Online Load Estimation for Efficient Random Access of Machine Type Communications in LTE Networks

Bahereh Hassanpour and Abdorasoul Ghasemi
Faculty of Computer Engineering,
K. N. Toosi University of Technology
Tehran, Iran, 16315-1355.
bahereh.hassanpour@ee.kntu.ac.ir, arghasemi@kntu.ac.ir

Abstract—**Massive access of Machine to Machine (M2M) devices may lead to overload at the access of the LTE networks as the current most popular technology for this type of communications. Several approaches have been proposed to relieve the massive access by time barring or time distributing of access requesting. In virtual-frame technique the requests are spread in consecutive frames to increase the successful accesses. The length of virtual-frame, the number of LTE uplink frames which are concatenated for managing the random access of M2M requests, should be selected appropriately according to the network load to compromise the successful access rate and the incurred delay of each access. In this paper, we propose an adaptive scheme by which the eNB is able to find the length of the virtual-frame dynamically and broadcasts this information at the end of each virtual-frame. We use an online load estimation technique to adjust the length of virtual-frame. Simulation results show that using this scheme the length of virtual-frame is just increased when the access network load is high which prevents unnecessary delays for lightly loaded conditions.**

Keywords- Machine type communications, Random access channel, Overload control, Online load estimation, Virtual-frame, LTE network.

I. INTRODUCTION

Machine-to-Machine (M2M) or Machine Type communications is one of the most important revolutions in smart world technology [1, 2]. The nature of M2M traffic is inherently different from the traditional human to human data traffic where the M2M devices generally generate small packets with moderate to high frequencies and mostly require network uplink resources at the access network. Therefore, overload due to the massive access requests of M2M devices to utilize network resources is probable in both Random Access Network (RAN) and Core Network (CN) of the underlying infrastructure [3, 4].

Long-Term Evolution (LTE) networks are currently the most popular technology to accommodate the M2M communications by using enhanced radio interface technology [5, 6]. However, the explosive growth of machine type User Equipment (UE) which demand to have communications with different Quality of Service (QoS) requirements over LTE network raises new challenges in resource management in these networks especially at the RAN.

The first and probably the most important challenge is the concurrent massive access requests of UEs which leads to overload at the RAN. This overload consequently leads to undesirable packet loss rate, delays, and retransmission overhead [7]. The focus of this paper is on massive access problem in RAN due to the 2M communications. Several solutions have been proposed in recent years to tackle this problem taking into account the specifications of the LTE networks and M2M traffic characterization.

Slotted Aloha (SA) which is indeed the basis of random access procedure [6], is one of the main protocols to prevent deluge of random access requests occurrence. However, this protocol is inherent unstable and requires other mechanisms to be deployed in practical scenarios [8].

M2M traffic requires huge network signaling resources compared to the traditional Human-to-Human (H2H) communication to perform random access procedure in the Random Access Channel (RACH). A simple but static and inefficient scheme is pre allocation of time slots to M2M and H2H traffics to avoid the possible effects of M2M on H2H traffic [2].

Access Class Barring (ACB) is one of the basic solutions to reduce congestion in an overload condition [9, 10]. The ACB mechanism stabilizes the system to maximum throughput by using a probability p called ACB factor. At initiation of the random access procedure, the cell's eNB broadcasts p which is between $[0,1]$. To access to RACH, each UE must have generate a random number $q \in [0,1]$. UEs with $q \leq p$ have permission to send their requests to eNB, otherwise barred for a barring time duration. Due to the fact that in ordinary ACB, eNB manage the traffic surge with ACB factor, in massive access conditions, eNB sets p to the lowest available value which leads to unacceptable delay. Therefore, methods such as cooperative ACB and Extended Access Barring (EAB) are proposed to improve the ACB in different ways. In cooperative ACB, eNB of close cells, jointly decide about their ACB factors rather than deciding individually [11]. Thus traffic load share among eNBs in different cells and access delay significantly decreases. In [12], EAB proposed to serve

