Preamble Barring: A Novel Random Access Scheme for Machine Type Communications with Unpredictable Traffic Bursts

Maxime Grau, Chuan Heng Foh, Atta ul Quddus, Rahim Tafazolli
5G Innovation Centre (5GIC), Institute for Communication Systems (ICS)
University of Surrey
Guildford, GU2 7XH, U.K.
Email: {m.d.grau, c.foh, a.quddus, r.tafazolli}@surrey.ac.uk

***Abstract*—In this paper, we present a novel random access method for future mobile cellular networks that support machine type communications. Traditionally, such networks establish connections with the devices using a random access procedure, however massive machine type communication poses several challenges to the design of random access for current systems. State-of-the-art random access techniques rely on predicting the traffic load to adjust the number of users allowed to attempt the random access preamble phase, however this delays network access and is highly dependent on the accuracy of traffic prediction and fast signalling. We change this paradigm by using the preamble phase to estimate traffic and then adapt the network resources to the estimated load. We introduce Preamble Barring that uses a probabilistic resource separation to allow load estimation in a wide range of load conditions and combine it with multiple random access responses. This results in a load adaptive method that can deliver near-optimal performance under any load condition without the need for traffic prediction or signalling, making it a promising solution to avoid network congestion and achieve fast uplink access for massive MTC.**

***Index Terms*—Random Access, massive MTC, IoT, Traffic bursts, Load estimation**

I. Introduction

UPCOMING 5G and beyond networks are expected to accommodate millions of user devices with a sporadic traffic pattern generated by Machine Type Communications (MTC), a major feature of the Internet of Things (IoT) paradigm. This represents a radical shift in the network load, and Random Access (RA), i.e. initial access to the system, has been identified as the main bottleneck [1]–[3].

3GPP Long Term Evolution (LTE) RA procedure typically consists of 4 steps described in Fig. 1: 1) users randomly choose a signature, referred to as simply preamble in the remainder on this paper, among N available ($N = 54$ in LTE [4]) and send the corresponding preamble; 2) the Base Station (BS) sends a RA Response (RAR) message for each detected preamble with an uplink grant; 3) users respond to RAR message and initiate a connection phase with the BS. If two or more users have chosen the same preamble at step 1, a collision occurs, otherwise the preamble is considered successful; 4) the BS confirms the connection or notifies a collision. In this paper, step 1 is referred to as the Slotted-ALOHA (S-ALOHA) preamble phase where users randomly choose a preamble. The maximum throughput of preamble phase follows that of S-ALOHA throughput which is $e^{-1} \simeq 0.37$ when the offered load is $k/N = 1$, where k is the number of users attempting RA. The other steps are referred to as the connection phase and are not random in legacy systems such as Long-Term Evolution. When the offered load is below 1, the system is under-utilized. Likewise, when the offered load is above 1, the system is over-utilized and may cause RA congestion and high access latency due to excessive collisions.

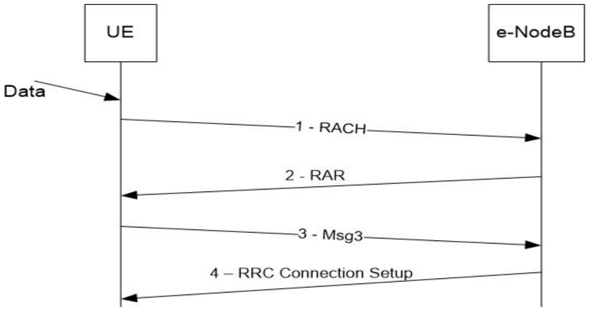


Fig. 1. RA 4-step procedure in LTE [4].

Massive MTC and event-driven communication (e.g. sensors during an earthquake) may cause most of these devices to try to establish connection at the same time, resulting in massive bursts of traffic [4]. Traffic bursts cause more devices to request RA, which leads to RA congestion and longer access latency. While LTE was designed with enough RA opportunities to achieve a 99% RA success probability at the preamble phase, a higher collision probability is expected for the massive MTC (mMTC) scenario [5].

In general, there are two approaches to address the RA congestion problem where both aim to bring the system to operate at its peak performance of 37% by reducing the offered load back to 1. The first approach is to allocate more RA such that the resources can cope with the high traffic load. The second approach is to regulate the number of incoming users so that the regulated load matches the available resources.

