Automatic Configuration of Random Access Channel Parameters in LTE Systems

*Seunghyun Choi, *Wonbo Lee, *Dongmyoung Kim, †Kyung-Joon Park, *Sunghyun Choi, and ‡Ki-Young Han
*School of Electrical Engineering and INMC, Seoul National University, Korea
†Department of Information and Communication Engineering, DGIST, Korea
‡System Lab., Next Generation System Team, Samsung Electronics, Korea
Email: {shchoi, wblee, dmkim}@mwnl.snu.ac.kr, kjp@dgist.ac.kr, schoi@snu.ac.kr, kiyoung.han@samsung.com

***Abstract*—In 3G long term evolution (LTE) systems, the random access channel (RACH) is used for initial access, resource request, and handover. Since the random access delay is determined by the arrival rate of the random access preambles and the number of RACH subframes, we should configure the number of RACH subframes given the arrival rate in order to guarantee the delay performance. In this work, by carefully taking account of a tradeoff between the number of RACH subframes and the random access delay, we present an optimization formulation that minimizes the number of RACH subframes for a given delay requirement. Furthermore, since the arrival rate of the random access preambles is time varying in reality, we further propose an estimation scheme for the arrival rate by reflecting the periodicity and the correlation between recent and future arrival rates. Our simulation results show that the proposed scheme for tuning the RACH subframes gives very promising network performance under time-varying environments.**

I. Introduction

The recent surge of increase in the smart-phone usage has significantly changed the traffic amount and pattern of cellular networks. Most of smart-phone applications are network applications, which often transmit various types of traffic even without user's perception. Because of this frequent access to the network, the whole amount of access requests in a given cell is very rapidly increasing, and it is required to carefully configure network parameters for proper accommodation of the access requests.

In particular, in LTE systems, random access channel (RACH) is responsible for achieving uplink time synchronization for a user which either has not yet acquired, or has lost its uplink synchronization [1]. Scenarios in which the RACH is used include initial access, resource request, and handover. When multiple users transmit the same random access preamble code in the same 'RACH subframe' (or more correctly, subframe used for RACH), those transmissions collide and they should retransmit their random access preambles, which in turn causes a delayed access. To satisfy the delay requirement of users, the cell should use a certain number of RACH subframes. One may think that a cell can use more RACH subframes in order to decrease the average delay. However, there exists a tradeoff between the RACH subframes and subframes for data transmission. If a specific subframe is used for RACH, a part (corresponding to 1.08 MHz bandwidth) of the subframe cannot be used for data transmission [1]. Hence, it is crucial to determine the number of RACH subframes in order to optimize the overall network performance while satisfying a given delay requirement.

978-1-4577-2028-4/11/$26.00 ©2011 IEEE

Furthermore, the network usage pattern is time varying in reality. For example, in business areas the number of users increases during daytime, and decreases after office hours. On the other hand, in residential areas the number of users decreases during daytime, and increases during the night. Therefore, a network should be able to detect the change of the arrival rate of the random access preambles, and adapt the number of RACH subframes accordingly.

In this paper, by modeling the tradeoff between RACH and data transmission in a quantitative manner, we propose an efficient method for selecting the number of RACH subframes for a given arrival rate of the random access preambles. Then, we present an optimization framework to estimate the arrival rate of the random access preambles and adaptively tune the number of RACH subframes. Our simulation results show that the proposed framework achieves a promising performance for RACH adaptation with time-varying traffic. To the best of our knowledge, this is the first work on exploiting the periodic information of RACH traffic pattern to dynamically configure the number of RACH subframes. We highlight the contributions of our study as 1) quantifying the efficiency of LTE RACH, 2) determining the number of RACH subframes according to the arrival rate of the random access preambles, and 3) estimating the arrival rate by exploiting both periodic and current traffic information.

The rest of the paper is organized as follows: we briefly explain the random access procedure in Section II, and then present our RACH performance analysis in an optimization framework in Section III. In Section IV, we propose a framework for the estimation of the arrival rate of random access preambles. We present our simulation results in Section V. Our conclusion follows in Section VI.