978-1-4673-8789-7/16/$31.00 ©2016 IEEE

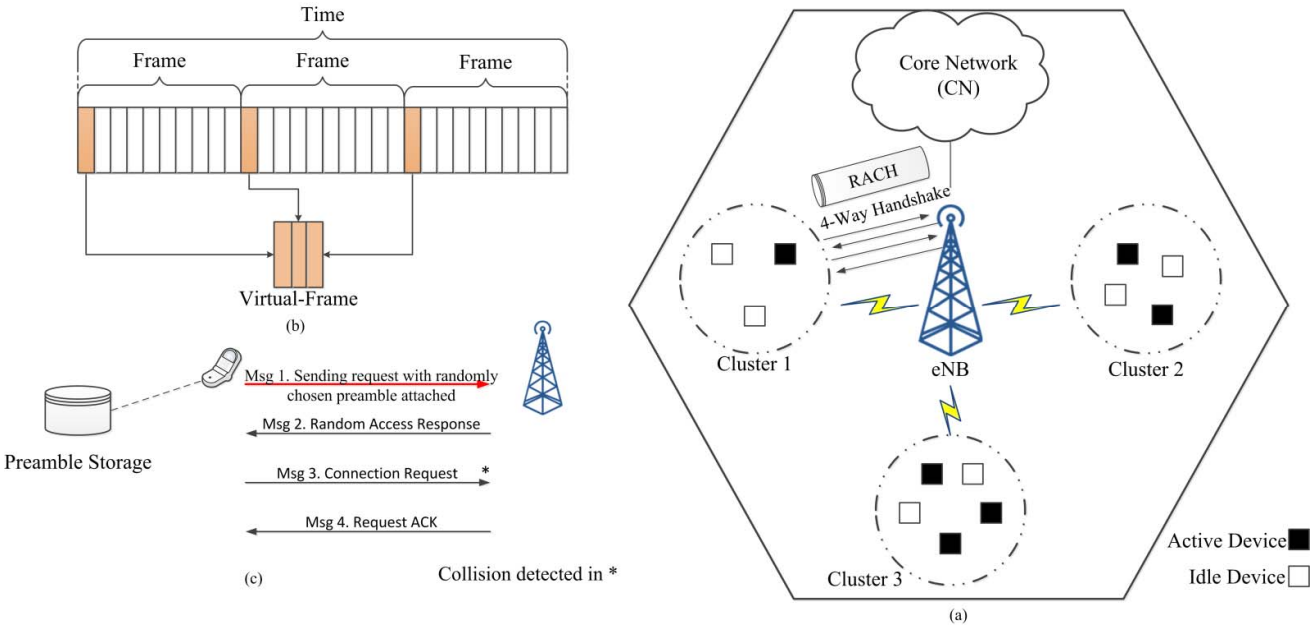

Fig.1. (a) RACH procedure in a cell with 3 clusters (b) The structure of sub-frame and virtual-frame

(c) 4-way handshake between UE and eNB

the delay-sensitive UEs by group based massive access management.

Despite the ACB based methods which are manage the collision resolution with preventing UEs from sending their requests, code-expanded random access method relieves the collision by virtually expanding the contention resources such as timeslots and preambles.

Code-expanded random access [13], proposed a mechanism by grouping the random access timeslots to virtual-frame. UEs have access to eNB at the end of each virtual-frame. In this mechanism random access procedure performs in each random access slot of virtual-frame and each UE sends a codeword to eNB according to the selected preambles for each M2M dedicated slot in the virtual frame. Transmitting codewords instead of preambles expands the number of random access resources which extremely improves the efficiency of the system in high load traffic at the cost of increasing the access delay.

The main issue in this mechanism is on adaptive adjustment of the length of virtual-frame taking into account the network load at the access. By proper adjustment of the length of virtual-frame the successful access rate is increased at heavy loaded scenarios without incurring access delay in moderate and lightly loaded conditions. For this purpose, eNB should infer the load of the system at the RAN. In this paper, we propose a scheme to improve code-expanded random access by using online load estimator at the eNB to adjust the length of virtual-frame according to the estimated expected load.

The rest of this paper is organized as follows. The system model and assumptions with a brief background are illustrated in sectionII. Adaptive virtual-frame length is discussed in section III followed by simulation results in section IV. Finally, section V concludes this work.

