A CONCEPTUAL FRAMEWORK: Provisioning Quality of Service in Internet of Things

Aabidah Nazir [1], Prof. Moin Uddin [2]
[1] *Department of Computer Science*
School of Engineering Sciences and Technology, Jamia Hamdard, New Delhi-110062, India
[2]*Department of Computer Science*
School of Engineering Sciences and Technology, Jamia Hamdard, New Delhi-110062, India

Abstract— Quality of Service (QoS) is the capability of providing satisfactory service to one of the critical components and in Internet of Things (IoT); there is great need of QoS to have better communication between devices. Various QoS parameters like delay, bandwidth, response time etc. need to be addressed. Delay is time period by which data packets gets postponed or late and ensuring delay free network is hard to obtain and due to fast growing of IoT, things are connecting at fast speed which in turn leads to more delay in network. The aim of connect the unconnected affects the Quality of Service and there is the need to address delay sensitive traffic or critical traffic that should be communicated in real time. This paper identifies various QoS parameters and proposes a model to reduce the queuing delay at the server side, so as to transmit the delay sensitive information in real time without packet loss. Also dedicated processors are allocated to both priority and non-priority data to have fast processing of critical data in real time in an efficient manner.

Keywords—: *Internet of Things, Dynamic memory allocation, Pool of Processors, Quality of Service, Queuing delay Introduction*

I. INTRODUCTION

The term Internet of Things was brainstorm by Kevin Ashton of Proctor and Gamble, later MIT's Auto-ID centre in 1999 [1].
The Internet of Things (IoT) is reforming the way of communication of physical objects with each other. Internet of Things is the network of things that are communicating and exchanging information with each other with the help of internet. The things in IoT can be any physical device like vehicles, buildings, home appliances, camera, oxygen masks, MRI dater, security alarms et. These devices are connected with personal computers or mobile phones by GSM, GPRS and #G networks [2]. Also these things participate actively in activities of exchanging information and making realistic decisions without the interruption of humans [3]. The IoT devices are embedded with sensors, actuators, gateways, and network. As things in IoT get added day by day, the network is widened which leads to slow down the performance of communication. For the delay sensitive things and the traffic they generate over the network it is required that they must communicate in real time. The smart devices are rigged in homes, hospitals etc. to assure safety, security, offices and immense conduct of electrical appliances. These devices are sending their information to server may be laptop or mobile through certain protocols like 6LoWPAN, Wi-Fi, Zigbee, and Bluetooth etc. These are the connectivity protocols and depending upon certain factors like range, security, data requirement, power and battery life of the devices will precept the preferred protocol taken for connectivity. The server is receiving all data packets whether critical data like security alarm, camera data or non-critical data like refrigerator data, vehicle data etc. and the server is supposed to response the critical/priority traffic where the quality of service is needed and response is send in real time. The priority data packets can be data coming from security alarms, cameras, oxygen masks etc. the non-critical data is the non- fatal data coming from refrigerators etc. The servers where all the data is stored for analysis are having small buffer size and the buffer cannot hold packets when it is full. The packets are lost whether priority or non- priority and also due to delay the response is not sent in real time. The model that is proposed in this paper reduces the delay and also has no packet loss. This model is developed to contemplate fast delivery of sensitive data to the server where IoT applications reside. The practical solution for queuing delay reduction is given with no packet loss.

In IoT, there is the interconnection of components having diverse technical features, and thus needs to be provide seamless and adaptive Quality of Service in order to have successful communication network. In delay sensitive things, the information needs to be communicated in real time and the parameters which are taken frequently by researchers are network layer parameters. These parameters take more time to communicate from source to destination and it needs to be minimized so that the packets/information gets transmitted with less delay and in real time.

II. ARCHITECTURE OF IoT

IoT architecture comprises of three layer and various QoS factors at layers of IoT architecture are as: i) Perception layer ii) Network layer iii) Application layer [4, 5]

Perception layer: - This layer includes sensing data and gathering data from real world objects, machines, and people. Also includes controlling devices/actions based on the sensed data. It is the physical layer which comprises of sensors, actuators etc. Also the function of this layer is governing of field devices/reactions based on anticipated data and curb requests received by uppermost layers of domain. The prime goal of things of this layer is to exclusive address identification and communication between tactical technologies like RFID, Bluetooth, and 6LoWPAN (Low Power Personal Area Network).