The first approach attempts to increase the RA success probability by allocating more resources for RA to ideally reach $N = k$. LTE has multiple Physical RA Channel (PRACH) configurations [6] and it is proposed in [7] to use it dynamically depending on the current system load. Other techniques in this category include the use of virtual preambles [8], [9] or multiple RAR messages [10], [11]. These methods virtually multiply the number of preambles by allocating more resources than received preambles to serve users that may have collided at the preamble stage. However, these methods also introduce ambiguity overhead [8], which can lead to a significant waste of resources.

The second approach attempts to bring the system to its peak performance operating point by regulating the incoming traffic load to ideally reach $k = N$. Back-off mechanisms [7] make users wait a random amount of time to attempt RA after a collision; following a traffic burst, this spreads traffic over time and reduces instantaneous traffic. Apart from back-off mechanisms, access barring can limit the access and reduces the load. Class barring separates users between high priority and each class has different RA parameters. A typical application of this scheme would be to separate Human to Human (H2H), mission critical and mMTC users in different classes. LTE implements Access Class Barring (ACB), where low priority users have a longer backoff timer than high-priority users as well as a barring factor p; each user draws a random number $0 \leq q \leq 1$ and if $q > p$ they may attempt RA. Enhanced Access Barring (EAB) further enhances this scheme by completely barring low-priority users from attempting RA in case of high traffic. These methods essentially sacrifice some throughput to ensure a better quality of service to high-priority users. First introduced in [12], adaptive traffic load (ATL) S-ALOHA uses traffic prediction to optimize the throughput by dynamically adapting the barring factor p to the current load. By setting p such that $(1 - p) \cdot k = N$, it can achieve the maximal theoretical throughput.

All these methods are optimal for a given traffic load and need to adapt over time to the varying number of users to continue to perform well, either by passively smoothing traffic peaks (e.g. back-off schemes), which is slow and requires users to fail until the system is at peak performance, or by actively changing system parameters (e.g ATL S-ALOHA), which relies on accurate traffic prediction and fast network adaptation to broadcast updated parameters. Indeed, in a typical PRACH configuration [7], there are 16 RA slots between each parameter update. This can severely damage the RA performance and access delay.

Our research effort in this paper falls under the scope of the first approach, more specifically the multiple RARs (M-RAR) scheme, and expands it by modifying the preamble phase to use it not only to admit users but for load estimation. Authors in [10] provide an analysis of the success rate of this scheme for fixed numbers $M \geq 1$ of RARs, which shows improvement in RA success probability. In this paper, we propose to dynamically and instantaneously adapt the number M of multiple RARs following a traffic load estimation at the preamble phase. However, our analysis shows that the existing preamble phase can be used for load estimation only for a limited range of loads (see Fig. 6), and the load estimation fails beyond this range of 235 users for $N = 54$ as shown later in Section III. Supporting a wide range of traffic load is essential for massive MTC. Therefore, instead of aiming to maximize the throughput in preamble phase as in the state-of-the-art, we challenge this paradigm by designing a preamble phase that aims to perform load estimation even when the system would normally be congested and use this information to later serve users. We call our RA method Preamble Barring (PB), which uses a probabilistic resource separation at the preamble stage to achieve accurate load estimation in a wide range of load conditions. We then use this estimation to achieve optimal throughput at the connection phase with M-RAR. In this paper, we focus on the robustness of the scheme to sudden unpredictable traffic bursts that happen over the course of one RA slot of up to 1000 users [10], rather than a steadily increasing long traffic burst [7], although our scheme is also perfectly suited for such events. The throughput analysis shows that this scheme can serve 1000 users with only 54 preambles while having a near-optimal throughput performance, which indicates that it is a promising solution to avoid congestion and achieve fast uplink access for mMTC devices. In Section II, we provide an analysis of M-RAR throughput performance and determine the best choice of parameter M for a given load, the difficulty of estimating and predicting load for traditional RA schemes is also studied. In Section III, we introduce a novel RA scheme, Preamble Barring, and show how it solves the load estimation problem under high load and achieves optimal performance both for high and low load conditions without relying on traffic prediction. The performance advantages of our proposed scheme are presented and discussed. We finally draw important conclusions and discuss future work in Section IV.