II. System Model and Backgrounds

We considered a cell of an LTE network in which M2M UEs attempt to transmit their data to eNB by contending on uplink radio resources as shown in Fig. 1 [14, 15]. The number of total UEs is denoted by N and it is assumed that these devices are distributed in G different clusters. Time is divided into static frame's length which consist of 10 sub-frames each with 1ms duration [2]. We assume that the first sub-frame of each frame is dedicated to M2M communications that is known as contention phase and the remainder is assigned to H2H traffic in the non-contention access phase. By attaching M2M sub-frame of some contiguous frames, the virtual-frame is constructed as it is shown in Fig. 1(a). M2M devices in active state generate traffic according to a given distribution. Let A_i and A_{g_i} denote the total number of active UEs and the number of active UEs in cluster g in the i-th virtual-frame respectively where we have $A_i = \sum_{g=1}^{G} A_{g_i}$.

In LTE networks the access to the eNB by UEs is performed through RACH in a 4-way handshaking [2]. In the first step, each active UE randomly chooses a preamble (including idle preamble) at the beginning of each sub-frame of the virtual-frame [13]. Therefore, at the end of the virtual-frame each user sends a codeword with the same length as virtual-frame length which is denoted by L.We assume that, there are M available preambles in each virtual-frame. Thus

the number of available codewords which is denoted by Y is $Y = (M+1)^L - 1$.

In the second step, eNB sends Random Access Response (RAR) which is consists of specific RB for each received codeword and some system information. UEs which receive RAR send their data on the announced RB in the third step. If two or more UEs choose the same codeword, collision occurs and eNB couldn't discern between their data and hence the data of all contended UE's are dropped. Machines that collided in current virtual-frame retransmit their request on the first available virtual-frame. In the fourth message eNB sends an ACK message to UEs which are successfully pass the third step.

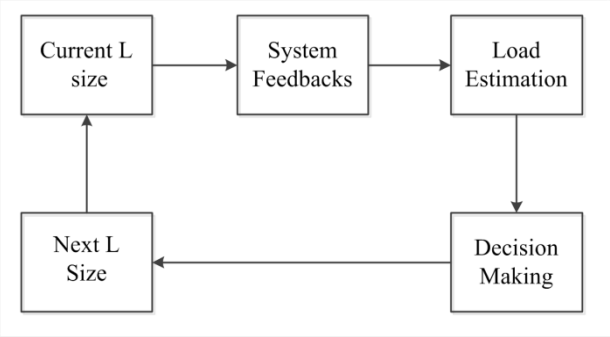

Fig.2. The block diagram of the proposed scheme for adaptive adjustment of virtual-frames' lengths according to the load at the access

In high load conditions the collision is more probable since more UEs select the same codeword. Therefore, the length of virtual frame should be selected adaptively according to the network load at the access as it is shown in Fig. 2.We assume that at the beginning, system starts with $L = 1$. Let X denotes the number of UEs that select codeword c. Preamble c is selected by k users simultaneously by the probability given in (1).

$$\mathrm{P_r}[X = k] = \binom{A_i}{k} \cdot \left(\frac{1}{Y}\right)^k \cdot \left(1 - \frac{1}{Y}\right)^{A_i - k} \quad (1)$$

Therefore, the expected number of codewords which are chose by just one UE is:

$$N_S = Pr[X = 1] \,.\, Y = A_i \,. \left(1 - \frac{1}{Y}\right)^{A_i - 1} \quad (2)$$

The expected number of successful codewords in specific virtual-frame is the same as the expected number of UEs with successful attempts and the probability that k UEs have successful attempts is the same as the probability that k codewords selected by just one UE. Efficiency of the system and the efficiency of using preamble are calculated by $E_{sys} = \frac{N_{S_i}}{A_i}$ and $E_{pre} = \frac{N_{S_i}}{X_i}$ respectively.

III. Adaptive Length of Virtual Frame

As despised in block diagram in Fig. 2, after receiving all the requests at the end of a virtual-frame, eNB finds the efficiency of the system by having system feedbacks such as A_i and N_{S_i}. The next sections elucidate the proposed load-estimating and decision-making scheme in details.

A. Online Load Estimation

The number of active UEs strongly depends on the network traffic in the previous virtual-frames. Hence to estimate the traffic load of the network at the end of each virtual-frame, we used online load estimators with low complexity at the eNB. We use a buffer with the length of β to save the information of the last β virtual-frames.