Network layer: - It the transport layer which is responsible for routing data from source to destination. Its function is addressing and processing sensor data. The functions of this layer include i) all kinds of network protocols ii) Routing protocols and functions iii) connectivity devices. This layer uses the IP protocol like IPV6

which is the default protocol. This layer is the most developed one in IoT architecture.

Application layer: - This layer comprises of application modules used for data analysis, computation, and for real world action. Its main responsibility is delivering application specific assistance to the end users. It consists of the functions like escalation of duplicate and unnecessary data from field devices, functions for storing and retrieval of data for consequential references, and retrieval of data for flexible and changing decisions etc. Some of these factors are dispensable between application and perception layers for accomplishing effective QoS.

The parameters at each layer of IoT architecture which can be used for maximization of QoS are summarized in table given below [4].

IoT Layer	***QoS Parameters***
Perception Layer	Sampling Parameters, Time synchronization and Locality/mobility, Sensing and actuation coverage.
Network Layer	Bandwidth, Delay, Packet loss rate, Jitter, Utilization of network resources, Life time of sensing network, Reliability, Throughput and Real time
Application Layer	Service time, Services availability, Service delay, Service accuracy, Service load, Service priority, Information accuracy, cost of network deployment, Cost of service usage, Maximum number of resources available per unit price and Penalties for service degradation and fault tolerance

Table: QoS Parameters of three layer IoT architecture.

The paper below is organised as: section ii is related work, section iii explains the proposed model, section iv are the results and section v gives the idea of future scope.

III. RELATED WORK

In [6], a cost-effective systematic model for finite size queuing system with deterrent resumes avail priority and protrude buffer management scheme was proposed. Also queue length and blockade probability of high and low antecedence traffic was analysed. In [7] a charismatic markov chain based scheduling was recommended to ensure QoS for delay responsive traffic, also data was divided into two-priority and non-priority queue. Based on the survey, some important issues are being noticed like packet loss. The parameter that is taken into account in this paper is delay which is an important parameter of QoS model. Delay is the period of time by which something is late, slow or postpone. In network, delay may be processing delay, transmission delay, queuing delay, and this paper proposes a model which reduces the queuing delay with no packet loss. In [8], an aggressive packet scheduling strategy is considered to administer service discernment and preferential analysis to delay sensitive freight. In [9], quality of service requirement are analysed with internet of things and provides a decomposition and optimization method for the quality of the Internet of Things, and puts forward four basic QoS calculation methods. In [10], an Emergency Response IoT based on Global Information Decision (ERGID) is proposed to improve the performances of decisive data transmission and adequate necessity response in IoT. In [11], a QoS architecture based on IoT layered structure was proposed and the architecture arranges QoS agent in curtailed layers then addresses QoS obligations, vexing to guarantee the flexibility as well as effectively use the existing QoS tools in every layer. In [12], system architecture of testbed is proposed with adaptive Quality of Service (AQoS). An AQoS concept administers a malleable experimentation of reacting to dynamic changes of network conditions and thus network based information can be made based on historic data of testbed and adjustments to network can be done. In [13], an approach for network administration which can be practiced to structure of traffic paths and behaviour of nodes incorporated in the paths in a software defined network is proposed. It gives the detail behaviour of each switch node that may be employed as network resources, so that a service path aiding a user-customized utility adequately may be provided. In [14], a huge buffer model is developed for WLAN media access protocol that provides the information of throughput and delay predictions, thus having increased buffering can sort out the inequity problem but increases delay. In [15], an Awareness Driven Schedule was introduced that aware about differentiated data services provided by sensors. It tells that higher the awareness on sensors resources it should provide more detailed data service. In [16], a cost-effective analytical model based on markov chain is proposed to ensure QoS for transmission of delay sensitive traffic but does not give details about blocking probability, delay, etc. In [17], one of the QoS mechanism is packet scheduling that is used to select packet to be serviced and packet to be dropped so, packet scheduling provides ability of service provisioning and differentiates critical from non-priority traffic. Thus QoS awareness is established in IoT for allocating traffic priorities and scheduling with appropriate algorithms