II. Multiple RAR

In this section we analyze M-RAR performance and resource overhead for different traffic loads and derive an optimal M for a given load; we also show limits of traffic estimation and its impact on RA performance for high load conditions.

A. Concept of multiple RAR (M-RAR)

The key idea of M-RAR is the use of multiple RAR messages for each received preamble at step 2 to avoid potential collisions. If there are two or more users choosing the same preamble at step 1, with multiple RAR messages, the users may choose to respond to a different RAR message. While the collision probability at the preamble phase remains unchanged, M-RAR reduces the collision probability at the connection phase. The drawback of this method is that some RAR messages may be sent unnecessarily resulting in resource wastage for step 2 and 3. This can be avoided by adjusting M based on the number of users choosing the same preamble. An appropriate choice of M depends on 1) the knowledge of the current load, 2) the targeted trade-off between collision probability and resource wastage and 3) the amount of RAR resources available. In this paper, we assume that 1) the bursty nature of MTC traffic makes load prediction very difficult but it can be estimated by looking at the number of chosen preambles after the preamble phase, 2) the metric we want to optimize is the overall resource efficiency (also referred to as channel throughput in S-ALOHA), and 3) we do not have restrictions on the number of RARs; indeed, the maximum throughput being 37%, RA resources will still need to be used at a later time, therefore we consider that efficiently connecting users on the first attempt saves time and energy for both the users and the BS.

B. Throughput analysis

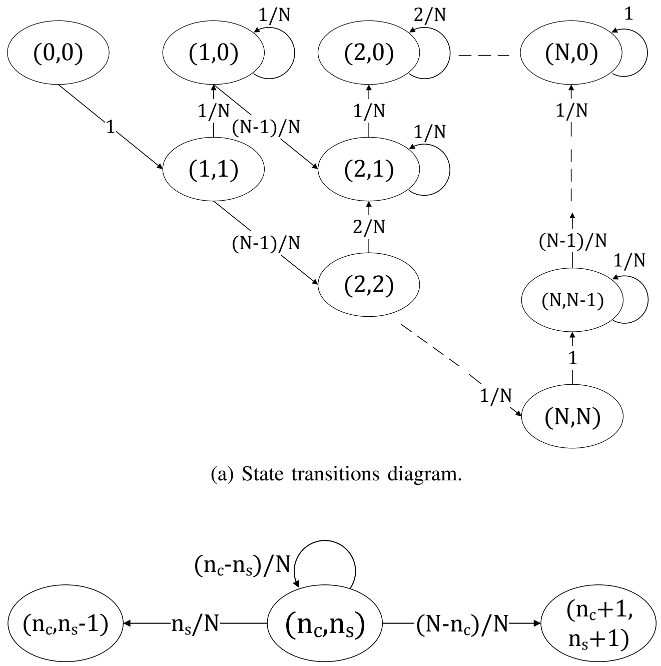
To be able to accurately estimate the performance of the M-RAR scheme, we need to estimate 1) the number of users that attempted RA at the preamble phase, 2) their distribution among the chosen preambles and 3) their success probability at the connection phase with M RARs.

Incoming traffic can be separated in two components: previous users who could not be served in previous attempts and new users. The change of load largely depend on new arrivals and the prediction can be very challenging with a sporadic and bursty traffic, as can be expected from a massive MTC scenario with event-driven communication. In this paper however, contrary to traditional RA schemes, we do not rely on the preamble phase where traffic prediction is required but rather on the connection phase where load estimation can be inferred from the preamble phase. Hence, we will only focus on load estimation following the preamble phase, although additional knowledge of the incoming load may still be used, in (2) for instance.

In this paper, we assume that every selected preamble is perfectly detected by the BS. Although a closed form formula for the number of selected preambles for a given number of users is provided in [13], we also need the exact probability of the number of successful preambles. In this paper, we propose a state transition analysis to compute both of these probabilities at the same time.

