International Journal for Research in Engineering Application & Management (IJREAM)
ISSN : 2454-9150 Vol-05, Issue-01, April 2019

Priority Based Algorithmic Design of Flow Control Protocol for IoT using LiFi

B R Vatsala , Assistant Professor, The National Institute of Engineering, Mysore, India

Dr C Vidya Raj, Professor, The National Institute of Engineering, Mysore, India

Abstract- **Internet of Things which is a network of physical objects can be used to ease the life style of people with better facilities such as smart cities, homes, hospitals that give a better service than the traditional one. Many such applications generate large volume of data that may not lead the network stable. The attention should be paid towards controlling such data flow. The flow control algorithms that are used in computer networks cannot be used efficiently here as they are independent of applications and the flow control developed for one application need not be applicable to other. Here we design a protocol that controls data flow based on priority of the data sent in various applications that use IoT. Also we give feasibility analysis of the protocol designed at different levels.**

Keywords —Algorithm, Effective Window, Internet of Things, Flow control protocol, Priority, TCP, Wireless Body area network.

I. INTRODUCTION

The IoT is seamless connection of billions of digital devices, people, services and other physical objects having the potential to interact and exchange information about themselves and their environment [2]. Internet of Things plays a vital role in the near future. People want their work to be done quickly with ease. The idea behind Internet of Things is that Things instead of humans can make use of Internet to communicate making life easy.

The scenarios where IoT can be used are in developing Smart Cities [3], Smart Homes, Smart Hospitals, Smart Offices, Smart Transportation, Military Systems, Smart Banks, Logistics Applications, Smart Parking System [4], Business and many more. Big data are generated by IoT systems such as Sensor data by weather monitoring systems, health and fitness data generated by Wearable Health care and fitness devices, data generated by vehicle tracking etc. The generation of big data in many IoT systems necessitates the need to develop a flow control protocol [5] to make the network stable.

A Flow Control Protocol for Internet of Things is proposed along with its feasibility analysis at sender, router and receiver level. Also IoT applications optimized for various levels are explored.

This paper is organized as follows. Section II focuses on the topology considered for IoT. Section III discusses the related work towards flow control protocol, Section IV discusses our proposed High level design of Flow control Protocol. Section V discusses the Analysis of Proposed protocol at sender side node level Section VI discusses the Analysis of Proposed protocol at intermediate router level. Section VII discusses the conclusions

II. IOT TOPOLOGY CONSIDERED FOR DEVELOPING FLOW CONTROL PROTOCOL

Figure 1 shows a high-level topology considered for Internet of Things [8]. IoT Nodes in the short range communication form IoT LAN using LiFi [14] technology which has more than 1GB data rate. IoT LANs are connected to the Internet through IoT gateways which routes Packets between IoT Nodes and any device connected to the Internet.

IoT nodes use UDP protocol [17] at layer 4, which does not provide any flow control resulting in buffer overflow at the intermediate IoT gateway. Intermediate IoT gate way uses layer 3 protocol to route packets and does not have any flow control mechanism. TCP is used as layer4 protocol [9] in the present Internet which provides flow control through AIMD and Slow start mechanism.

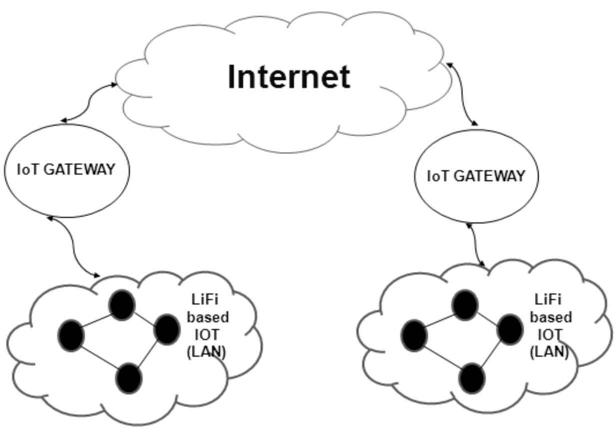

Fig.1: IoT Topology

 DOI : 10.18231/2454-9150.2019.0324

III. RELATED WORK

Flow control is imposed in the Internet through TCP using Slow-Start and AIMD (Additive Increase Multiplicative Decrease) Each TCP connection starts with initial congestion window of size 1 MSS. In Slow-Start congestion window is added with one MSS for every new acknowledgment received, A connection enters Slow-Start on newly starting or on experiencing a packet retransmission timeout, and exits Slow-Start when it detects a packet loss or when the congestion window has reached a dynamically computed threshold, ssthresh, which is set to half of the current congestion window when packet loss was detected. TCP exits Slow-Start to enter the Congestion Avoidance phase, where it continues to probe for available bandwidth. During periods when no packet losses are observed, TCP performs an Additive Increase of the window size, by 1 MSS each time a full window is acknowledged and decreases window size by half when congestion is detected. Flow control in different flavors of TCP is as follows:

TCP Tahoe: Van Jacobson introduced Tahoe congestion control. Initially sender sets cwnd to 1 MSS and enters Slow-Start in which the packets are increased exponentially on receipt of each acknowledgement until packet loss occurs which is detected through time out, and enters congestion Avoidance phase in which packets are increased linearly [9]. The main drawback of Tahoe is that whenever a packet is lost it waits till timeout occurs to retransmit it, also initial size of congestion window which is 1 MSS is the main inefficiency of Tahoe in high bandwidth delay product links

TCP Reno: The procedure followed by reno is Fast Recovery and Fast-Re-Transmit, Packet loss is detected through three duplicate acknowledgment packets, upon receiving 3 duplicate ACKs packets are retransmitted without waiting for time out [10]. After retransmission Reno enters into the Fast Recovery, in Fast Recovery after the packet loss, the congestion window size is reduced to half of current window size. TCP Reno is efficient when only 1 packet is lost; in case of multiple packet loss it is not efficient.

TCP NewReno: NewReno reduces cwnd by half on detecting a packet loss through three duplicate acknowledgments, similar to Reno but it is optimized for multiple packet losses [11]. Here Fast Recovery phase is not entered until all outstanding data are acknowledged. In this way it avoids unnecessary multiple fast retransmit from single window data.

TCP SACK: This TCP extension called Selective Acknowledgment improves New Reno’s retransmission mechanisms. It allows the receiver to indicate up to four non-contiguous blocks of sequence numbers received correctly, thus allowing the sender to retransmit lost data more efficiently [15].

There exist many flow control protocols which are based on explicit feedback from the network such as:

XCP: XCP (Explicit Control protocol) works by involving the routers in congestion control. The routers network explicitly tells the receiver the state of congestion and how to react to it. This allows senders to adjust their windows based on the precise feedback information from receiver [12].

RCP: RCP (Rate Control Protocol) involves explicit feedback from routers along the path. Here a router maintains a single rate, R(t), for every link. The router “stamps” R(t) on every passing packet (unless it already carries a slower value). The receiver sends the value back to the sender so that it knows the slowest (or bottleneck) rate along the path. In this way, the sender quickly finds out the rate it should be using (without the need for slow-start). The router updates R(t) approximately once per roundtrip time (RTT), and strives to emulate processor sharing among flows [7].

Three level ECN: three-level ECN (Explicit congestion notification scheme improves TCP over wireless links. Using three-level ECN as the congestion feedback mechanism it avoids majority of end-to-end retransmissions, unnecessary slowdowns and timeouts caused by wireless errors and hence improves the performance of TCP over wireless links [13].

IV. PROPOSED ALGORITHM FOR PRIORITY BASED FLOW CONTROL IN IOT

Flow Control protocols developed so far such as TCP flow control tend to minimize the data flow by adjusting the data rate at the sender , RCP uses router to achieve flow control , but in all these protocols data to be sent are never analyzed. The idea behind proposed flow control protocol is to develop an analytical algorithm that is going to interpret the data to be sent and decide whether data are to be sent with high priority or not whenever the effective window size at the sender is beyond a threshold size.

Many applications such as weather monitoring, health and fitness monitoring systems that use IoT have senders transmitting sensed data. But there is no need to send these data with priority if they are similar to normal data [16].

Figure 2 represents the proposed Algorithm for flow control. First the current effective window, cwnd is computed as in TCP then it is compared with threshold window size. If the current effective window size is less than the threshold window size then the mode of flow control is priority mode otherwise it is normal mode.

During priority mode the current sensed data is compared with normal data and the data is sent through a priority queue if the data is critical, otherwise the data are sent through normal queue. Priority level of the data is found by using The Data Mask as stated in the algorithm, which is chosen based on application to select a range of normal values using principles of machine learning.

DNORM is the variable set to 1 if data can be inserted to

 DOI : 10.18231/2454-9150.2019.0324

International Journal for Research in Engineering Application & Management (IJREAM)
ISSN : 2454-9150 Vol-05, Issue-01, April 2019

normal queue, otherwise Data is sent through a priority queue

Also the algorithm increases the priority level of packets put in normal queue which is achieved by moving a packet from normal queue to priority queue after inserting n number of packets in priority queue this prevents starving of normal packets.