The estimated number of UEs at time t is denoted by $N'(t)$ which is calculated by estimation function $F(t,\beta)$ as denoted in (3) and (4) respectively. Where $F(t,\beta)$ deals with last β load traffic information from time $(t-\beta)$ to $(t-1)$ and $t \in \{1,2,\ldots . T_A\}$ where T_A is the last seen frame and $Q(j)$ denotes the buffer queue with $j \in \{1,2,\ldots,\beta\}$.

$$N'(t) = F(t,\beta) \quad (3)$$

$$Q(1{:}\beta) = Arr(t-\beta, t-1) \quad (4)$$

If we use small β then the estimated value correlated with a short duration of traffic load pattern and the accuracy of the estimated value will be reduced. Therefore, our estimated value has a large error on next frames after burst traffic occurrence in the system. On the other side, if we use a large β, then the estimated value loses its correlation with the previous traffic load pattern.

If we use small β then the estimated value correlated with a short duration of traffic load pattern and the accuracy of the estimated value will be reduced. Therefore, our estimated value has a large error on next frames after burst traffic occurrence in the system. On the other side, if we use a large β, then the estimated value loses its correlation with the previous traffic load pattern.

We evaluate the following six simple estimator functions to estimate the network.

1) Moving Average Filter (MAF): As shown in (5), by using this function, the estimated value for the number of UEs is equal to the average number of requests which are seen in the last β virtual-frames.

$$F_{MAF}(t,\beta) = \frac{1}{\beta} \sum_{i=t-\beta}^{t-1} Q(i) \quad (5)$$

2) Moving Median Filter (MMF): This function estimates the number of UEs at the next virtual-frame as the median of buffered values as illustrated in (6).

$$F_{MMF}(t,\beta) = \begin{cases} Q'\left(\frac{\beta+1}{2}\right) & ;if\ \beta\ is\ odd \\ \frac{Q'\left(\frac{\beta}{2}\right) + Q'\left(\frac{\beta}{2}+1\right)}{2} & ;\ if\ \beta\ is\ even \end{cases} \quad (6)$$

3) Moving Average plus Variance (MAV): In this estimator, we use arithmetic variance in addition to MAF in order to consider the traffic load pattern fluctuations as in (7).

$$F_{MAV}(t,\beta) = \frac{1}{\beta}[F_{MAF}(t,\beta) + \sum_{i=t-\beta}^{t-1}\left(Q(i) - F_{MAF}(t,\beta)\right)^2] \quad (7)$$

4) Maximum Simultaneous Requests (MSR): By using this estimator, the estimated value is set to the maximum number of requests which are seen in the last β virtual-frames as in (8).

$$F_{MSR}(t,\beta) = \max_{i\in\{t-\beta,\dots,t-1\}} Q(i) \quad (8)$$

5) Least Simultaneous Requests (LSR): As shown in (9), we used the least number of simultaneous requests which are registered in the buffer as the estimated UEs number for next virtual-frame.

$$F_{LSR}(t,\beta) = \min_{i\in\{t-\beta,\dots,t-1\}} Q(i) \quad (9)$$

6) Recent Seen Simultaneous Requests (RSR): In this function, the number of UEs which are requested in the last virtual-frame is assumed to the next estimated value. RSR function is calculated as (10).

$$F_{RSR}(t,\beta) = Q(t-1) \quad (10)$$

Two naturally desirable properties of estimators are being unbiased and have minimal mean squared error (MSE) which in general are not satisfied simultaneously by an estimator. The MSE of an estimator measures the average of the squares of the errors. The bias of an estimation is the distance between the average of the collection of estimates and the single parameter being estimated.

B. *Decision Making*

eNB should be able to establish a trade-off between delay and efficiency of the network. For this purpose decision making operation at the eNB should take into account the estimated traffic load and the tolerable delay. Thus here we defined $\alpha = \frac{N'(t)}{A_i}$ and $\gamma\epsilon[0.1,1]$ to reach a compromise. By setting A_i to the estimated valueand calculating Y per each available virtual-frame length, eNB chooses least α subject to $\alpha \geq \gamma$ and finds proper L consequently. When γ adjusted near 0.1, eNB expands the V-frame's length immediately which leads to increment in access delay. Also by setting it near to one, efficiency decreases and access delay reduces.

