International Journal of Engineering and Information Systems (IJEAIS)
ISSN: 2000-000X
Vol. 2 Issue 1, January – 2018, Pages: 190-203

New model 6G LTE Systems M2M Sushisen Algorithms

Prof P.Senthil MA.,PGDCA.,M.Sc.,MPhil.
Associate professor in MCA Computer Science,
Kurinji College of Arts and Science, Tiruchirappalli-620002.India.
Email:senthilkanapathy@gmail.com

Abstract: *Unique With the development of machine to machine (M2M) correspondence, transmitting information from various heterogeneous gadgets will make movement and corrupt the nature of administration (QoS)in the not so distant future . Existing cell systems give an extensive variety of scope; in any case, gauges foresee up to 50 billion M2M interconnected gadgets by 2020. This is required to make conditions for over-burden in the RAN (Radio Access System) and Center System in 6G LTE systems. There are many examinations that look at different attributes of M2M specialized gadgets including securing the physical gadgets, validation techniques, clog controls, security insurance and numerous others. Be that as it may, blockage will be a persevering issue with the expanded gadgets and is the concentration of this paper. There are specialists on the techniques to control blockage, in any case, this paper is thinking about expanding accessibility through lessening all out bytes transmitted and hence staying away from or diminishing over-burden and clog in LTE arrange.*

Keywords: LTE, M2M, RAN validation techniques, clog controls, security insurance

1. INTRODUCTION

With the rapid evolution of M2M devices, service operators and researchers estimate the mobile traffic will see a tremendous growth in coming years. Fourth Generation Long Term Evolution (5G LTE) is a part of the Third Generation Partnership Project (6G) which offers high data rates, improved coverage and better resource management; however, with better-quality broadband, it is currently designed to support fast growing human to human communications. With the inclusion of M2M devices, the current trend shows a 26% growth every year in M2M devices using the cellular systems such as 5G LTE. This will substantially increase the traffic in LTE network thus leading to increased congestion levels. With 50 billion interconnected devices in two or three years, congestion in LTE network remains a challenge for service providers. Many studies have been conducted to address congestion, though few consider availability by rejecting or delaying the packet flow and not providing the service whenever required. A method to control congestion by reducing total bytes transmitted is proposed through this research. This can be achieved by using lightweight compression methods in Java cards, a robust header compression method and by transmitting only when values are outside of the normal range determined by the user.

Network congestion is a predicted challenge for future cellular networks. In future, traffic generating from M2M is estimated to increase at a rate of 71%where M2M devices itself will be 43% of total devices and connections. In LTE, a Radio Resource Control (RRC) is established before any data transmission irrespective of the data size. This RRC connection process has 12 interactions with the RAN and 18 interactions with the core network devices for device setup. During this time if enormous numbers of devices attach to the network this may cause congestion in the RAN and core network, thus deteriorating the QoS [3]. To compensate for loss of packets and congestion, aggressive retransmission is used by some network protocols which puts addition load onto network devices.

TCP is the common protocol used which by itself has the ability to control congestion. However, TCP here is less efficient since the congestion avoidance mechanism is triggered only after a congestion has occurred [4]. At present, 6G System Architecture is on the point of establishing requirements for improving the network systems. Much of the research to deal with congestion is by implementing algorithms that reject packets in time of congestion [5]. However, little research has tried to optimize the network using test-bed approaches or by applying big data dynamically to identify areas of congestion. This research will be focusing on reducing congestion by examining high availability in 5G LTE network due to massive devices that are increasingly attached to the network. Reduction can be achieved by eliminating values sent when they are within a normal range generated from a M2M device thereby only sending values when above a user defined maximum or below a user defined minimum level.M2M devices such as e-health, and smart grids generate a lot of unsolicited and duplicate data for real time monitoring. Finding methods to eliminate the values within normal limits and transmitting only the values above and below the threshold could drastically bring down the congestion when massive M2M devices are considered and this can thus increase availability. For example, a smart pressure monitoring system which sends pressure values every designated time with the use of the above solution could be restricted to send only the values above and below the normal limits. The values within the normal range indicate no action required by the monitoring personnel, but transmitting only values above maximum and below minimum could alert staff where an action needs to be taken. A similar approach can be applied to other real-time monitoring apps such as e-health using LTE network. Sending only that is only above or below a range could save resources and reduce signalling overhead and congesting the bandwidth.

2. CONGESTION BACKGROUND

Chang, Chia-Wei, et al. in [3] proposed two congestion controlling algorithms which efficiently reduce congestion. Using the proposed algorithm, they achieved 20-40% improvement with respect to the accept ratio, overload degree, and waiting time compared with those in LTE. To control congestion, 6G rejects the request of delay-tolerant devices and restrict them from accessing the network during the congestion predefined time. The issue relates to how long the pre-defined time should be; anything too small or too large will have impacts on network traffic and signalling. If the predefined time is too small, the devices will impose other requests resulting in higher signalling overhead. In contrast, if pre-defined time is too large there would be idle time in network resulting in lower utilization and efficiency of network devices. In the future, for large amount of M2M devices it's a challenge to reduce congestion maintaining high utilization. The authors[3] have proposed an algorithm that rejects M2M device requests and determines an appropriate back-off timer during network congestion. The authors have also worked on an effective queuing system to select the back-off period. The value of the back-off timer depends on the number of requests waiting in the queue. The benefit of using this is the chance of a device issuing another request with low utilization will be reduced. Load distribution over the space domain is another algorithm that has been proposed. This algorithm is effective if the load remains high due to huge number of requests in that the algorithm helps to shift load to another network device. If all network devices are busy, a suggestion is using network function virtualization to create a new instance of certain network entities to distribute the load.