IV. PROPOSED MODEL

The approaches proposed before provide unsatisfactory solutions for delay sensitive traffic and are having certain limitations like packet loss [6], more complex, more computations, less reliable, less efficiency due to single processor system of server. Considering these limitations, a new model is proposed in this paper. In the model, the IPV6 data packets sent by IoT devices through internet to the server are divided into two-based on the priority of data packets. The traffic arriving from IoT devices are directed into two queues in the buffer: priority and non-priority at the server. The data that needs immediate response goes to priority queue and the data that if takes some time for response back is send to non-priority queue. In the server, the pool of processors is taken which are responsible for servicing the requests from IoT devices. In the pool some processors are dedicated for priority traffic and some for non-priority traffic. So that there will not be any delay in sending the response for priority traffic. As the queues are of fixed size like buffer in real time there can be packet loss when the queue becomes full. In order to handle the problem of packet loss the memory is allocated dynamically can be cache memory. In this model the dynamic list is allocated to store the packets coming from internet when the static array or queue becomes full. So for both priority and non-priority queue the dynamic list is allocated to hold the packets. When the priority queue starts becoming free, from the list the packets are transferred

to queue so that all the data packets gets processed with minimum queuing delay.

In the server, for having the pool of processors the multithreading is done. So that the data from priority queue gets a dedicated thread which can be free at that time to reduce the processing time. In real sense there can be server with seven processors and some processors are particularly servicing the priority traffic and some are servicing non-priority traffic.

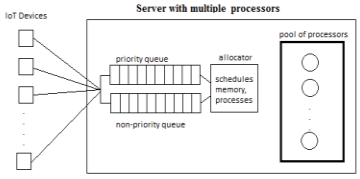


Figure 1: A New Model providing maximum QoS

It is supposed that traffic arrives as per poisons distribution and the queuing system taken is M/M/1/N (arrival process/service process/no. of servers/no. of nodes). Let a1 be the no. of packets arriving at priority queue and a2 is the no. of packets arriving at non-priority queue. The buffer size of each queue is taken 10 i.e. N=10, at a time queue can accommodate only 10 requests and rest of the requests goes to list. The arrival rate of priority traffic is λ_1 and for non-priority traffic is λ_2. The service rate priority traffic is μ_1 and for non-priority traffic is taken as μ_2. The arrival rate follows poisons distribution (random) and service rate follows exponential distribution. For M/M/1/K model some expression for calculating delay and queue size are as [18]:

For priority traffic:

Average queue size = arrival rate * average waiting time (queuing delay and service rate)

$$E(n_1) = E(\upsilon_1)$$

Average waiting time $E(w_1) = \lambda_1/ \mu_1(\mu_1- \lambda_1)$

Average total delay $E(\upsilon_1) = 1/(\mu_1- \lambda_1)$

Expected no. of units in system $E(n_1) = \rho_1\{1-(N+1)\rho_1^N+N\rho_1^{N+1}\}/1-\rho_1(1-\rho_1^{N+1})$

According to little's Law: $E(w_1) = \lambda_1E(\upsilon_1)$

Expected no. of units in the queue $E(m_1) = \lambda_1*E(w_1)$

Mean queue length $\lambda_1E(w_1) = \lambda_1^2/\mu_1(\mu_1-\lambda_1)$

Probability of n units in system $P(n_1) = (1-\rho_1)(\rho_1)^{n_1}/1-\rho_1^{N+1}$

Probability of more than N packets in queue $P(n_1>N) = \rho_1^{N+1}$

For non-priority traffic:

Average queue size = arrival rate * average waiting time (queuing delay and service rate)

$$E(n_2) = E(\upsilon_2)$$

Average waiting time $E(w_2) = \lambda_2/ \mu_2(\mu_2- \lambda_2)$

Average total delay $E(\upsilon_2) = 1/(\mu_2- \lambda_2)$

Expected no. of units in system $E(n_2) = \rho_2\{1-(N+1)\rho_2^N+N\rho_2^{N+1}\}/1-\rho_2(1-\rho_2^{N+1})$

