CSEIT206360

Robust Communication Protocol Over Lossy Network

Kaustubh Chandorkar, Gajendra Dhaytadak, Siddhant Negi, Subodh Karve
Computer Engineering, Datta Meghe College of Engineering Airoli, Navi Mumbai, Maharashtra, India

ABSTRACT

A computer network is the infrastructure that allows two or more computers (called hosts) to communicate with each other. The network achieves this by providing a set of rules for communication, called protocols, which should be observed by all participating hosts. The need for a protocol should be obvious: it allows different computers from different vendors and with different operating characteristics to 'speak the same language'. Communication protocols are formal descriptions of digital message formats and rules. They are required to exchange messages in or between computing systems and are required in telecommunications When two computer system (server and client) communicate with each other they may encounter data loss due to many obstructions like thick walls,jammers etc. these may cause the data packets to be lost while transmission. We used a protocol to minimize or avoid this loss. This can be done by intelligently routing the data packets via paths that are not jammed and time effective as well.If the messages or data packets are not delivered to the receiver the protocol will resend the data packet. If there is traffic means data packets are delayed due to some obstacles then the server will choose feasible route to deliver the data packets. We use random linear network coding (RLNC) based scheme for multipath communication in the presence of lossy links with different delay characteristics to obtain ultra-reliability and low latency. A sliding window version of RLNC is proposed where the coded packets are generated using packets in a window size and are inserted among systematic packets in different paths. The packets are scheduled in the paths in a round robin fashion proportional to the data rates. We use finite encoding and decoding window size and do not rely on feedback for closing the sliding window. Our implementation of two paths with LTE and WiFi characteristics shows that the proposed sliding window scheme achieves better latency compared to the block RLNC code. It is also shown that the proposed scheme achieves low latency communication through multiple paths compared to the individual paths for bursty traffic by translating the throughput on both the paths into latency gain.

Keywords : Robustness, Slidng Window Protocol

I. INTRODUCTION

TCP was first designed for wired network. Packet loss is considered to be caused by network congestion, which triggers the congestion avoidance mechanism and then reduces the congestion window. Because random packet loss is rare in wired network, this design works well in wired network, however we cannot ignore random packet loss in wireless network. As a result, the strategy that as long as the loss event happens the congestion window is reduced seriously affects the efficiency of TCP in wireless network To solve these problems, TCP/NCfirstly introduced network coding to TCP, which has been developed quickly since it was initial proposed and researched in Network coding can mix data to mask packet loss, across time and across flows, which makes data transmission over lossy wireless networks robust and effective. TCP/NC adds network coding layer between TCP layer and IP layer. The coding layer will send the random linear combination of packets which are organized into coding block, with additional redundancy packages added in each coding block. Since the ACK in TCP/NC is no longer used to acknowledge the sequence number of packet but degrees of freedom, traditional congestion control algorithms which are based on packet loss are not suitable as congestion control algorithm of TCP/NC which selected Vegas based on RT T finally. The test results showed that TCP/NC has a good performance improvement compared with standard Cubic TCP. TCP/NC did not propose an appropriate transmission scheduling policy and dynamic adjustment strategy of redundancy parameter which is the key element of whether NC can perfectly meet TCP or not. P. Karafillis and K. Fouli proposed an advantages of network coding and Cubic by adjusting the transmission scheduling mechanisms and can adjust redundancy parameter dynamically algorithm to improve sliding window of TCP/NC whose main process identifies and then retransmits the number of outstanding lost packets. Xudong Fan and Hui Li proposed an algorithm that adjusts redundancy parameter dynamically by introducing time-cache and time-classification

management algorithm which is mainly based on time difference between ACKs. This scheme works well in the environment which has stable RT T relatively, but it is easy to overreact when fluctuation of RT T is large. Hamlet Medina Ruizt and Michel Kieffer[6] distinguish the random packet loss and congestion packet loss, then dynamically adjust the redundancy according to the number of received duplicate ACK. But it is inappropriate that the sender decreases redundancy parameter a fixed value immediately as long as packet loss happens, because this handling will reduce redundancy parameter to the minimum quickly. In terms of congestion control algorithm, CTCP with coding layer between user space and TCP layer proposed complete new congestion control algorithm which adopts a new backoff factor based on RT T leading to sensitivity to RT T. YongsuGwak and Young Yong Kim[8] presented WiCUBIC, which begins the convex window growth when exiting from recovery phase and entering the stage of congestion avoidance if the sender have judged that the duplicate ACKs is cause by the random packet loss. Because WiCUBIC uses interval time between two congestion events to judge whether the duplicate ACKs is cause by the random loss or not, WiCUBIC is not applicable for the situation that has short interval time between two random loss events. In this Paper, the proposed Coded Cubic combines network coding and Cubic which is a less aggressive and more systematic derivative of BIC and allows for more fairness between flows since the window growth is independent of RTT. Coded Cubic has absorbed the advantages of network coding and Cubic by adjusting the transmission scheduling mechanisms and can adjust redundancy parameter dynamically