The proposed algorithm improved the performance for accept ratio, overload degree and waiting time, thereby efficiently decreasing the load on a high utilized network entity. However, although the authors improved the performance, congestion still exists in the study and rejecting M2M devices during congestion remains a question of availability and QoS for the end user.

The contributions of the work done by Bouallouche, Dalicia et al. in [4] identified where congestion occurs particularly in LTE networks. The main types of LTE network congestion withM2M devices areradio network congestion and core network congestion. Radio network congestion frequently happens in eNodeB when high number of devices are trying simultaneously to connect to the network and activate a channel. The core network congestion because a large number of devices try to simultaneously transmit data. It generally happens in Mobility Management Entity (MME) and Home Subscriber Server (HSS), Serving Gateway (S-GW) and Packet Data Network Gateway(P-GW). Work by authors in [4] is mainly focused on the congestion due to signalling messages from several/numerous M2M devices. The authors have distinguished congestion avoidance schemes such as rigid (reject connection request) and soft (minimize the frequency) methods. The authors have proposed to adapt a feedback controller known as proportional–integral–derivative(PID) controller which corrects the behavior of the process that needs to be managed.

However, they insist that not much defined work has been done on controlling congestion in the core network. In their work, they have tried to resolve or control congestion in the radio network, but controlling congestion in the core LTE network remains relatively un-researched.

The work contributed by Amokrane, Ahmed, et al.in [5] propose congestion aware admission control which selectively rejects signalling messages from M2M devices at the radio access network based on a PID controller reflecting a congestion level from the core network. They have tested their proposed solution and succeeded in reducing the signalling to reach a target utilization ratio. In this research, they have focused on the problem of congestion when deploying M2M devices. Their proposal includes a method to reject M2M signals at eNodeB when core network is congested. The clear majority of signalling congestion uses a rigid method by rejecting requests. There is a time-period set for the M2M to connect at a certain time and there is time-period that does not allow any connection from M2M devices. The solution the authors have proposed is a CAAC (congestion aware admission controller) which combines congestion control and congestion admission control. It ensures to avoid the immediate re-initialization of requests and set a predefined time to request for admission. Using this, it can reject traffic depending on its priority. Traffic with higher priority will have less rejected packets as compared to traffic with a lower priority. It also uses a PID feedback loop for congestion monitoring which overall help to calculate the amount of M2M traffic to be rejected by the eNodeB.

The work by Chen, Jiann-Liang et al.in [6] propose an optimization method for controlling congestions in LTE networks. Two platforms are suggested by the authors such as NetFPGA and Open-Flow platforms to achieve a test-bed framework. Net field programmable grey array (NetFPGA) is a low cost open platform proposed by Stanford

University. The advantages of using this is line-rate, flexible, open platform, are in fast networking hardware prototyping. The open flow is an open standard which is based on Ethernet switch. During congestion in core network, eNB will reject or release the connection with higher delay tolerance and feedback a back-off value. M2M connection does not initiate a RRC connection when the device issues a request but issues a timer to wait. Access class barring is also implemented during high traffic.

The work contributed Parwez, Md Salik et al.by in [7] uses a big data approach to find unusual traffic using Call Detail Record (CDR). This will analyze anomalous behavior of cellular network. For anomaly detection purposes, they have used k-means and hierarchical clustering. An anomaly is considered during high network load and this can help to identify regions of interest for special action such as fault avoidance and resource allocation. With anomalous and anomaly free data, a neural network based prediction model is trained. In their paper, they have identified anomalies using the clustering methods mentioned above. Thus, they could analyze user activities at different locations and time. Using the clustering techniques, user activities that were unusually high caused unusual traffic demand. Finally, when they verified these anomalies with the factual information, they found the unusual traffic was resulting from a sports stadium. Therefore, with the help of these methods, proper resource allocation can be provided in advance to meet the requirements. Also with the neural network training method they found anomaly free data had less mean square error than with anomalous data. With big data, by extracting this information, smart and intelligent resource allocation algorithms could be developed for efficient resource utilization. However, with this approach they will not be able to reduce congestion in core network but will be able to identify resources in advance that are required during particular network events. The feasibility of detecting anomalies every time requires a lot of manual intervention and analytics. Implementing an analytic and big data method in the core networks to identify the region of interest can be useful for setting up additional resources such as eNodeB's (eNB's) for load control.

Most of the congestion control methods explained above suggests congestion could be reduced by rejecting or delaying the requests from M2M devices. These methods proved to be most useful and this can reduce congestion during high load. However, by rejecting the requested service to the end user the utility of the M2M device from a user perspective will be reduced. This end user availability aspect is still not being fully addressed and research must focus on something that ensures all data are efficiently transmitted with a low rejection rate while at the same time avoiding congestion.

As it is evident from the above, current mobile standards are not efficient enough to handle tiny payloads and support large number of devices simultaneously which consequently leads to congestion. Therefore, extensive research has to be done to make sure the radio resources for M2M are utilized efficiently. traffics [8].By not efficiently utilizing radio resources, M2M traffic is expected to degrade the performance of traditional cellular traffic. Y. Mehmood et al. analyzed the inclusion of e-healthcare traffic being prioritized over other LTE traffic, traffic such as HTTP and FTP which has low priority traffics are degraded considerably with the rise of the number of M2M devices [8].