IV. Simulation and Results

We perform Monte Carlo simulations to evaluate the performance of the proposed scheme. In Fig. 3 the effect of virtual-frame length on the efficiency of the system is shown when the network parameters are $M = 6, T_A = 1$ and $1 \leq N \leq 30$. As expected by increasing the length of the virtual-frame the user efficiency is increased. In Fig. 4, the response time delay for each request is shown for the same scenario which indicates that for $N \leq 5$ increasing the length of the virtual-frame just impose further delay to the network access.

In Fig. 5 the efficiency of preamble usage against traffic load for different static length of virtual-frame is depicted which shows that increasing the length of virtual frame is just beneficial when the network load is high at the access. Using Fig. 3, 4 and 5 we find that eNB should broadcast the proper length of virtual frame to sustain the efficiency of the network access and at the same time avoids undesirable imposed delays.

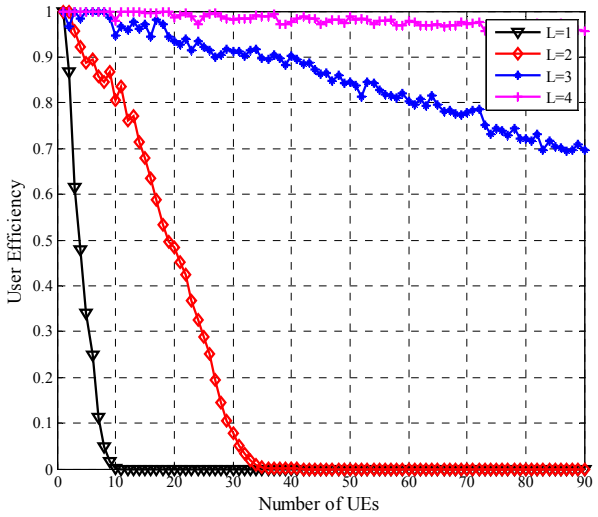

Fig.3. User Efficiency in different L size

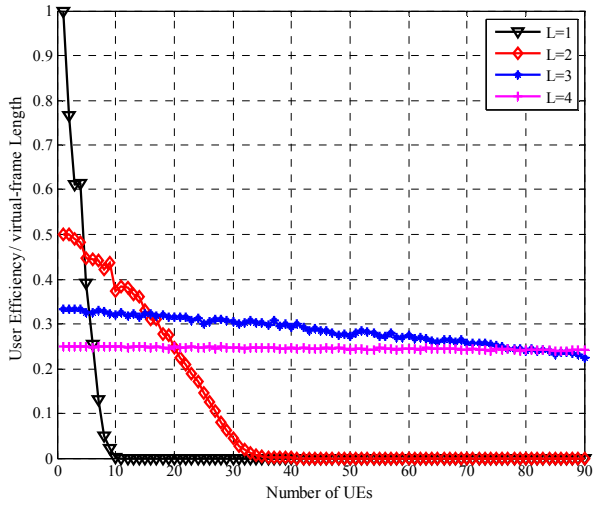

Fig.4. Tradeoff between Efficiency and Delay

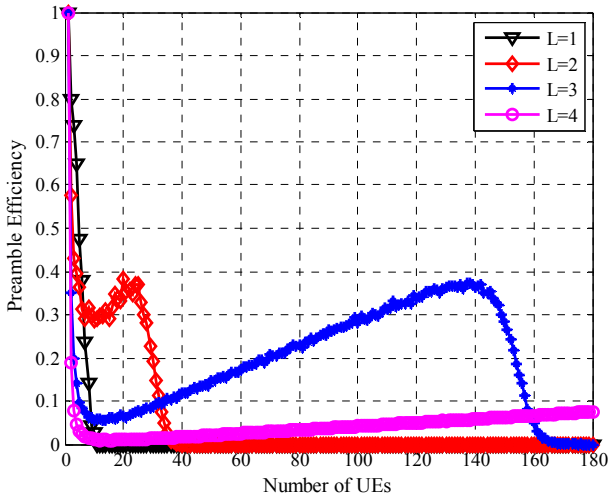

Fig.5. Preamble Efficiency in different L size

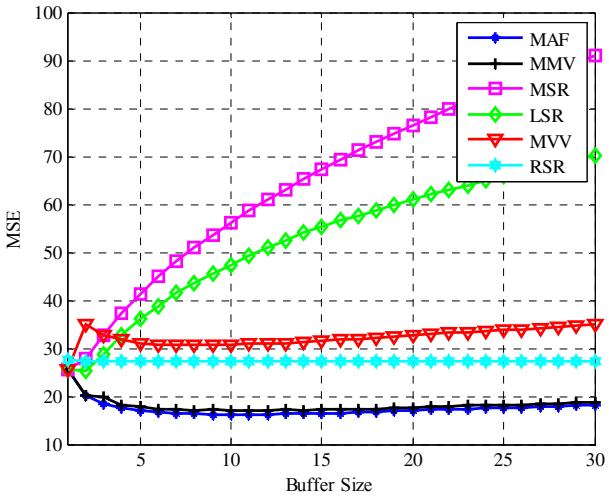

Fig.6. MSE of six estimators with different Buffer Size

To justify the proposed scheme in both low and burst traffic load conditions, we consider three different types of UE clusters which send requests to eNB with different patterns. Cluster1 which has half of the users in the cell generates beta distributed traffic, cluster 2 with quarter of users generates uniform distributed traffic, and the last cluster generates traffic with poison distribution.