II. LITERATURE REVIEW

Some research papers regarding Robust communication were studied by us. These included various techniques of communication between computer systems, how the AI component of the system can be applied that is what algorithm can be used for the training set, how these methods can be implemented in a feasible way, etc. Some of the papers which were more relevant to this are mentioned below.

Saad Rizvi proposed the multiple routing protocol which says The protocol constructs multiple paths based on residual energy of the nodes in the network and allows the source node to make energy based decisions to select a path for data transmission from the set of discovered paths. Using alternative paths for routing data packets incorporates load balancing in the network which maximizes network lifetime and minimizes energy consumption. The results show that the proposed algorithm has lower residual energy (84%, 78%, and 60% less) and longer network lifetime (394%, 195%, and 105%) than directed diffusion and its variants.

G. KarjothThe author introduces a simple language, a subset of the formal language LOTOS, expressive enough to describe a wide range of communication behavior. He shows the applicability of this specification language to a sliding-window protocol taken from the literature. The specifications start with the definition of the service of the protocol. The underlying service is precisely defined, giving the frame into which the protocol has to fit. The first specification defines the protocol entity as a single process. Then this sequential process is decomposed into four parallel processes, taking the replication of data objects into account. In this way, the author obtains a specification that contains adequate detail for use as an implementation specification To distribute the total traffic among available paths the source node performs traffic allocation based on empirical jamming statistics at individual network nodes. If any path to be disturbed/jammed a routing path is requested an existing routing path is not be updated, the responding nodes along the path will disconnect the routing path. We propose techniques for the network nodes to estimate and characterize the impact of jamming and for a source node to incorporate these estimates into its traffic allocation. We show that in multi-source networks, this centralized optimization problem can be solved using a distributed algorithm based on decomposition in network utility maximization.

III. REQUIREMENT ANALYSIS

This project requires three laptops where one of them will be server where all three laptops should be connected to the same network. Network can be LAN network or mobile wi-fi hotspot. A high configuration system to install the software so that it works seamless. (3.2.2 hardware) Protocol will help the areas where data connectivity is not strong enough and where data packets are frequently lost. It will help to find alternate paths for data transmission. Unlike its antecedent it will also report or reply about the lossy path to the source node which will help the source node to make better estimations for future use of that route.

A.Hardware requirements:

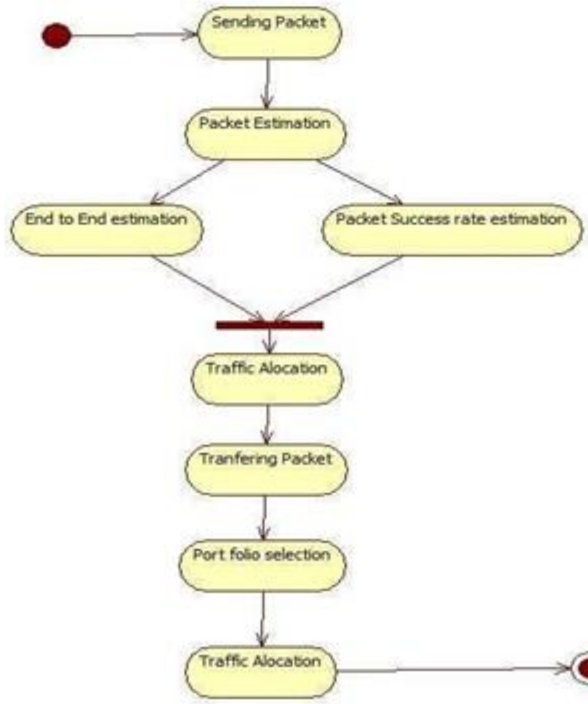
1)Processor: Intel Processor above than 3.0 Ghz

2)Hard Disk: 20 GB.