Group based M2M authentication and communication could play a crucial role to decrease congestion in signalling on the air interface. In a group based mechanism to manage M2M traffic, a group head is identified which collects data packets, status information and other details. and forwards the information to eNB or M2M devices. [8] Moreover in several models of group authentication only the group leader is authenticated by the HSS and all group members can be authenticated by the MME, thus alleviating the core network of this authentication load by forcing it to the edge of the core network (i.e. the MME).

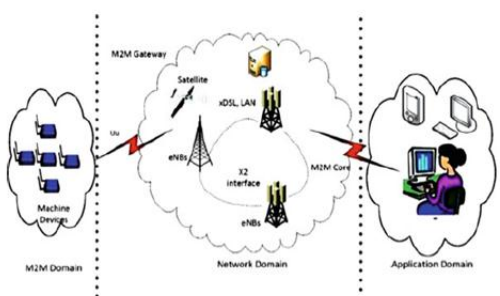

Fig. 1. High level architecture for M2M communication system [8]

International Journal of Engineering and Information Systems (IJEAIS)
ISSN: 2000-000X
Vol. 2 Issue 1, January – 2018, Pages: 190-203

The work contributed by Blessing Orakwe et al.in[11] describes an attack tree that shows various ways to launch Denial of Service (DoS) attack through paging the Universal Mobile Telecommunication System (UMTS)3Guser. In their work, they have described signaling attacks which can cause congestion in RAN by constantly sending low volume bursts at an appropriate time-period immediately after the Radio Access Bearer (RAB) is inactive in 5G(i.e. during power saver mode). Once the RAB is released a burst from the attacker will trigger a new RAB establishment causing severe congestion in the signaling path between RNC and Base station with the RNC processor being busy with maintaining state and processing signaling messages. This congestion can lead to valid signaling message not getting the resources allocated and getting denied by RNC due to significant delays leading to timeouts. Flooding HLR or VLR can make the valid service requests to be denied by the RNC to legitimate users. The authors recommend solving most of the DoS attack in UMTS by using Public Key Infrastructure (PKI) which protects the initial authentication, thus protecting IMSI and RAND [11].

[12] Potsch et al. contributed by showing how LTE will handle the growth of M2M devices. For this the authors specified a M2M traffic model with normal LTE traffic present in the network. The performance of M2M with conventional traffic is evaluated in this paper. Thus, they showed the influence massive numbers of M2M devices can have on cellular network. With the use of OPNET network simulator the authors were able to simulate the traffic model. The OPNET was performed under a parameter setting on how M2M will behave in the simulator. The performance of various M2M traffic loads are compared by analyzing the QoS performance of mobile devices with normal LTE traffic. Video, voice and file transfer was considered for the simulations keeping the normal LTE load constant and varying the M2M traffic. The first scenario had no M2M devices, followed by 300 M2M devices, followed by 450, 600 and 750 M2M devices. During the analysis, they found that the voice and video was having not much impact due to the priority assigned being delay sensitive data. However, with best effort traffics like file transfer it was evident from the performance test that the M2M load was high. The load significantly reduces the performance of the file transfer. The overall cell performance is shown in the below figure 2 [12.]

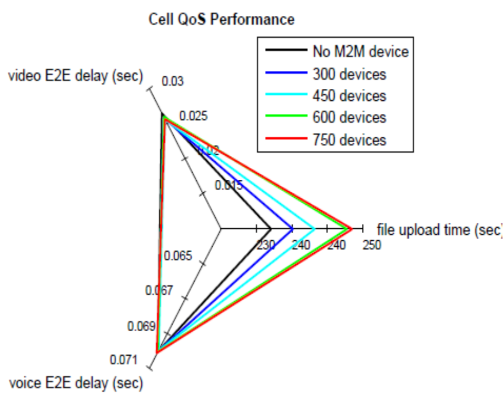

Fig. 2. Cell QoS Performance with varying M2M load. [12]

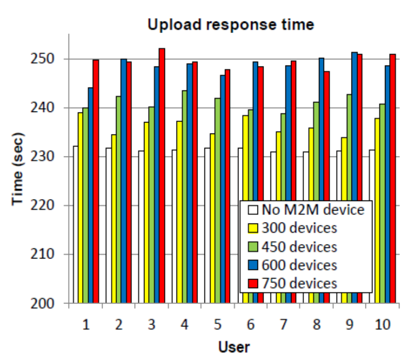

Fig. 3. Upload time for File transfer [12]

Figure 3 shows with no M2M devices the time taken to transfer file took just above 230 seconds which kept increasing as the number of M2M devices increased. From the graph, it's evident that the upload time without M2M devices was constant and there was a delay of ~7 sec for 300 devices, ~8-9 sec delay by adding 450 devices, ~15-20 sec delay by including 600 devices and ~20-23 sec delay by including 750 devices. This clearly shows with the inclusion of M2M devices it will cause very significant delay in traffic.

Figure 4 shows a comparison of future traffic model of 4Kb and far future of 8Kb which depicts the impact on QoS performance of normal LTE traffic with the potential growth of M2M devices [12].