According to little's Law: $E(w_2) = \lambda_2E(\upsilon_2)$

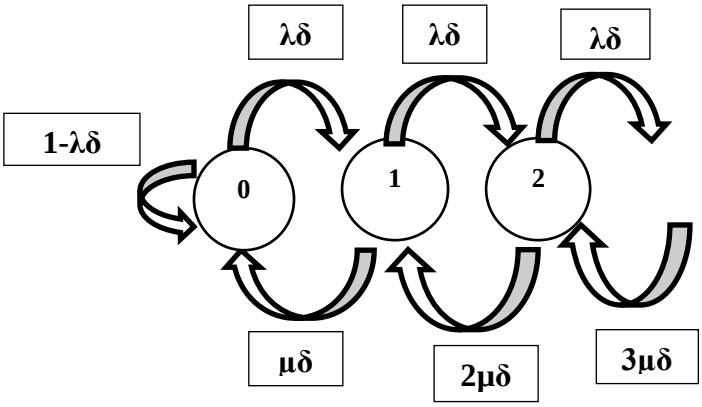
Expected no. of units in the queue $E(m_2) = \lambda_2*E(w_2)$

Mean queue length $\lambda_1E(w_2) = \lambda_2^2/\mu_2(\mu_2-\lambda_2)$

Probability of n units in system $P(n_2) = (1-\rho_2)(\rho_2)^{n_2}/1-\rho_2^{N+1}$

Probability of more than N packets in queue $P(n_2>N) = \rho_2^{N+1}$

When the λ packets enter the queue and m no. of processors are allotted to provide service to the processes, then μ packets departs per second from per processor of the server. The departure rate is proportional to the number of processors in use [19].



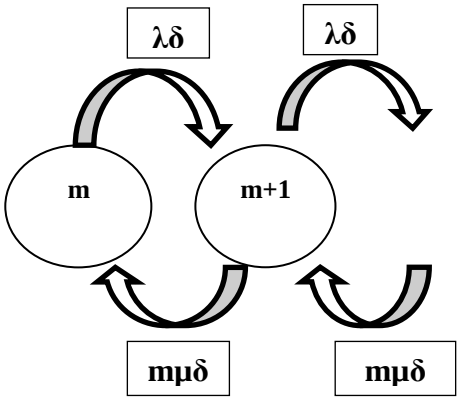


Figure 2: M Processor System (M/M/1/K)

The time taken by each process for its completion is given total nodal delay. There are types of delay in network stated as processing delay, queuing delay, transmission delay and propagation delay. If the delay are represented as

Processing delay-d_{proc}

Transmission delay-d_{trans}

Queuing delay-d_{queue}

Propagation delay-d_{prop}

Thus nodal delay $d_{nodal} = d_{proc} + d_{trans} + d_{queue} + d_{prop}$

This delay component's addition varies significantly. The propagation delay can be negligible in a single network like LAN and in the given model it is done on same network thus its value is zero.

The transmission delay can be from zero like in LAN to momentous. And the processing delay is often zero as depends on rate of processing by server. The only and complicated delay that influences the nodal delay is queuing delay. The queuing delay depends on arrival rate of traffic whether arrives periodically or in bursts. In the given model, the traffic arrives in bursts that are randomly having no periodicity. The traffic intensity La/R, where L is the no. of bits in packet, a is the average arrival rate and R is the transmission rate of pulling packets from queue, is not taken greater than 1as if taken greater the queue will increase without bound and queuing delay increase infinitely [20].

Packet loss is the shedding of data packets when the queue is full and not in a mode to accept more packets. In real world, queue is having

finite capacity due to design and cost. When the packet arrive the finite capacity queue and finds it full, the server will drop the packet when it does not find any place to store or hold the packet and the packet will be lost. Thus performance of network gets affected as performance depends on both delay and packet loss. In the model given in this paper, the packets loss is negligible as the dynamic memory is allocated when the queue is full to hold the packet whether priority or non-priority.

V. IMPLEMENTATION AND RESULTS

In this particular section the proposed model is implemented in java platform and its performance is analysed. The parameters like arrive time, queue time, list time and execution time are evaluated. The request model here sends 15 messages at one click but the static queue size is 10 only so 5 request goes to dynamic list and 10 requests goes to queue for processing. The scenario is for both priority and non-priority traffic. The request model is created in which all the parameters like arrive time are noted to calculate the total time taken from source to destination

The request model is created in which all the parameters like arrive time are noted to calculate the total time taken from source to destination.