II. Random Access Procedure

The LTE random access procedure comes in two forms, i.e., contention-based or contention-free. Basically, a user initiates its random access procedure in a contention-based manner by

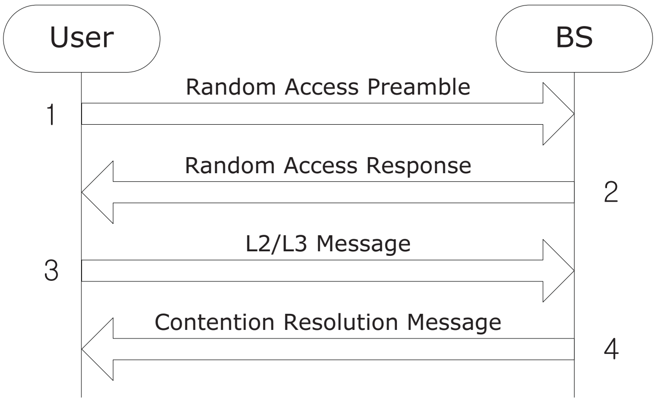

Fig. 1. Contention-based random access procedure.

randomly choosing a code. However, when a user needs to request resources for ACK/NACK transmission over the uplink or needs to handover, the cell has an option of preventing contentions by allocating a dedicated code to a user, resulting in contention-free access. Out of 64 codes available in each cell, N_{cf} codes are reserved for contention-free access while the rest $(64-N_{cf})$ codes are used for contention-based access.

The contention-based procedure consists of the following four steps as shown in Fig. 1 [1]:

1) **Random access preamble:** A user selects one of the $64 - N_{cf}$ available RACH codes.
2) **Random access response:** Random access response (RAR) is sent by the base station (BS) on physical downlink shared channel (PDSCH), which is the main data-bearing downlink channel. RAR conveys the identity of the detected preamble, a time alignment instruction, and an initial uplink resource grant for an L2/L3 message. If multiple users collided by selecting/transmitting the same code, they would receive the same RAR.
3) **Layer 2/layer 3 (L2/L3) message:** This message is the first scheduled uplink transmission on the physical uplink shared channel (PUSCH). It conveys the actual random access procedure message, such as an initial access request or a resource request. In case of a preamble collision at the first step, the colliding users receive the same RAR, and hence, their L2/L3 messages will also collide.
4) **Contention resolution message:** This message echoes the user identity contained in the successfully received L2/L3 message. A user which receives no contention resolution message containing its own identity learns that there was a collision, exits the current random access procedure, and starts another one after random back-off.

On the other hand, the contention-free random access reduces the random access delay by allocating a dedicated code to a user on a per-need basis. Therefore, the contention-free random access procedure starts with a random access preamble assignment by a BS. Then, the user transmits the assigned random access preamble. Finally, the procedure terminates with an RAR.

In this paper, we consider only the initial access and uplink resource request scenarios, not the handover and downlink resource request scenarios. Therefore, only the contention-based procedure is considered.

III. Analysis of RACH in LTE Systems

A. Collision probability of a random access preamble

We assume a frequency division duplexing (FDD) mode LTE system since it is widely used in LTE systems. 10 ms-long frames repeat over time, where each frame is composed of 10 subframes. For the FDD mode LTE system, if we denote the number of RACH subframes per frame by n, then we have $n \in \{0.5, 1, 2, 3, 5, 10\}$. Here, n equal to 0.5 means that there is one RACH subframe per every two frames. Since these RACH subframes are almost evenly separated, the interval between two adjacent RACH subframes can be approximated as $10/n$.

When more than one user transmits random access preambles using the same code in the same RACH subframe, those transmissions collide and they retransmit their random access preambles. If we assume that the arrival of the random access preambles follows Poisson distribution [2] with a rate λ_0, and that a maximum of L retransmissions is allowed, then the relation between the new arrival rate λ_0 and the aggregate arrival rate λ_t (first time preamble arrival plus a certain number of retransmissions arrival) is given as follows [3–6]:

$$\lambda_0 = \frac{\lambda_t e^{-\lambda_t}}{1 - \{1 - e^{-\lambda_t}\}^{L+1}}. \tag{1}$$

By considering that the arrival rate λ_t is the same for all the subframes, the average number of arrivals in m subframes becomes $m\lambda_t$. Because the interval between two adjacent RACH subframes is $10/n$, there are $10\lambda_t/n$ random access preambles in each RACH subframe on average. We denote the number of random access preambles per RACH subframe, i.e., the RACH load, as G. When the number of codes for RACH is q, the success probability of each random access $P_s(G,q)$ can be represented as follows:

$$\begin{aligned} P_s(G,q) &= \sum_{i=1}^{\infty} \left\{ P_{s|i} \cdot P_i \right\} \\ &= \sum_{i=1}^{\infty} \left\{ \left(\frac{q-1}{q}\right)^{i-1} \cdot \frac{i e^{-G} \frac{G^i}{i!}}{\sum_{j=1}^{\infty} \left(j e^{-G} \frac{G^j}{j!} \right)} \right\} \\ &= \frac{\sum_{i=1}^{\infty} \left\{ \left(\frac{q-1}{q}\right)^{i-1} \frac{G^{i-1}}{(i-1)!} \right\}}{\sum_{j=1}^{\infty} \frac{G^{j-1}}{(j-1)!}} \\ &= e^{-G/q}, \end{aligned} \tag{2}$$

where $P_{s|i}$ is the success probability of a random access given that there are i random access preambles in the RACH subframe and P_i is the probability that there are i random access preambles in the RACH subframe. Hence, we can express the collision probability of a random access preamble as

$$P_c(G,q) = 1 - e^{-G/q} \\ = 1 - e^{-10\lambda_t/nq}. \quad (3)$$

B. Optimization formulation

As already mentioned in the previous section, there is a tradeoff between the number of RACH subframes and the random access delay. If a cell allocates more subframes for RACH, the collision probability of a random access preamble decreases according to (3). The expected random access delay $\overline{D}$ can be expressed using the collision probability of a random access preamble as follows:

$$\overline{D} = \sum_{k=1}^{L+1} k\overline{D}_b {P_c}^{k-1}(1-P_c), \quad (4)$$

where $\overline{D}_b$ is the average delay per retransmission. According to (4), the decrease in the collision probability in turn results in the decreased random access delay.

On the other hand, as the number of RACH subframes increases, the number of subframes that can be used for data transmission decreases. Therefore, each cell can determine its delay performance and the amount of resources for data transmissions by adjusting the number of RACH subframes.

Here, we propose an optimization formulation based on the analysis of the RACH in the previous section. In the formulation, we minimize the number of RACH subframes as long as the expected access delay is smaller than the given delay bound as follows:

$$\begin{aligned} \min \quad & n \\ \text{s.t.} \quad & n \in \{0.5, 1, 2, 3, 5, 10\} \\ & \overline{D} \leq \overline{D}_{max}, \end{aligned} \quad (5)$$

where $\overline{D}_{max}$ is the maximum delay bound. Consequently, we can rewrite the overall optimization problem as follows:

$$\begin{aligned} \min \quad & n \\ \text{s.t.} \quad & n \in \{0.5, 1, 2, 3, 5, 10\} \\ & \sum_{k=1}^{L+1} k\overline{D}_b {P_c}^{k-1}(1-P_c) \leq \overline{D}_{max}. \end{aligned} \quad (6)$$

Since P_c is a function of λ_0 from (1) and (3), we can determine the number of RACH subframes for a given arrival rate λ_0.

IV. Adaptation to a Time Varying Environment

A. Motivation for estimating the arrival rate

In the previous section, we have formulated an optimization problem that minimizes the number of RACH subframes satisfying a given delay constraint. With a given λ_0, each cell solves the optimization problem, and can determine the number of RACH subframes. However, the arrival rate of random access preambles may change in practice. For example, in business areas, the number of users increases during daytime, and decreases after office hours. As the number of users changes, the arrival rate of random access preambles also changes with a high probability. Thus, we need to iteratively optimize the number of RACH subframes. To this end, there should be a technique to detect a meaningful change of the arrival rate of the random access preambles.

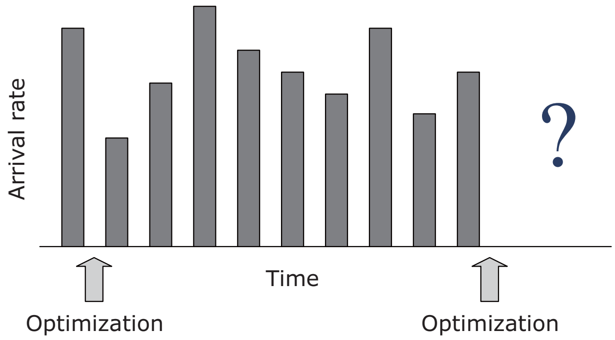

Fig. 2. Arrival rate of random access preambles to be estimated.

Meanwhile, it should be noted that the arrival rate λ_0 that is used for determining the number of RACH subframes is not the current arrival rate but the arrival rate in the near future. In Fig. 2, vertical bars represent the arrival rate of random access preambles at each time interval. At the point where the first optimization is performed, the optimization problem should use the arrival rate which represents the period between the point where the first optimization is performed and the point where the next optimization is performed. Thus, the arrival rate λ_0 used for the optimization problem should be the average value of the next period. By using the arrival rate in the upcoming period, we can solve the optimization problem, and then determine the number of RACH subframes. Consequently, the remaining issue is how to predict or estimate the arrival rate in the next period.