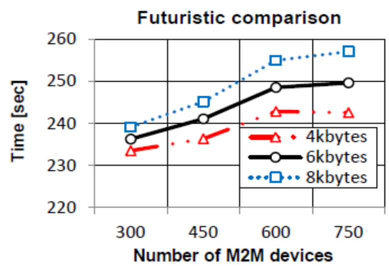

Fig. 4. Comparison of M2M Traffic Models in Terms of Average Cell File Upload Time [12]

Java cards are widely used in SIM cards for securing sensitive information managed by the service providers. M2M devices also uses Java cards to transmit information to the LTE network. Hence, any means to control the size of the transmitting information could save many network resources. Data Compression is one such method of reducing the byte size of transmitted information. A novel light weight compression method for Java cards was proposed by Massimiliano Zilli et al, by combining both dictionary compression and folder mechanism compression[13]. In their paper, they have proposed two techniques leading to higher space saving and a smaller execution time. Dictionary compression includes substituting repeated sequence of byte codes with macros, where the definitions are stored in a dictionary. Folding compression is based on a folding mechanism on a Java processor known as pico Java. A defined sequence of Java opcodes can be condensed to a single register machine like instruction. The use of register like instruction saves three instruction fetches and avoid all the memory writes and read to and from the operand stack. Light weight compression is a combination of the two-compression method mentioned above. In this paper, the authors have proved a good compression ratio with a space saving of 12% and a lesser execution time [13].

3. DESCRIPTION OF RESEARCH

With the increased dependency on the Internet for people to communicate and get timely access to information, the demand and necessity for seamless communication technology has been evolving at a rapid rate [9].The crucial role M2M will play in the near future has increased concerns about network congestion among the service provider and end users and made security of the network even more critical [10]. The main objective of this paper is to find an effective method to reduce congestion in the RAN side and core network side of the LTE architecture.

When 5G was designed, the researchers stressed network wide security architecture integrating them into both the network core and end user devices.

The key objectives in designing security architecture are as follows:

1) Availability – that imposes services are not interrupted by any attackers.
2) Interoperability – make sure to exchange and make use of information
3) Usability: convenient for end-users to use the security enabled services.
4) Maintaining QoS – Cryptographic algorithms to meet QoS implications of voice and videos.

5) Cost-Effectiveness – Reduces the additional cost security and manages the costs of risk.

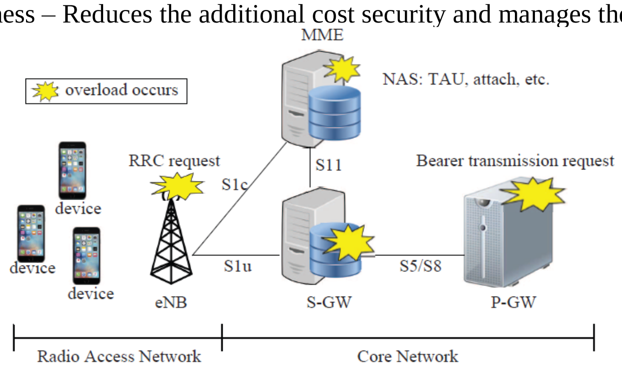

Fig 5. RAN and CN congestion [1]

Data at rest or transmitted should follow the security principle of Confidentiality, Integrity and Availability. The aim of this research is to focus on reducing congestion in order to promote high availability in LTE 5G networks. From the literature, it is evident that congestion is a significant concern due to increased devices trying to communicate each day. This can significantly increase the load in both core area network and RAN.

RAN side congestion – M2M devices and UE (user equipment) contend for same random-access channel at the same time. Continuous signalling traffic generated from the M2M devices tries to attach to the network is considered as the main fact of congestion in RAN side.

Core Network side congestion - Non-access stratum (NAS) and data transmission leads to congestion on MME, S-GW, P-GW. This is caused by the data sent by the M2M devices on the uplink. Even if the amount of data per M2M device is small, the sum of all data can lead to congestion at the S-GWs and/or P-GWs and MME.

The one of the basic requirements of several M2M devices such as e-healthcare, ITS, logistics, etc. is mobility. Devices such as e-healthcare are implemented to gather the health-related body readings such as heartbeat, decreasing rate of fat level, respiratory/ pulse rate etc. These devices transfer the gathered data to the monitoring centers where they store recorded data and take necessary actions. Generally, this is attained using a WLAN technology, however during low WLAN coverage it may create difficulty for patients in case of mobility outside the home. Likewise, smart grids rely on a M2M device that is used for monitoring electricity, gas, water consumption, monitoring and controlling utility consumption, tracing, tracking and protecting people, and assets. The majority of M2M devices such as e-healthcare, smart grid would transmit similar data at regular or irregular intervals of time.

Once M2M user devices has established the connection, all the group likely go into an idle state during inactivity which releases eNB and MME radio resources. Then, if there is data the core network needs to send to the UE, the UE needs to be woken up from idle. The MME and eNB send a paging request which serves two purposes- then the MME and eNB updates the location in case the UE is moving and secondly, a channel can be made available. Similarly, if on a timer the UE wakes up and needs to send data, it requests a paging update that again triggers the creation of a channel and moves from idle state to ready state.

4. METHODOLOGY

Though few tests have already been performed to determine congestion in the LTE network with the increase in UE devices, our purpose of this test is to get a baseline of the problem using a simulator. In this paper, the congestion problems identified above have been verified using Omnet++ simulation tool. The method adopted for this research is analyzing the congestion areas and studying each part where the congestion occurs, thereby coming up with a solution which can be efficiently worked out to reduce congestion in both RAN and core network parts of the LTE network architecture. In the Omnet++ Simulation tool, an LTE network used by many devices was set up as shown in fig 6. The LTE network consists of 10 to 50 devices to communicate over the LTE. The LTE network also consisted of an eNB, PGW, and a receiving server. The tool simulated how devices initiate a connection request to enode-B allocating an IP address from PGW and transmitting data. The objective of the simulation tool was to show the more traffic it increases the more it gets congested. Hence, we used 10 UE devices to generate traffic through a limited bandwidth. The load was at peak within 7seconds, subsequently 20 devices showed a peak traffic at 3 seconds, 30 devices showed a peak traffic at 2 seconds and 50 devices showed a peak traffic within a half second.