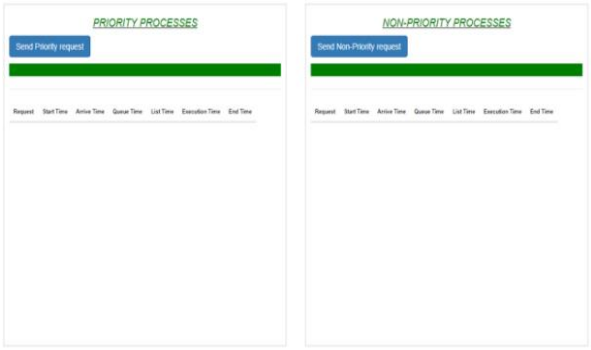


Figure 3: Server and Queue with no request

The implementation figure 3 given shows the empty queue when there is no request in the queue and no processor is allocated to process the request. The server is idle which does not happen and when there is not any request sent by smart devices.

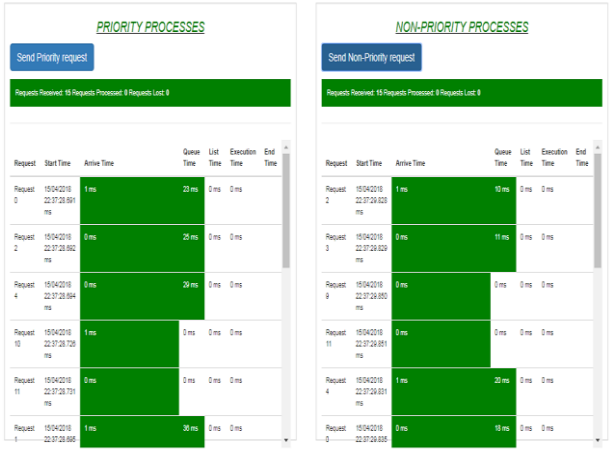


Figure 4: Server and Queue with minimum no. of requests

The implementation figure 4 shows that the processors are processing requests and queue is full and also some of the requests are in list.

The time is calculated for every request in list, queue and total execution time so that efficiency of servicing the requests can be calculated.

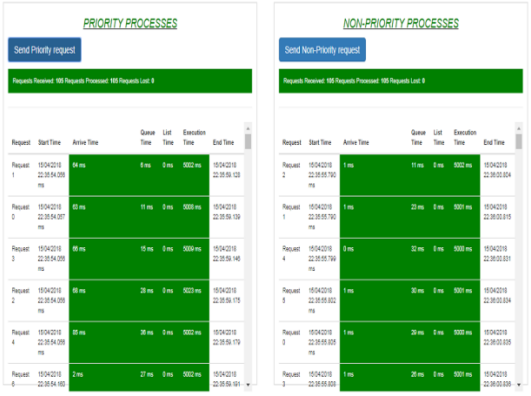


Figure 5: Server and Queue with maximum no. of requests.

The implementation figure 5 shows the maximum no. of requests in list and the processors are processing the requests very fast. The processors in server are allocated to process both priority and non-priority requests.

The impact of dynamic memory allocation is that blocking probability is zero. And also the impact of dedicated processors is to minimize queuing delay. The effect of arrival rate hike on blocking probability is negligible. The load is balanced on processors by scheduling processes to free processor. The graph shows the time at which packets arrive the queue, time at which enters the list, and the total execution time. The graph's colour changes to green at every stage of processing like entering queue, entering list and total execution. The graph also shows the total packets arrived and packets lost status.

VI. CONCLUSION

A cost-effective and less time consuming model for processing critical data packets and providing maximum QoS in IoT is proposed. In IoT network, large amount of requests/data are sent to the server at a time which leads to congestion in network and also servers with small buffer size or finite capacity queuing system are not able to hold more packets. The model gives an efficient way to have less delay with no packet loss. The interpretive model can be used to conclude the performance of devices by varying the traffic in order to reconcile the QoS constraints. The model uses the queuing management scheme in an efficient way and provides services to both priority and non-priority traffic in quick manner with zero blocking probability. The model can be analysed for different traffic by varying the arrival rate of data traffic. In future, there is the need of making IoT devices more efficient so that they can send only the priority data in real time. And non-priority data can be sending directly for historic analysis.