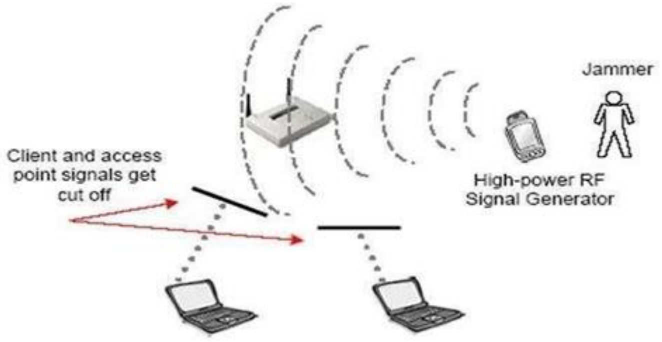
3)Ram: 1 GB.
B.Software requirements:

1)Operating system : Windows XP SP2 or Higher
2).NET Framework : 3.5 Or Higher
3)Reporting Tool : Crystal Reports XI Or Higher
4)Other Application :Adobe Acrobat Reader, M S Excel

IV. DESIGN PROCESS



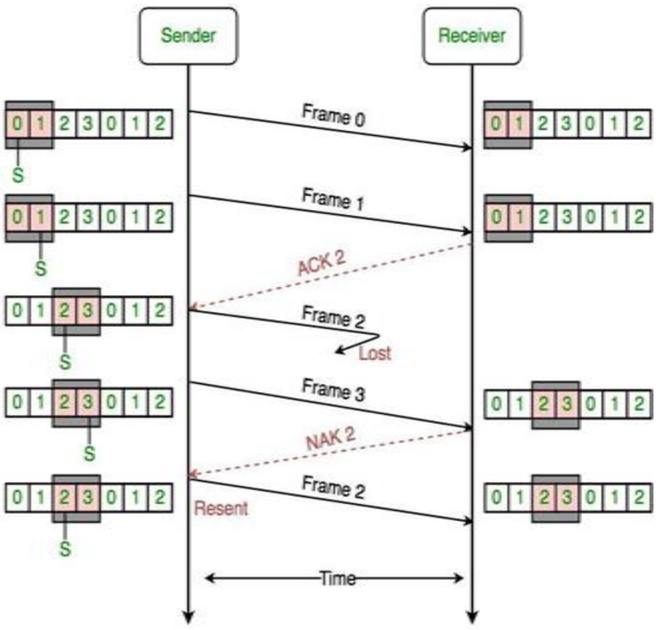
JAMMING BEHAVIOURS:



- disturb wireless communications
- proactive / reactive
- constant, random, repeat, deceive
- single bit/packet
- outsider / insider
- static / mobile

V. DESIGN AND IMPLEMENTATION

SLIDING WINDOW PROTOCOL:



1)Anti-jamming techniques diversity
2)Multiple frequency bands
3)Different MAC channels
4)Multiple Routing paths
5)Multi-Path Routing
6)Each source node chooses multiple p
7)Each path is allocated with different t amount (how to avoid congestion?)
8)Each path has different probabilities jammed (how to measure this?)

VI. TECHNOLOGIES USED

1) Visual Studio: Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps. Visual Studio uses Microsoft software development platforms such as Windows API, Windows

Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code.

Visual Studio includes a code

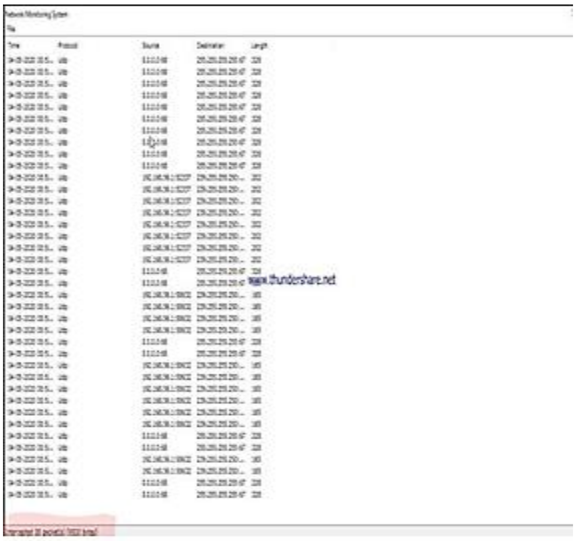
Editor supporting IntelliSense (the code completion component) as well as code refactoring. The integrated debugger works both as a source-level debugger and a machine-level debugger. Other built-in tools include a code profiler, designer for building GUI applications, web designer, class designer, and database schema designer. It accepts plug-ins that enhance the functionality at almost every level—including adding support for source control systems (like Subversion and Git) and adding new toolsets like editors and visual designers for domain-specific languages or toolsets for other aspects of the software development lifecycle (like the Azure DevOps client: Team Explorer).