Fig:6. Number of UE's in a LTE network.

TABLE II. TIME TAKEN ON NUMBER OF DEVICES TO OVERLOAD TRAFFIC.

Number of UE Devices	Time taken for Traffic load
10	7 seconds
20	3 seconds
30	2 seconds
50	0.5 seconds

Optimization techniques to reduce congestion can be determined such that they minimize bytes that are transmitted and are not required. This can be achieved by:(a) implementing a Robust Header compression on LTE networks for M2M devices, (b) eliminating values within the normal limits by programming SIM (subscriber identity module) cards to send required data only, and (c) using a light weight compression method to compress data sent from M2M to the application domain. All three optimization techniques are described below:

SUSHISEN ALGORITHMS:

1. Plug in $f_n = x^n$ to get characteristic equation
2. Solve for roots of characteristic equation.
3. Set up general solution.
4. Use initial conditions to set up linear equations to solve for constants in general solution:
   a. Degree d 6G relation -> degree d characteristic equation -> d constants (unknown coefficients) in general solution
   b. d initial conditions -> d equations.
5. Always has a solution of the form $f_n = x^n$.
6. Plug into the 6G and solve for x:
7. $f_n = 5f_{n-1} – 6f_{n-2}$
8. Any linear combination
9. $f_n = \alpha_1 \cdot 2^n + \alpha_2 \cdot 3^n$ satisfies: $f_n = 5f_{n-1} – 6f_{n-2}$
10. $f_n = \alpha_1 \cdot 2^n + \alpha_2 \cdot 3^n$ is called the **general solution** of
11. the 6G relation $f_n = 5f_{n-1} – 6f_{n-2}$
12. Linear: the coefficient of each term is a constant.
    a. $g_n = 3g_{n-1} + 2g_{n-2} + n^2$ (**linear**)
    b. $g_n = 3(g_{n-1})^2 + 2g_{n-2} + n^2$ (**not linear**)
    c. $g_n = 2g_{n-1} \cdot g_{n-2} + n^2$ (**not linear**)
    d. $g_n = n \cdot g_{n-2} + n^2$ (**not linear**)
13. Homogeneous: no additional terms that do not refer to earlier numbers in the sequence.
    a. $g_n = 3g_{n-1} + 2g_{n-2}$ (**homogeneous**)
    b. $g_n = 3g_{n-1} + 2g_{n-2} + n^2$ (**not homogeneous**)
14. $g_n = 3g_{n-1} + 4g_{n-2} + n^2$
15. $g_n = 3g_{n-1} + (g_{n-2})/5$

International Journal of Engineering and Information Systems (IJEAIS)
ISSN: 2000-000X
Vol. 2 Issue 1, January – 2018, Pages: 190-203

16. $g_n = 3g_{n-1} + 2$
17. $g_n = \log(2)\cdot g_{n-2} + 5\, g_{n-7}$
18. $g_n = n\cdot g_{n-2} + 5\, g_{n-7}$
19. $g_n = g_{n-1}\cdot g_{n-2} + 5\, g_{n-7}$
20. A linear homogeneous 6G relation has the form:
    a. $f_n = c_1\cdot f_{n-1} + c_2\cdot f_{n-2} + ... + c_d\cdot f_{n-d}$
21. $c_{1,}\, c_{2,\ldots,}\, c_d$ are constants
22. If $c_d \neq 0$, **degree d** 6G relation

1) A Robust Header compression (RoHC) can be applied to M2M packets originating from the devices.

A Robust Header compression method is a potential compression method for the voice LTE (VoLTE). There are many small sizes of data carried by huge IP packets since the header part consumes more resources than the real data during a voice transmission. For an end-to-end connection, which consists of multiple hops, the protocol header can be compressed over just one link (hop-to-hop). To save the bandwidth and use the costly resources efficiently, it is possible to compress those headers which in most of the cases gives more than 90% savings. Unlike the IP header compression, RoHC compress both the dynamic and static fields of a header in packets being transmitted. A similar concept can be applied to M2M packets. M2M data are mostly bursty and frequent packets that transmit over the network. M2M often transmits small chunks of data which can be less in size compared to its IP header. Therefore, RoHC can be applied to decrease the header size to save significant resources by compressing both the dynamic and static fields in IP header. With considering similar concept if RoHC is applied to all M2M packets originating devices, it can reduce the size of bytes that are transmitted and can reduce chance of congestion.

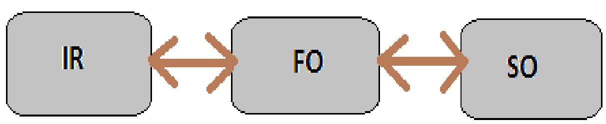

Fig. 7. Compressor State Diagram

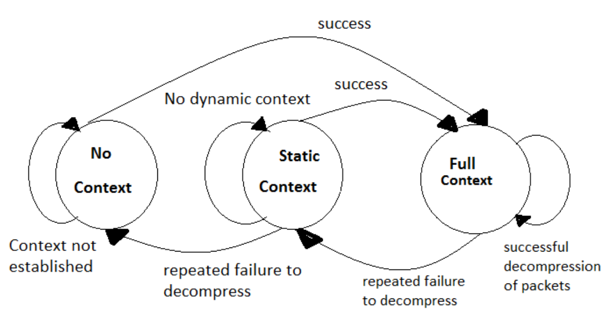

Fig. 8. De-compressor state diagram

RoCH States	Description
Initialization & Refresh (IR) state	-Compressor created and reset. -Full IP packet header is sent.
	-The compression is detected and static fields are stored.