For N available preambles, we consider the probability of having n_c chosen preambles and n_s successful preambles. There are $(N+1)(N+2)/2$ possible states (n_c, n_s), with $0 \leq n_s \leq n_c \leq N$, as shown in Fig. 2a. The state transition probabilities for any given state (n_c, n_s) are described in Fig. 2b. Indeed, the arrival of a new user results in a probability $(N - n_c)/N$ to choose a new preamble which results in state $(n_c + 1, n_s + 1)$, a probability $(n_c - n_s)/N$ to choose a previously collided preamble which results in state (n_c, n_s) and a probability n_s/N to choose a previously successful preamble and collide which results in state $(n_c, n_s - 1)$. We can then derive the transition matrix P and by initializing at $(0, 0)$ with $v_0 = (1, 0, ..., 0)^T$, we can then compute the probability of being in a state (n_c, n_s) for k users with $v_k = P^k \cdot v_0$.

This approach is similar to Markov chain analysis, however we do not look for steady state as there is only one absorbing state $(N, 0)$ as shown in Fig. 2a. Here we are looking at transient states that give us state probabilities after a certain number of steps corresponding to the number of users k.



(a) State transitions diagram.

(b) Transition probabilities of a particular state (n_c, n_s).

Fig. 2. System model for the analysis of preamble outcomes.

We now need to determine the distribution of the k users among the n selected preambles. While we could use the same approach to determine all the possible states and their corresponding probabilities but this would be very complex. We choose here to use an independent Poisson distribution approximation, which is particularly accurate since k and n are known so this does not affect the overall throughput predictions as shown in Fig. 3. Let l be the number of users who selected a given preamble. We model l as a conditional Poisson distribution ($l \geq 1$) of parameter $\lambda = k/n$: $P(l = q | l \geq 1) = \frac{\lambda^q}{q!} e^{-\lambda} \frac{1}{1 - e^{-\lambda}} = \frac{\lambda^q}{q!} \frac{1}{e^{\lambda} - 1}$.

Finally, we need to compute the probability of success of l users for M RARs. There is no closed-form expression for this probability and in the literature, we normally consider a high number of users and preambles and the approximation $P_{Success} = e^{-l/M}$ is usually used [4]. This approximation does not hold for lower values of l and M, as is the case in this paper. Using our proposed method described in Fig. 2a and Fig. 2b, we can accurately derive this probability. We verify the accuracy of our analysis for $k = 250$ users and varying values of M using Monte Carlo simulations (with $T = 10^4$ steps), as displayed in Fig. 3.

C. Optimal choice of M and load estimation accuracy

We derived the exact performance of the M-RAR scheme. To determine the optimal choice of M, we now seek to optimize the channel throughput for a given number of users k, i.e. the ratio of successfully used RARs with respect to

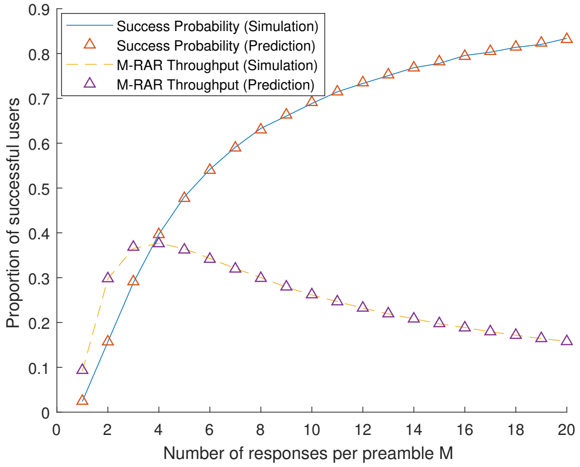


Fig. 3. M-RAR performance for $k = 250$ users.

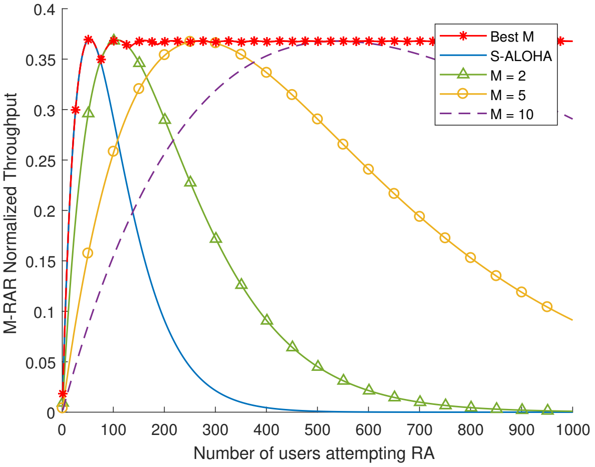


Fig. 4. M-RAR performance for fixed M and varying load.