Here we used aggregated traffic model when $N = 20000$ UEs are activated during 1200 frames, $\gamma = 0.1$ and $M = 6$. Due to generating the traffic in each frame and changing the length of virtual-frame dynamically, UEs which are activated between virtual-frame $t-1$ and t are queued. We assume that backlogged UEs retransmit their requests on the next available virtual-frame.

Therefore, at the beginning of each virtual-frame the new activated UEs along with queued active users in previous frames and backlogged UEs from last virtual-frame contending to get access to RACH. Fig. 6 and Fig. 7 show, respectively, the MSE and bias of different load estimators when the length of buffer changes from 1 to 30. From these figures we conclude that the AM estimator with $\beta = 10$ has the minimum MSE and bias.

Finally, we compare the efficiency of the system using the proposed scheme with a scenario in which eNB knows the actual amount of traffic in the next virtual-frame in Fig. 8. The results show that by estimating the network load at the RAN and dynamically adapting the length of the virtual-frame we can sustain the performance of the system near the optimal performance (The average difference between results of proposed scheme and optimal performance in 1200 frames is 3.48%). For more transparency only the first 200 frames shown in Fig.8.

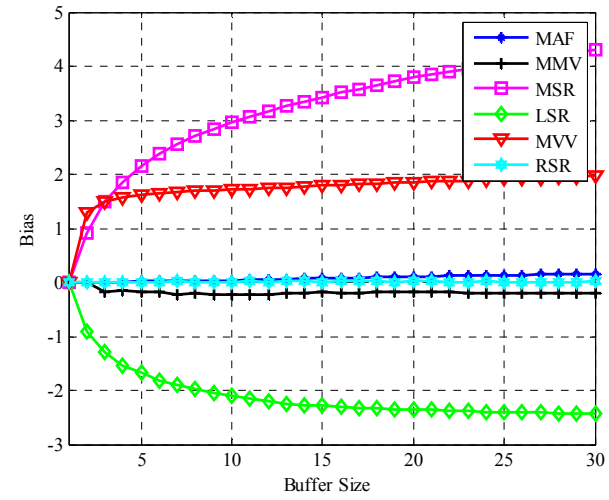

Fig.7. Bias of estimators with different Buffer Size

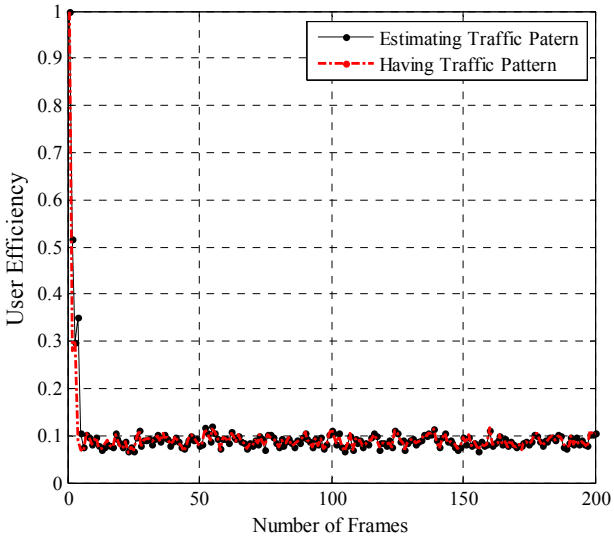

Fig.8. Efficiency of the system with knowing traffic pattern and estimating pattern