International Journal of Engineering and Information Systems (IJEAIS)
ISSN: 2000-000X
Vol. 2 Issue 1, January – 2018, Pages: 190-203

First Order (FO) state	-The dynamic packet filtering is sent in FO state by the compressor.
Second order (SO) state	-Supress all dynamic fields. -Only logical sequence number and partial checksum is sent which causes other side to predictively generate and verify the headers of next expected packets.

2) A method to eliminate the user information within the normal limits defined by the user of the M2M devices.

This can substantially reduce load in the network and will decrease the RA request from M2M devices to attach to eNB thereby reducing signalling traffics. This method can be achieved by coding placed into the Java card of the SIM using C or assembly language to keep a small code size with high performance. It could be programmed in such a way that when a M2M device send values frequently to the SIM for transmitting to the 5G network, the program checks to see if the values are within in normal limit already defined in the C program. It eliminates any values within the normal limit and just transmits the values which are above or below the normal range. The unsolicited values are eliminated which can save bandwidth and network resources. If there is no need to transmit due to redundant values, the SIM will drop the request to attach to network. This brings advantage to both the user and service providers in terms of data usage and bandwidth respectively.

3) A light weight compression method can be applied to M2M packets originating from the devices.

As described in the work done by authors in [13], we can use the similar concept by applying light weight compression method in Java SIM card. As mentioned in the paper, light weight compression is a combined version of dictionary compression and folding compression, which can be applied to M2M devices that uses the 5G SIM to transmit data. This can substantially reduce the bandwidth and provide a faster execution.

5. EXPERIMENTAL RESULTS

From the optimization techniques described above the following tests were conducted:

1) RoHC Header Compression method was tested to compress a stream of IPv4 packets. RoHC was implemented to a Linux-based virtual machine. Since M2M packets have a relatively smaller data size compared to the header, in the test we have used a data size similar to a M2M IPv4 packet. In the test, we created five IPv4 packets with 26,21,22,23, and 21 bytes of packets. When the stream of IP packets are combined it gives 113 bytes of IPv4 packets. The RoHC program was executed for these stream of IP packets using the RoHC program, we could achieve a robust header compression of packets from 113 bytes to 33 bytes with 71% savings intotal size as shown in Appendix.

The above experiment proves that when tiny streams of IP packets are transmitted from M2M devices through LTE, robust header compression could be an effective method to reduce the packet size thereby decreasing the bandwidth. Implementing RoHC in the LTE devices at the service provider end could reduce the chance of congesting network and thereby increasing availability.

2) Eliminating the redundant information transmitted by M2M devices at regular intervals and transmitting only the required ones. In this test using a C program, we are trying to build a table that has the threshold values of basic health monitoring. The C program will check for the health rates defined in the table. If the value of health is getting within the normal range the packet will not be transmitted, and if the value gets above or below the normal limit it would be transmitted through devices using LTE as shown in Appendix. The M2M device would invoke a radio channel only when there is an information to transfer i.e. when the value is above or below the threshold limit. With this regard, the RAN side congestion would decrease considerably by avoiding frequent RRC request to the eNB.

One challenge here is during a chip failure, the concerned personnel would get no information about the malfunction and assume the values are within the normal range. The user controlling the M2M device could compensate for this problem by programming the chip to send data at a user specified time regardless of whether the data is within the threshold limit or not. In this way, a chip which had failed would not respond and would trigger the user to implement other measures to deal with a non-functioning M2M device or malfunctioning chip.

3) Using a light weight compression experimented from the work of author M Zilli et al. have proved space saving and faster execution in Java cards. Space saving can be achieved by defining

$$S_{xcompr} = \frac{AppSize_{original} - AppSize_{Xcompr}}{AppSize_{original}} [13]$$

where $AppSize_{original}$ is the size of the original application and $AppSize_{Xcompr}$ is the size after the compression. With light weight compression, they could achieve 12% space saving. For application running on Java cards the amount of space saving could significantly help the resource utilization by keeping application size minimum.

6. CONCLUSION

This paper studied the problem of numerousM2M devices in the LTE network. From the study, it is evident with the rise of M2M and IoT (Internet of things) devices LTE network can become congested both at the radio access network and core network leading to loss of availability. This paper addresses the goal of availability from the stand point of reducing total bytes transmitted. Our experiment in this paper proves the more the number of UE devices increases, the more it gets congested. This paper discusses potential solutions to reduce congestion both at the RAN side and core network of the LTE network architecture. Since M2M devices, such as e-health and many smart grids, transmit data with a size much lower than its IP header, a lot of space is currently being used by its IP header. The experiment result shows using the robust header compression technique could be an effective solution to reduce IP header overhead. In the experiment, we successfully achieved 71% savings in total size of the stream of IP packets. Application of RoHC in the LTE network could reduce bandwidth usage and help to prevent network congestion and thus provide higher availability and QoS. Another solution proposed in this paper is to reduce signalling overhead at the RAN side by eliminating redundant information sent by M2M devices using a C program in the SIM card. This could help the SIM to identify required information to be sent and eliminate unsolicited data which could reduce frequent RRC requests eventually saving resources and reducing signalling overhead. The impact of this would free up cells the eNB can allocate in their controlling area.

Future work can be considered on this method by implementing and testing the technique using real time captured information form an e-health device. Research may also consider improving on the reliability of communication during a chip failure. This paper promotes to use the optimization techniques discussed to reduce congestion in the LTE network considering the massive growth of M2M and IoT devices in the network.