The variable p is set to zero initially and it is incremented by 1 every time a packet is put into normal queue. Variable p is compared with n to decide whether a packet is to be moved from normal queue to priority queue once a packet is moved from normal queue to priority queue it is reinitialized to zero

During normal mode the priority based flow control is not required and the TCP sliding window protocol is sufficient hence the packets are put into priority queue.

```
// Algorithm for Flow control in IoT systems using LiFi

Effective Window=Advertised Window -
(LastByteSent - LastByteAcked)   // follows TCP

If (Effective Window >Threshold)

DNORM:=1  else DNORM=0;

if (DNORM==0) // Priority mode
{
D:=( Current data-Normal data) AND (data mask);
If (D != 0)
{Mark data as prioritized data;
Send it to   priority queue;
p++;
}
Else
Send it to normal queue;
If (p==n)
{
Move a packet from normal queue to priority
queue
p=0;
}
}
Else
{send packet to priority queue; p=0;} // Normal
```

Fig.2 High level design of Algorithm for Flow Control

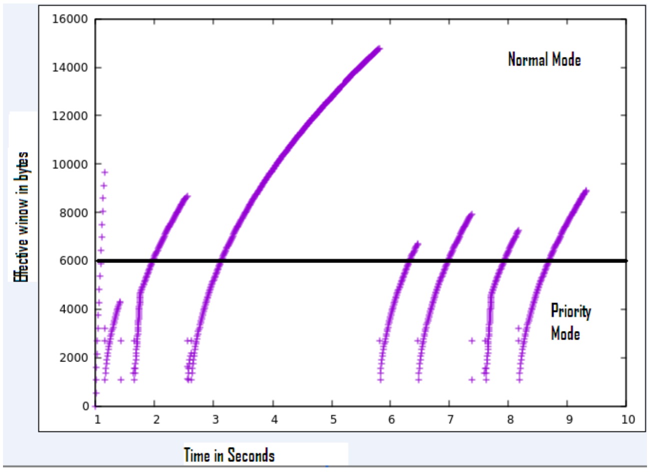

Fig.3 Growth of Effective window in TCP

Figure 3 demonstrates the growth of Effective window in on-off application using TCP in NS3. The Effective window drops to small values whenever there is packet drop when effective window is below threshold (which is se to 6000 bytes) Priority mode is used and when effective window is above threshold normal mode is used. The threshold is calculated based on past history of congestion.

Performance Analysis: Nodes that send data that are critical get more bandwidth than the other nodes as in fig 4.

Fig.4 Bandwidth assignment for node with priority and node with normal data

V. Feasibility Analysis of Proposed Protocol at Sender IoT Node Level

The protocol can be applied at the sender IoT node level for each IoT Node as in Figure 5. The Normal data are stored in a data store. The sensed data are processed as proposed. The advantage is not much burden on IoT gateway and gateway used in computer networks can be easily ported to form IoT gateway. The main disadvantage here is that there will be more processing at sensor node level which leads to more power requirement and increased sensor node size. This approach is suitable for weather monitoring system where data are collected from IoT devices that monitor temperature, humidity pressure etc where the sensor node size does not affect the system.

 DOI : 10.18231/2454-9150.2019.0324

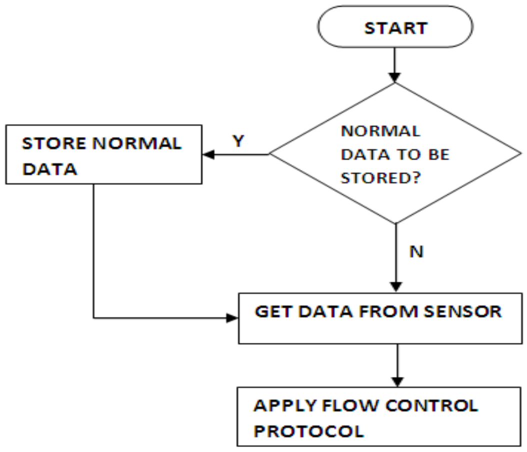

Fig.5 Flow Control protocol at sender level

VI. Feasibility Analysis of Proposed Protocol at Intermediate IoT Node Level

Router Level: The protocol can be applied at the router level in layer 3 as in Fig 6. The Normal data are stored in a data store for each IoT Node connected. The data of each sender are extracted from the packet and processed as proposed. The main advantage here is that there will not be any flow control processing at sensor node level leading to low power light weight IoT Nodes, but the burden is put on Router leading to increased size, power consumption, resource requirement and buffer management at the router.