VII. REFERENCES

[1]. Gubbi, Jayavardhana, et al. "Internet of Things (IoT): A vision, architectural elements, and future directions." Future generation computer systems 29.7 (2013): 1645-1660.

[2]. Atzori, Luigi, Antonio Iera, and Giacomo Morabito. "The internet of things: A survey." Computer networks 54.15 (2010): 2787-2805.

[3]. Nef, Marie-Aurélie, et al. "Enabling qos in the internet of things." Proc. of the 5th Int. Conf. on Commun., Theory, Reliability, and Quality of Service (CTRQ 2012). 2012.
[4]. Bhaddurgatte, Ravi C., and V. A. Kumar. "Review: QoS Architecture and Implementations in IoT Environment." Research & Reviews: Journal of Engineering and Technology (2015): 6-12.
[5]. Bilal, Muhammad. "A Review of Internet of Things Architecture, Technologies and Analysis Smartphone-based Attacks Against 3D printers." arXiv preprint arXiv:1708.04560(2017).
[6]. Awan, Irfan, Muhammad Younas, and Wajia Naveed. "Modelling QoS in IoT applications." Network-Based Information Systems (NBiS), 2014 17th International Conference on. IEEE, 2014.
[7]. Sharma, Reema, and Navin Kumar. "QoS-alert markov chain based scheduling scheme in internet of things." Globecom Workshops (GC Wkshps), 2015 IEEE. IEEE, 2015.
[8]. Sharma, Reema, et al. "Waiting Time Analysis for Delay Sensitive Traffic in Internet of Things." Region 10 Symposium (TENSYMP), 2015 IEEE. IEEE, 2015.
[9]. Ming, Zhou, and Ma Yan. "A modeling and computational method for QoS in IoT." Software Engineering and Service Science (ICSESS), 2012 IEEE 3rd International Conference on. IEEE, 2012.
[10]. Qiu, Tie, et al. "ERGID: An efficient routing protocol for emergency response Internet of Things." Journal of Network and Computer Applications 72 (2016): 104-112.
[11]. Duan, Ren, Xiaojiang Chen, and Tianzhang Xing. "A QoS architecture for IOT." Internet of Things (iThings/CPSCom), 2011 International Conference on and 4th International Conference on Cyber, Physical and Social Computing. IEEE, 2011.
[12]. Ezdiani, Syarifah, et al. "An IoT environment for WSN adaptive QoS." Data Science and Data Intensive Systems (DSDIS), 2015 IEEE International Conference on. IEEE, 2015.
[13]. Kim, Eun Joo, Jong Arm Jun, and Nae-Soo Kim. "The method of controlling traffic paths in IoT-based software defined network." Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), IEEE Annual. IEEE, 2016.
[14]. Duffy, Ken, and Ayalvadi J. Ganesh. "Modeling the impact of buffering on 802.11." IEEE Communications Letters 11.2 (2007).
[15]. Guo, Haoming, Shilong Ma, and Feng Liang. "Enabling awareness driven differentiated data service in IoT." Journal of Networks 6.11 (2011): 1572-1577.
[16]. Awan, Irfan, and Muhammad Younas. "Towards QoS in internet of things for delay sensitive information." International Conference on Mobile Web and Information Systems. Springer, Cham, 2013.
[17]. Abdullah, Saima, and Kun Yang. "A qos aware message scheduling algorithm in internet of things environment." Online Conference on Green Communications (GreenCom), 2013 IEEE. IEEE, 2013.
[18]. En.wikibooks.org. (2018). Fundamentals of Transportation/Queueing - Wikibooks, open books for an open world. [online] Available at: https://en.wikibooks.org/wiki/Fundamentals_of_Transportation/Queueing [Accessed 14 Apr. 2018].
[19]. Web.mit.edu. (2018). [online] Available at: http://web.mit.edu/modiano/www/6.263/lec5-6.pdf [Accessed 14 Apr. 2018].
[20]. Net.t-labs.tu-berlin.de. (2018). Delay and Loss in Packet-Switched Networks. [online] Available at: https://www.net.t-labs.tu-berlin.de/teaching/computer_networking/01.06.htm [Accessed 14 Apr. 2018].