7. ACKNOWLEDGEMENT

This paper is made possible through the help and support from everyone, including: My wife M.suganya and Daughter S.S.Inakshi and in essence, all sentient beings. I sincerely thank to my parents, varagupadi or Varagubady(village) family, and friends, who provide the advice and financial support. The product of this paper would not be possible without all of them.

REFERENCES

[1] Arul, R., Raja, G., Kottursamy, K., Sathiyanarayanan, P., Venkatraman, S., 2017. User path prediction based key caching and authentication mechanism for broadband wireless networks. Wirel. Pers. Commun. 94 (4), 2645–2664. http://dx.doi.org/10.1007/s11277-016-3877-5.

[2] Anwar, S., Mohamad Zain, J., Zolkipli, M.F., Inayat, Z., Khan, S., Anthony, B., Chang, V.,2017. From intrusion detection to an intrusion response system: fundamentals,requirements, and future directions. Algorithms 10 (2), 39. http://dx.doi.org/10.3390/a10020039

[3] Abu-Lebdeh, M., Sahoo, J., Glitho, R., Tchouati, C.W., 2016. Cloudifying the 3GPP IP multimedia subsystem for 4G and beyond: a survey. IEEE Commun. Mag. 54 (1),91–97.

[4] Agiwal, M., Roy, A., Saxena, N., 2016. Next generation 5G wireless networks: acomprehensive survey. IEEE Commun. Surv. Tutor. 18 (3), 1617–1655. http://dx.doi.org/10.1109/COMST.2016.2532458

[5] Basudan, S., Lin, X., Sankaranarayanan, K, 2017. A privacy-preserving vehicular crowdsensing based road surface condition monitoring system using fog computing.IEEE Internet of Things Journal

[6] Devi, R., Jha, R.K., Gupta, A., Jain, S., Kumar, P., 2017. Implementation of Intrusion Detection System using Adaptive Neuro-Fuzzy Inference System for 5G wireless communication network. AEU - Int. J. Electron. Commun. 74, 94–106. http://dx.doi.org/10.1016/j.aeue.2017.01.025.

[7] Ferrag, M.A., Maglaras, L., Ahmim, A., 2017. Privacy-preserving schemes for ad hoc social networks: a survey. IEEE Commun. Surv. Tutor., 1. http://dx.doi.org/10.1109/COMST.2017.2718178.

[8] Gupta, A., Jha, R.K., Jain, S., 2017. Attack modeling and intrusion detection system for 5G wireless communication network. Int. J. Commun. Syst. 30 (10), e3237. http://dx.doi.org/10.1002/dac.3237.

[9] Hamandi, K., Bou Abdo, J., Elhajj, I.H., Kayssi, A., Chehab, A., 2017. A privacy-enhanced computationally-efficient and comprehensive LTE-AKA. Comput. Commun. 98, 20–30. http://dx.doi.org/10.1016/j.comcom.2016.09.009.

[10] Hasan, K., Shetty, S., Oyedare, T., 2017. Cross layer attacks on gsm mobile networks using software defined radios. In: Proceedings of 14th IEEE Annual Consumer Communications & Networking Conference (CCNC), IEEE, pp. 357–360

[11] Kolias, C., Kolias, V., Kambourakis, G., 2017. TermID: a distributed swarm intelligence based approach for wireless intrusion detection. Int. J. Inf. Secur. 16 (4), 401–416.http://dx.doi.org/10.1007/s10207-016-0335-z.

[12] M. Mukherjee, R. Matam, L. Shu, L. Maglaras, M. A. Ferrag, N. Choudhury, V. Kumar,Security and Privacy in Fog Computing: Challenges, IEEE Access (2017) doi: 10.

[13] Miguel García-Pineda, Jaume Segura-García, Santiago Felici-Castell, A holistic modeling for QoE estimation in live video streaming applications over LTE Advanced technologies with Full and Non Reference approaches, Computer Communications, Volume 117, February 2018, Pages 13-23, ISSN 0140-3664, https://doi.org/10.1016/j.comcom.2017.12.010.

[14] Yasser Gadallah, Mohamed H. Ahmed, Ehab Elalamy, Dynamic LTE resource reservation for critical M2M deployments, In Pervasive and Mobile Computing, Volume 40, 2017, Pages 541-555, ISSN 1574-1192,

[15] Dong-hyu Kim, Heejin Lee, Jooyoung Kwak, Standards as a driving force that influences emerging technological trajectories in the converging world of the Internet and things: An investigation of the M2M/IoT patent network, In Research Policy, Volume 46, Issue 7, 2017, Pages 1234-1254, ISSN 0048-7333, 1109/GLOCOMW.2010.5700310.]

[16] Hafiz Husnain Raza Sherazi, Luigi Alfredo Grieco, Gennaro Boggia, A Comprehensive Review on Energy Harvesting MAC Protocols in WSNs: Challenges and Tradeoffs, Ad Hoc Networks, Available online 5 January 2018, ISSN 1570-8705, https://doi.org/10.1016/j.adhoc.2018.01.004.

[17] Nazmus S. Nafi, Khandakar Ahmed, Mark A. Gregory, Manoj Datta, Software defined neighborhood area network for smart grid applications, In Future Generation Computer Systems, Volume 79, Part 2, 2018, Pages 500-513, ISSN 0167-739X, https://doi.org/10.1016/j.future.2017.09.064.