V. CONCLUSION

In this paper, an online load estimation scheme is used in random access procedure within the LTE networks in order to sustain the efficiency of the system which deploys code-expanded scheme to relive the massive access. In the proposed scheme the cell's eNB uses a traffic load estimation to estimate the number of requests in next virtual-frame for proper adjustment of the length of next virtual-frame in the code-expanded scheme. Six simple estimators are evaluated by simulations and it is shown that for a scenario with mixed beta, uniform, and Poisson distributed access arrival request the moving average filter shows acceptable performance in terms of mean square error and bias criteria.

REFERENCES

[1] F. Ghavimi, H. Chen, "M2M Communications in 3GPP LTE/LTE-A Networks: Architecture, Service Requirements, Challenges, and Applications", *IEEE Communication. Surveys Tutorials*, vol. 17, no. 2, pp. 525–549, May. 2015.

[2] A. Laya, L. Alonso, and j. Zarate, "is the random access channel of LTE and LTE-A suitable for M2M communication? A survey of alternatives", *IEEE Communication. Surveys Tutorials*, vol. 16, no. 1, pp. 4–16, Feb. 2014.

[3] U. Phuyal, A. T. Koc, M. H. Fong, and R. Vannithamby, "Controlling Access Overload and Signaling Congestion in M2M Networks", *Proc.IEEE ASILOMAR,* pp. 591–595, Nov. 2012.

[4] A-H. Tsai, L_C. Wang, J-H. Huang and T-M. Lin, "Overload Control for Machine Type Communications with Femtocells", *Proc. IEEE VTC,* pp. 1-5,Sept. 2012.

[5] Akyildiz IF, Estevez DMG, Reyes EC. "The evolution to 4G cellular systems: LTE-advanced", *PhysCommun,*vol. 3, no. 4, pp. 217-244, Oct. 2010.

[6] P. Osti, P. Lassila, S. Aalto, A. Larmo, T. Tirronen, "Analysis of PDCCH performance for M2M traffic in LTE", in *IEEE Transactions on Vehicular Technology*, vol. 63, no. 9, Apr. 2014.

[7] S. Dharmaraja, V. Jindal, and U. Varshney, "Reliability and survivability analysis for UMTS networks: An analytical approach", *IEEE Trans. Netw. Service Manage.,*vol. 5, no. 3, pp. 132–142, Sept. 2008.

[8] S.-Y. Lien, K.-C. Chen, and Y. Lin, "Toward ubiquitous massive accesses in 3GPP Machine-to-Machine Communications", *IEEE Communications. Magazine*, vol. 49, no. 4, pp. 66-74, Apr. 2011.

[9] 3GPP TR 37.868 V11.0, Study on RAN Improvements for Machine-type Communications, Oct. 2011.

[10] J.-P. Cheng, C. han Lee, and T.-M. Lin, "Prioritized random access with dynamic access barring for RAN overload in 3gpp LTE-A networks", *GLOBECOM Workshops*, pp. 368 –372, Dec. 2011.

[11] S.-Y. Lien, T.-H. Liau, C.-Y. Kao, and K.-C. Chen, "Cooperative Access Class Barring for Machine-to-Machine Communications", *Wireless Communications*, *IEEE Transactions on*, vol. 11, no. 1, pp. 27-32, Jan. 2012.

[12] U. Phuyal, A. T. Koc, Mo-Han Fong, R. Vannithamby, "Controlling access overload and signaling congestion in M2M networks", *in Proc. Conference Record of the Forty Sixth Asilomar Conference on Signals, Systems and Computers* (ASILOMAR) 2012, pp. 591-595, Nov. 2012.

[13] N.K. Pratas, H. Thomas, C. Stefanovic, P. Popovski, "Code-Expanded Random Access for Machine-Type Communications", *in IEEE Globcome Workshop*, pp. 1681-1686, 2012.

[14] 3GPP TS 36.212, Multiplexing and channel coding.

[15] 3GPP TS 36.211, Physical Channels and Modulation.