the amount of allocated RARs $M \cdot N$. We can achieve this by exhaustive search of the previously computed success probabilities, which can later be stored in a table. We draw the throughput curves for different fixed values of M for traffic loads varying from 1 to 1000 users in Fig. 4. We can see that by always selecting the best M, we are practically always achieving optimal throughput. It must be noted that, similarly to ACB, we choose $M = 1$ for low loads and then achieve optimal throughput for each M at $k = M \cdot N$; where ACB reduces the number of users to $(1 - p) \cdot k = N$, our scheme increases the number of resources to $M \cdot N \simeq k$. Given that M is an integer, this value is not always reached but Fig. 4 shows that, in practice, we have over 35% throughput from $k = 35$ users and it converges to the ideal 37% value. However, the advantage of this scheme over ACB is that it does not require load prediction or fast parameter update, so the system can work at its optimum in any situation. Indeed, the aim of this scheme is to be able to provide users with an optimal connection phase throughput on the first try with a scalable number of RARs. On the contrary, legacy schemes need to adjust a barring parameter or have users back off on a trial-and-error basis but with a bounded number of RARs.

In practice, we estimate the load using the most likely number of users k for n received preambles as described previously. We then choose the optimal M for this estimated load. Performance results are shown in Fig. 7. As we can observe, the performance is very close to the ideal case up to 250 users and then suddenly drops as the preamble set is saturated (all preambles are used) and the BS is not able to infer whether there are 250 or 1000 users. This load estimation error could prove particularly harmful in case of a sudden unpredictable traffic burst, which is expected in event-driven communication.

Thus, we change the paradigm of trying to predict the load to optimize the preamble phase throughput, and instead use the preamble phase to optimize the throughput at the connection phase (which is what the ultimate goal is), we can achieve near optimal S-ALOHA throughput all of the time, even when all or most of the users actually collided at the preamble phase. However, this method shows its limits when the number of users is so high that the system is unable to give an estimation of this number, which defeats its purpose.

III. Exploiting Preamble Barring to optimize Multiple RAR

We saw in the previous section that Multiple RAR can theoretically always achieve near-optimal throughput but it relies on a good load estimation to achieve its full potential. Hence, a preamble phase that could help the BS more accurately estimate the load for a wider range of incoming number of users would solve this problem. In this section, we introduce such a method.

A. *Preamble Barring*

In regular S-ALOHA, all users randomly choose any of the N signatures with uniform distribution; this optimizes the throughput $S = k/N \cdot e^{-k/N}$ (as shown in Fig. 5) with a maximum of $1/e \simeq 0.37$ when $k = N$.

In the PB scheme, the preambles are divided in different sets and users are given a probability p_i to access the i^{th} set, which consists of n_i preambles. The choice of preamble is therefore not uniform, it consists in a probabilistic separation of resources, where some preambles are more likely to be selected than others. In case of a sudden traffic burst, a set that targets to serve low load (dense set) will be saturated but not a set that targets to serve high load (sparse set). Thus, sparse sets will reach saturation for higher loads and can therefore provide an accurate load estimation where legacy S-ALOHA would be saturated. We can then extrapolate the load on all the other sets with good precision.

Following the description of the PB procedure, we can determine that each preamble set receives an offered load $k \cdot p_i/n_i$, leading to a normalized throughput of $k \cdot p_i/n_i \cdot e^{-k \cdot p_i/n_i}$. As