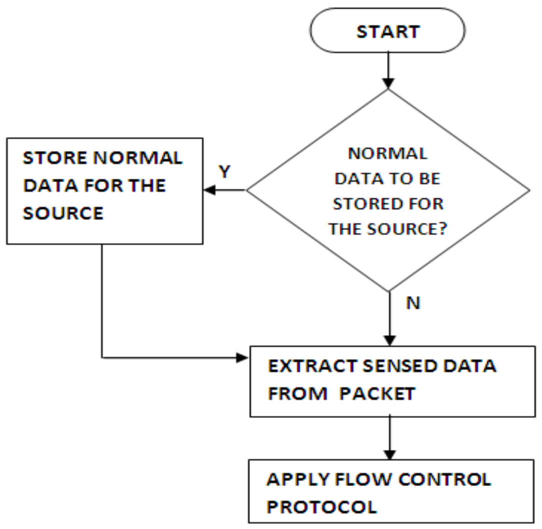

Fig.6 Flow Control protocol at Router level

Health and fitness monitoring systems [1] help doctors to treat their patients even when they are away from patients. Here the patients wear IoT devices that allow continuous monitoring of physiological parameters such as body temperature ,heart rate, blood pressure, ECG, EEG ,Glucose level etc these wearable devices form body area network [6]. IEEE 802.15.6 proposes the standard Low power, short range, and extremely reliable communication for wireless Body Area Network. Thus the proposed flow control protocol at router level is applicable in this case as the devices (sensor nodes) are free from running flow control

VII. Conclusion

Many IoT Applications generate big data which necessitates the need for application specific flow control protocols. Flow control protocols such as TCP flow control protocol cannot be applied here since these are not based on interpretation of data.

Flow Control Protocol for applications that use IoT is proposed by considering the priority of data. The packets with high priority data are transmitted first whenever congestion occurs. Feasibility Analysis of proposed flow control protocol is discussed at Sender and Router level. Applications that are apt for the proposed protocol are explored.

The proposed protocol is not feasible if applied at receiver level as it does not ensure flow control but makes receiver complex.

Acknowledgment

We acknowledge the support of the National Institute of Engineering in the completion of this work and also acknowledge authors whose paper has been referenced.

References

[1] Amna Abdullah, Asma Ismael, Aisha Rashid ,Ali Abou-EINour andMohammed Trique "Real Time Wireless Health Monitoring Application using Mobile Devices" , International Journal of Computer Networks & Communications, Vol 7, No, 3,May 2015.

[2] Charith Perera, Arkady Zaslavsky, Chi Harold Liu, Michael Compton, Peter Christen, and Dimitrios eorgakopoulos, "Sensor Search Techniques for Sensing as a Service Architecture for the Internet of Things", IEEE Sensors journal, Vol. 14, Feb 2014.

[3] Jiong jing, Jayardhana Gubbi, Slaven Marusic, Marimuthu Palaniswamy, "An Information Framework for Creating a Smart City through Internet of Things", IEEE Journal PP 2327-4662,2013

[4] S. Lee, D. Yoon, and A. Ghosh, "Intelligent parking lot application using wireless sensor networks," in Proceedings of International Symposium on Collaborative Technologies and Systems (CTS), 2008, pp. 48-57.

[5] B R Vatsala, Dr C vidya Raj, "Internet of Things: Usage of LiFi and Need for Flow Control Protocol" IJRITCC Vol 2, Issue 8, Aug 2014

[6] Pedro Brandao "Abstracting information on body area networks " PhD thesis, University of Cambtridge,2011

 DOI : 10.18231/2454-9150.2019.0324

[7] Nandita Dukkipati: "Rate Control Protocol (RCP): Congestion Control to Make Flows Complete Quickly", phd thesis university of Stanford, Oct 2007.

[8] Teemu Savolainen, Jonne Soininen, Bilhanan Silverajan, "IPV6 Addressing Strategies for IoT", IEEE Sensors Journals,Vol 13, No 10,Oct 2013

[9] V.Jacobson, "Congestion Avoidance and Control," SIGCOMM Symposium on Communications Architectures and Protocols , pages 314-329, 1988

[10] V.Jacobson "Modified TCP Congestion Control and Avoidance Alogrithms".Technical Report 30,Apr 1990

[11] V.Jacobson, "Congestion Avoidance and Control," SIGCOMM Symposium on Communications Architectures and Protocols , pages 314-329, 1988

[12] D. Katabi, M. Handley, C. Rohrs, " Internet Congestion Control for High Bandwidth-delay product networks", Proceedings of ACM Sigcomm, Pittsburgh, Aug 2002

[13] Mohammad Usman Ali Khan, Shahid Khan, Yasir Ali Rasheed, "Optimization of TCP over Wireless Networks" , International Journal of Electrical & Computer Sciences IJECS-IJENS, Oct 2011

[14] Haas, Herald "Wireless data from every light bulb",TED Global. Edinburgh, Scotland, Jul 2011

[15] http://www.rfc-editor.org/rfc/rfc2018.txt

[16] www.ccpe-cfpc.com

[17] http://www.rfc-editor.org/rfc/rfc768.txt

 DOI : 10.18231/2454-9150.2019.0324