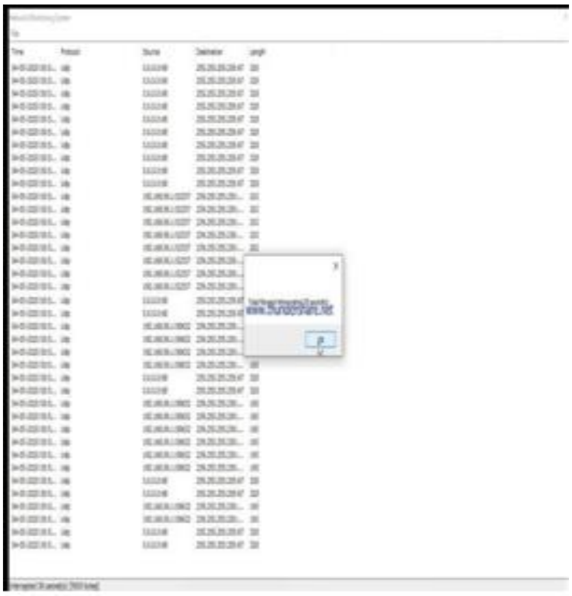
Visual Studio supports 36 different programming languages and allows the code editor and debugger to support (to varying degrees) nearly any programming language, provided a language- specific service exists. Built-in languages
include C,[8] C++, C++/CLI, Visual Basic

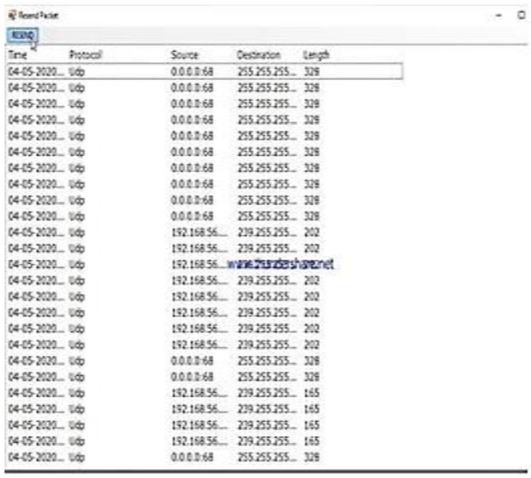
.NET, C#, F#,[9] JavaScript, TypeScript, XML, XSLT, HTML, and CSS. Support for other languages such as Python,[10] Ruby, Node.js, and M among others is available via plug-ins. Java (and J#) were supported in the past.

VII. RESULTS

NUMBER OF PACKETS TRANSFERRED	421
NUMBER OF PACKETS INTERCEPTED	99 (17673 BYTES)







VIII. CONCLUSION

This Protocol will help to mitigate the Lossless of the path for future as well as help to avoid such paths for data transmission.

Each time a new routing path is requested or an existing routing path is updated, the responding nodes along the path will relay the necessary parameters to the source node as part of the reply message for the routing path. Efficiently allocate the traffic to maximize the overall throughput. The packets which are not sent will be resend. Hence there will be no data loss. No packet delaying. Data will not be modified by anyone or there will be no eavesdropping .

IX. REFERENCES

[1]. E. M. Sozer, M. Stojanovic, and J. G. Proakis, "Underwater acoustic networks," IEEE J. Ocean. Eng., vol. 25, no. 1, pp. 72–83

[2]. R. Anderson, Security Engineering: A Guide to Building Dependable Distributed Systems. New York: Wiley

[3]. Y. M. Lu, V. W. S. Wong, "An energy-efficient multipath routing protocol for wireless sensor networks," Int. J. Commun.Syst., vol. 20, pp. 747–766, 2007.

[4]. Z. Wang, E. Bulut, and B. K. Szymanski, "Energy efficient collision aware multipath routing for

wireless sensor networks," In Proc. of the 2009 IEEE Int. Conf. on Commun., Dresden, 2009, pp. 91-95.

[5]. D. Ganesan, R. Govindan, S. Shenker, and D. Estrin, "Highlyresilient, energy-efficient multipath routing in wireless sensor networks," in ACM mobile comput. andcommun. Review, vol. 5, no. 4, 2001.

Cite this article as :

Kaustubh Chandorkar, Gajendra Dhaytadak, Siddhant Negi, Subodh Karve , "Robust Communication Protocol Over Lossy Network", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 6 Issue 3, pp. 272-277, May-June 2020. Available at doi : https://doi.org/10.32628/CSEIT206360
Journal URL : http://ijsrcseit.com/CSEIT206360