B. Baseline estimation model

Before we propose an estimation scheme, we present two baseline models, which are the basic building blocks for our overall estimation framework.

1) Profile-based scheme: Time series of count data measures the aggregated behavior of individual human beings. Even though the behavior of individual does not show periodicity, the aggregated behavior may exhibit a certain periodicity that reflects the rhythms of the underlying human activity. This underlying human activity usually follows a weekly profile, and from the profile, one can expect that upcoming data will be similar to the profile value with a high probability [7].

By using the periodic characteristics of human activities, we build up a weekly profile of the random access preambles. We first average the arrival rate of the random access requests every week at a specific time and day. Through the profile, each cell can estimate the current and future arrival rates.

However, this profile-based scheme has its own limit in that it cannot reflect the trend of a specific week. That is, the arrival rate of the random access requests in a given week can be quite different from the profile value. In this case, the

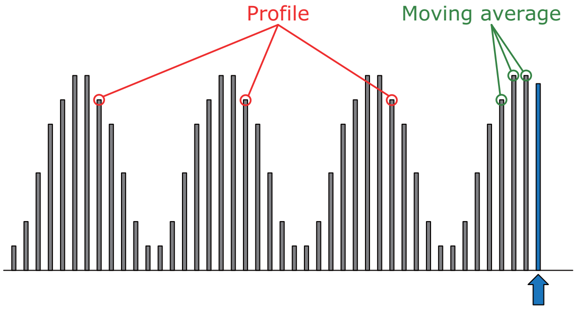

Fig. 3. Characteristics of profile and moving average.

actual arrival rate can be quite different from the estimated arrival rate. Consequently, the number of RACH subframes determined based on the estimation scheme may be inaccurate. Therefore, along with the profile-based estimation, we need to reflect the current trend of the arrival rate of the random access preambles.

2) Moving average scheme: In order to reflect the current trend of the arrival rate of the random access preambles, we introduce a moving average scheme. A moving average is a type of finite impulse filter used to analyze a set of data points by creating a series of averages. By using a moving average scheme, we can estimate the current arrival rate of random access preambles recursively. Especially, we use an exponential moving average defined as follows [8]:

$$E(t) = \sigma y(t) + (1-\sigma)E(t-1), \tag{7}$$

where $E(t)$ is an estimated value at time $t+1$. That is, $E(t)$ is estimated at time t, and the value is for time $t+1$. Hence, each cell estimates the arrival rate that will show up in the next interval that is needed to optimize the RACH parameters. Here, $y(t)$ is the actual value at time t, and $\sigma \in [0,1]$ is a weighting factor of the current data value. Therefore, the estimated value depends more on the current data value as σ increases. Eq. (7) calculates an estimation value recursively, by using the previously estimated value.

C. Proposed framework

As we explained in the previous section, a profile uses periodic characteristics of human activities, and a moving average reflects the current trend of data. In a similar manner, we can assume that the arrival rate of the random access preambles follows a periodic profile, and the current trend can be obtained by a moving average scheme. In Fig. 3, the arrow on the right side of the figure shows the point where estimation should be performed. As in the figure, each profile value contains periodic information, and a moving average value contains the current trend information. Hence, in order to incorporate the periodic characteristic and the temporal behavior at the same time, we consider the profile and moving average together. We estimate the arrival rate by using the following equation:

$$E_{proposed} = \omega E_{profile} + (1-\omega)E_{ma}, \tag{8}$$

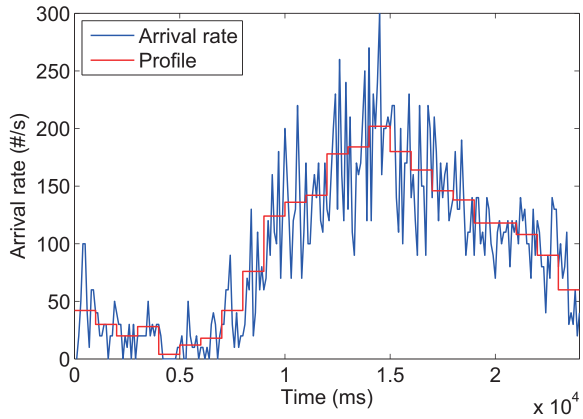

Fig. 4. Profile and the arrival rate of random access preambles.

where $E_{profile}$ and E_{ma} are the estimated value from the profile-based scheme and the moving average scheme, respectively, and $\omega \in [0,1]$ is a weighting factor. With the proposed framework, we can estimate the arrival rate of the random access preambles as $E_{proposed}$.