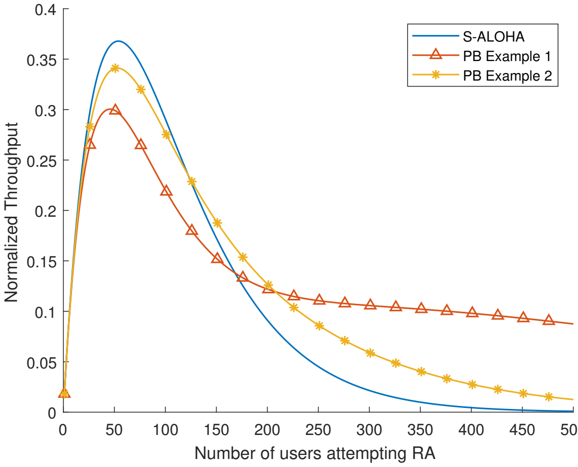


Fig. 5. Preamble phase throughput for S-ALOHA and PB.

a result, we can derive the overall normalized throughput as a weighted sum of the throughput of each group as follows

$$S = \frac{k}{N} \cdot \sum_{i \geq 1} p_i \cdot e^{-k \cdot p_i / n_i}. \tag{1}$$

Fig. 5 shows examples for two sets with $p_1 = 0.95$, $n_1 = 39$, $p_2 = 0.05$ and $n_2 = 15$ (PB Example 1) and $p_1 = 0.2$, $n_1 = 20$ and $p_2 = 0.8$, $n_2 = 34$ (PB Example 2). These configurations have near-optimal throughput at low loads since no user is barred from attempting RA. We also achieve a flatter curve, allowing a near-constant throughput at very high loads. A drawback is that the flatter the curve, the lower its maximum throughput, as we can see with examples 1 and 2.

However, this sub-optimal throughput only concerns the preamble phase (so the preambles that are only chosen by a unique user) and as we showed in Section II, achieving a high throughput at the preamble phase is not as important as being able to estimate the load and this can only be done by having non saturated sets, which the PB scheme allows.

B. Load estimation precision

The load estimation accuracy depends on the offered load (i.e. k/N) on a given set and the number of preambles it contains. The more preambles a set contains, the better it can accurately estimate a load at low offered loads. However, preamble sets become saturated when the load is too high, the BS then has to make an arbitrary decision as to what load to estimate when all the preambles are used. In this paper, we define the saturation point as the smallest load for which the probability of having all preambles used is greater than 0.5. Using the state probabilities derived in Section II, we can show the a 54 preambles set reaches saturation for 235 users; this means that if there are 235 users or more attempting RA with 54 available preambles, there is more than 50% probability that all preambles will be chosen. Once the saturation load is reached, the load estimation error will increase linearly. Preamble barring allows different offered loads on the various preamble sets. While the lower number of available preambles makes low load estimations less accurate, this methods allows arbitrarily high saturation points according to the PB parameters. For example, using the aforementioned Preamble Barring 1 parameters, the sparse set of 15 preambles has a saturation point of 47 users, which means a maximum accurate estimate of $47/p_1 = 940$ users overall.

On top of being able to push the saturation point further, we can also infer information from the other sets' observation. Indeed, for J sets with parameters $(n_i, p_i)_{1 \leq i \leq J}$, we can derive the probability of observing $(n_{c,i})_{1 \leq i \leq J}$ chosen preambles for any given k. Thus the probability of having $(k_i)_{1 \leq i \leq J}$ users for each set is

$$\begin{aligned} &P[(k_1, .., k_J) | (n_i, p_i, n_{c,i})_{1 \leq i \leq J}] \\ &= P(X = k) \cdot P[(k_1, .., k_J) | (n_i, p_i)_{1 \leq i \leq J}] \cdot \prod_{1 \leq i \leq J} P(k_i | n_{c,i}), \end{aligned} \tag{2}$$

where $k = \sum_{1 \leq i \leq J} k_i$. As mentioned in the Section II.B, knowledge about the expected load can always be added to further enhance the accuracy for the estimation of $(k_i)_{1 \leq i \leq J}$, hence the term $P(X = k)$ in (2). However, in this paper we do not consider any knowledge about k so $P(X = k)$ is not taken into account. Thus, to derive the optimal Maximum Likelihood (ML) $(k_1, k_2)^*$ given $(n_{c,1}, n_{c,2})$ for examples 1 and 2 with two sets, we need to solve:

$$(k_1, k_2)^* = \max_{(k_1, k_2)} P(k_1, k_2) \cdot P(k_1 | n_{c,1}) \cdot P(k_2 | n_{c,2}), \tag{3}$$