BIOGRAPHIES

P.Senthil was born varagupadi or Varagubady(village) in Tamilnadu in India in 9 May 1987.He received his B.Sc., degree in Information Technology from Bharayhiyar University for coimbatore, Tamilnadu. He was received the M.Sc Information Technology and M.Phil degrees in Computer Science from Kurinji College Arts and Science, Trichy, Tamilnadu in 2010, and 2014, respectively. In 2012, He joined in the Department of Computer Science, Kurinji College of Arts and Science in Trichy, Tamilnadu as a Lecturer and now He is working as Associate professor in the department of MCA, KCAS College of Arts and Science, TrichyTamilnadu from June 2014 onwards. He is doing his research in Image mining & Digital Image processing at Bharathidasan University, Tamilnadu in India. He is the examination board member of various Colleges and Universities and He guided more than 6 MPhil Research scholars for various universities.

8. APPENDIX EXAMPLE OF ROBUST HEADER COMPRESSION FOR STREAM OF IPV4 PACKETS

```
root@ubuntu:/home/jitto/Downloads/rohc-2.0.0# gcc -o srpp -Wall $(pkg-config rohc --cflags) srpp.c $(pkg-config rohc --libs)
root@ubuntu:/home/jitto/Downloads/rohc-2.0.0# ./srpp

create the ROHC compressor

enable several ROHC compression profiles

 IP packet 1
0x45 0x00 0x00 0x1a 0x00 0x00 0x00 0x00
0x01 0x86 0xa9 0x3f 0x01 0x02 0x03 0x04
0x05 0x06 0x07 0x08 0x48 0x65 0x6c 0x6c
0x6f 0x20
size of packet 1 is 26 bytes

IP packet 2
0x45 0x00 0x00 0x15 0x00 0x00 0x00 0x00
0x01 0x86 0xa9 0x3f 0x01 0x02 0x03 0x04
0x05 0x06 0x07 0x08 0x41
size of packet 2 is 21 bytes

IP packet 3
0x45 0x00 0x00 0x16 0x00 0x00 0x00 0x00
0x01 0x86 0xa9 0x3f 0x01 0x02 0x03 0x04
0x05 0x06 0x07 0x08 0x42 0x43
size of packet 3 is 22 bytes

 IP packet 4
0x45 0x00 0x00 0x17 0x00 0x00 0x00 0x00
0x01 0x86 0xa9 0x3f 0x01 0x02 0x03 0x04
0x05 0x06 0x07 0x08 0x44 0x45 0x46
size of packet 4 is 23 bytes

 IP packet 5
0x45 0x00 0x00 0x15 0x00 0x00 0x00 0x00
0x01 0x86 0xa9 0x3f 0x01 0x02 0x03 0x04
0x05 0x06 0x07 0x08 0x47
size of packet 5 is 21 bytes

Consider the stream of IPv4 packets :1,2,3,4,5

total size of stream of IPv4 packets : 113 bytes

compress the stream of IP packets

ROHC packet resulting from the ROHC compression:
0xfc 0x00 0xb7 0x45 0x00 0x00 0x71 0x00
0x00 0x00 0x00 0x01 0x86 0xa9 0x3f 0x01
0x02 0x03 0x04 0x05 0x06 0x07 0x08 0x48
0x65 0x6c 0x6c 0x6f 0x20 0x41 0x42 0x43
0x44

size after ROHC is 33 bytes

The program ended successfully.

root@ubuntu:/home/jitto/Downloads/rohc-2.0.0#
```

9. APPENDIX EXAMPLE OF C PROGRAM IN LTE SIM CARDS TO ELIMINATE REDUNDANT PACKETS

1. Each signal monitored shall have a unique ID at all the instances this application is used.

For current implementation, the diseases and index are as follows:

Fever	0
Diabetes	1
Blood Pressure	2
Heart Rate	3

2. Units of the Signals are as follows

Fever	Deg cel
Diabetes	mg/dL
Blood Pressure	mmHg

Heart Rate	bpm

```
/* Update the signal_idxtype in the order as given in the internal requirements */
/*COUNT_E will have number of diseases configured*/

Enum
{
FEVER_E,
DIABATES_E,
BLOOD_PRESSURE_E,
HEART_RATE_E,
COUNT_E
}Signal_idxtype;

Struct
{
      Signal_idxtype input_sigIdx;
     Float signal_value;
}InpSig_type;

Struct
{
      Signal_idxtype limit_sigIdx;
     Float upper_limit;
Float lower_limit;

}Limit_type;

InpSig_type Input_signals[COUNT_E];

Const Limit_type Limit_table[]=
{
 FEVER_E,            35, 38,
DIABATES_E,            70, 120,
BLOOD_PRESSURE_E,       80, 120,
HEART_RATE_E,          60, 80,
};

Void Update_Signals(void);
```

```
Void Montior Signals(void);
Void main()
{
            Update_Signals();
Montior Singals();
}

Void Update_Signals(void)
{
            For (Signal_idxtype loop, loop<COUNT_E, loop++)
{
    Input_signals[loop].signal_value = ReadTheSignal(loop); // a dummy fnctn as actual fnctn not known

}

}
Void Montior Singals(void)
{
            For (Signal_idxtype loop1, loop1<COUNT_E, loop1++)
{
            For (int loop2, loop2<COUNT_E, int++)
{
            If(loop1== Limit_table. limit_sigIdx)
            {
            If((Input_signals[loop1].signal_value<= Limit_table[loop2].lower_limit) ||
    (Input_signals[loop1].signal_value>= Limit_table[loop2].upper_limit))
{
            SendtheSignal(); // Dummy function to send signal
}
Break;
 }
            }
            }
```