V. Performance Evaluation

In this section, we incorporate the proposed estimation scheme into the optimization formulation. Here, we consider both the number of RACH subframes and the access delay as performance metrics.

To our best knowledge, no real data on the random access preambles is publicly available. Hence, we use Wi-Fi session tracking data which is obtained from APs in Montreal [9], which are collected across three consecutive years from 2005 to 2007. In a sense that Wi-Fi session opening also reflects the rhythms of the communication activities, it is reasonable applying the data to our simulation. Since the rate of Wi-Fi session opening is not high enough to see collisions in RACH, the time dimension is scaled down from 24 hours to 24 s.

Our simulation parameters are given as follows. We set back-off window size as 80 subframes. Coefficient σ of the moving average and weighting factor ω of the proposed estimation scheme are set to 0.5, respectively. The number of codes used for contention-based random access is 32, and the delay bound is set to 0.4 ms.

Fig. 4 shows both the arrival rate of a specific day and the profile. As shown in the figure, the arrival rate of the random access preambles shows quite a similar value to the profile in an average sense. As we have assumed in the previous section, this observation explains the periodic pattern of human activities. At the same time, we can also find in Fig. 4 that the instant arrival rate varies frequently and the values are quite deviated from the profile.

Fig. 5 shows the number of RACH subframes used in each frame. We compare the proposed scheme with the moving average scheme and the fixed scheme, which uses a constant number of RACH subframes per frame. By comparing with Fig. 4, we observe that the trend of the graph in Fig. 5 follows

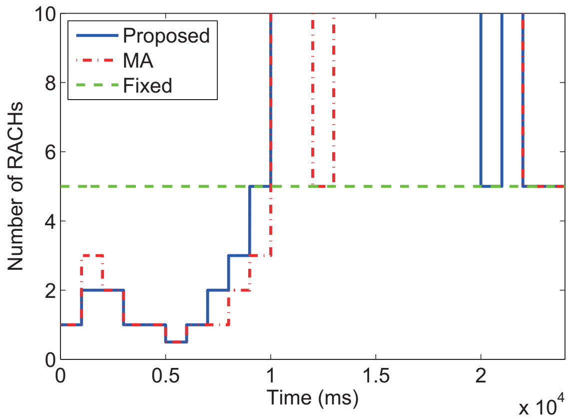

Fig. 5. The number of RACH subframes used per frame.

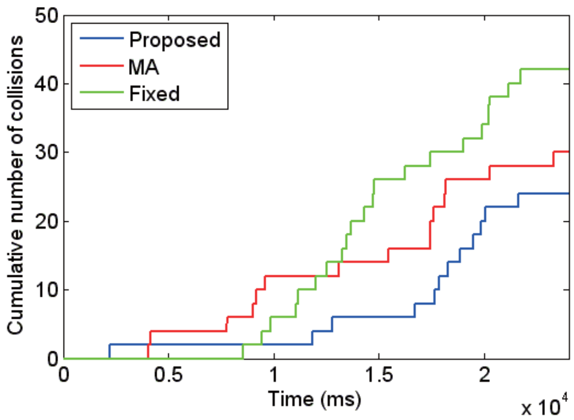

Fig. 6. Cumulative number of collisions.

that of the arrival rate for the cases of the proposed and the moving average schemes. We can also notice that the number of RACH subframes between 5,000 and 10,000 ms is smaller in the moving average case. This phenomenon results from the fact that the proposed scheme can anticipate from the profile that the arrival rate will increase. On the other hand, the pure moving average scheme does not have the profile information, and shows relatively slow adaptation to the traffic change.

As shown in Fig. 6, the moving average scheme results in more collisions than the proposed scheme when the arrival rate increases rapidly. This is because the moving average scheme underestimates the arrival rate. In case of the fixed scheme, there occur few collisions when the arrival rate is low because the cell uses more RACH subframes than it needs. On the other hand, we can find that the number of collisions rapidly increases when the arrival rate becomes higher.

We present the average number of RACH subframes used per frame and the average delay performance in Table I. The table uses three consecutive month traffic data. As can be seen in the fixed scheme, there exists a tradeoff between the average number of RACH subframes and the average delay. When the number of RACH subframes increases, the average delay

TABLE I
PERFORMANCE COMPARISON AMONG THE ESTIMATION SCHEMES

	Average number of RACH subframes (/frame)	Average delay (ms)
Proposed	5.98	0.36
MA	5.85	0.42
Fixed	0.5	8.82
	1	3.87
	2	1.34
	3	0.79
	5	0.53
	10	0.26

decreases. Therefore, with the fixed scheme, each cell should use 10 subframes for RACH in order to satisfy the delay bound of 0.4 ms. In case of the proposed scheme, on the other hand, the average delay is 0.36 ms, which satisfies the delay bound. In the meantime, the number of RACH subframes is 5.98 per frame, which is smaller than the case of the fixed scheme. This result clearly shows that adaptation to the time varying arrival rate outperforms the fixed scheme.

When the cell uses the moving average scheme, the average delay is 0.42 ms, which exceeds the given delay bound while the number of RACH subframes is smaller than the case of the proposed scheme.

VI. Conclusion

In this paper, we have proposed an adaptive algorithm for determining the number of RACH subframes per frame considering the time varying arrival rate of the random access preambles. First, we have numerically analyzed the collision probability of the random access request, and presented an optimization framework that minimizes the number of RACH subframes with a given delay constraint. Then, in order to take into account the time-varying rate of the random access preambles, we have introduced a periodic update algorithm for our optimization. To this end, by using both periodic information from a profile and the current information, we have proposed a scheme that can efficiently estimates the arrival rate of the random access preambles.

In our simulation study, we have comparatively evaluated the performance of the proposed scheme in terms of the number of RACH subframes per frame and access delay. By comparing the proposed scheme to a scheme that uses a fixed number of RACH subframes, we have shown that the proposed scheme uses a fewer number of RACH subframes than the scheme with a fixed number of RACH subframes while satisfying the given delay bound. We have also compared the proposed scheme with a scheme employing a moving average for traffic estimation. Since the proposed scheme exploits the periodic information from a profile, its access delay is relatively lower than the moving average scheme whenever the arrival rate of the random access preambles abruptly increases. Based on the evaluation, We conclude that the proposed framework for the self configuration of LTE RACH parameters can be a very useful scheme for improving the overall performance of LTE systems.

ACKNOWLEDGMENT

This research was supported in part by the KCC (Korea Communications Commission), Korea, under the R&D program supervised by the KCA (Korea Communications Agency) (KCA-2011-08913-04003), and in part by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (No. 2011-0017027).

REFERENCES

[1] S. Sesia, I. Toufik, and M. Baker, *The UMTS Long Term Evolution*, West Sussex, UK: John Wiley & Sons, 2009.

[2] S. Bregni and L. Primerano, "The Modified Allan Variance as Time Domain Analysis Tool for Estimating the Hurst Parameter of Long Range Dependent Traffic," in *Proc. IEEE GLOBECOM*, Dallas, USA, Dec. 2004.

[3] J. H. Sarker and S. J. Halme, "Optimizing the Use of Random Access Channels in GSM-GPRS," *Wireless Personal Communications*, vol. 22, no. 3, pp. 387–408, Sep. 2002.

[4] C. Jain and D. J. Goodman, "General Packet Radio Service in GSM," *IEEE Communications Magazine*, vol. 35, no. 10, pp. 122–131, Oct. 1997.

[5] G. Brasche and B. Walke, "Concepts, Services, and Protocols of the New GSM Phase 2+ General Packet Radio Service," *IEEE Communications Magazine*, vol. 35, no. 8, pp. 94–104, Aug. 1997.

[6] M. Mouly and M. B. Pautet, *The GSM System for Mobile Communications*, Telecom Publishing, 1992.

[7] A. Ihler, J. Hutchins, and P. Smyth, "Adaptive Event Detection with Time-Varuing Poisson Processes," in *Proc. ACM SIGKDD*, Philadelphia, USA, Aug. 2006.

[8] L. Sun, Y. Gao, H. Tian, H. Xu, and P. Zhang, "An Adaptive Random Access Protocol for OFDMA System," in *Proc. IEEE VTC-2007 Fall*, Baltimore, USA, Oct. 2007.

[9] M. Lenczner, B. Gregoire, and F. Proulx. (2007, Aug.) Dataset of user session traces collected from wi-fi hotspots for three years. [Online]. Available: http://www.crawdad.org/data.php