where $P(k_1, k_2) = f_{Binomial}(k_1, k = k_1 + k_2, n_1) = \binom{k_1 + k_2}{k_1} \cdot p_1^{k_1} \cdot (1 - p_1)^{k_2}$ and $P(k_i | n_{c,i})$ are determined using the state transitions in Section II. We find the ML solution using an exhaustive search for $0 \leq k \leq k_{max}$ for a complexity of $\mathcal{O}(k_{max}^2)$. The value k_{max} is determined as the saturation load. When both sets are saturated, we calculate the saturation point of both sets and choose the highest $k_{max} = \max(k_{1,max}/p1, k_{2,max}/p2)$, $k_{i,max}$ being the regular S-ALOHA saturation point for n_i available preambles. Using (3) for example 1, Fig.6 compares the average relative errors (defined as $E[|\frac{k - \hat{k}}{k}|]$) for load estimations for k_1, k_2, k for PB and $k_{S-ALOHA}$ for S-ALOHA for traffic loads between 1 and 1000 users. As we can see, S-ALOHA maintains an average relative load estimation error below 10% until it becomes saturated and the error steadily converges to 100% since its saturation point is at 235 users. PB on the other hand has a saturation point of 940 users and we observe a much more contained relative error for every load between 1 and 1000 users.

C. Results and Discussion

Similarly to what was done in the previous section, having estimated the number of users in each set using (3) we can select the optimal M_i corresponding to each set. Indeed, sparse sets will require lower M_i than dense sets. Thus, if the combination of sets is able to accurately estimate the

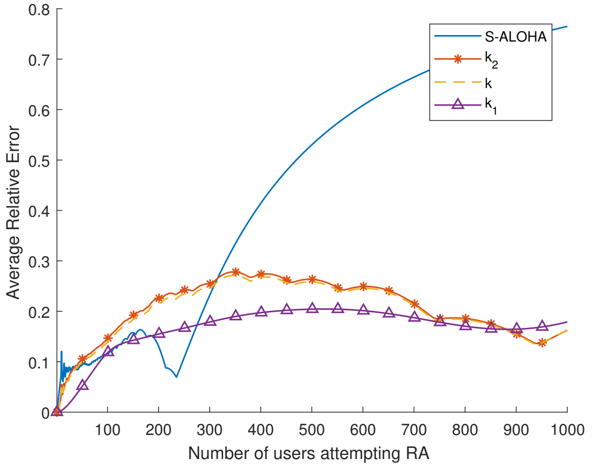


Fig. 6. Average relative error when estimating k_1, k_2 and k for PB using ML and S-ALOHA.

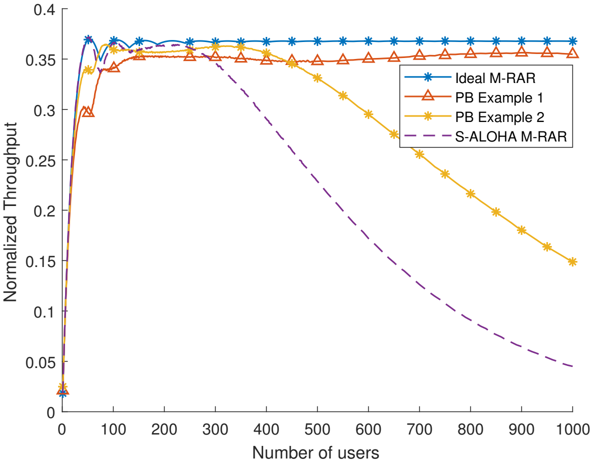


Fig. 7. Connection phase throughput for ideal and practical M-RAR S-ALOHA and M-RAR PB.

load, each set can theoretically achieve optimal throughput, hence achieving optimal throughput for all N preambles. Fig. 7 shows the performance of the PB scheme for the two examples mentioned earlier in section II (see Fig. 5). As we can see, both manage to perform remarkably well at very high loads while still offering very good throughput at low loads as well. We can clearly see the peak performance points for example 1 and example 2 at their saturation point ($k_{max} = 940$ and $k_{max} = 340$ users respectively) and how example 2 starts to lose performance as it goes past its saturation point. Comparing S-ALOHA with examples 1 and 2 from the preamble phase (see Fig. 5) and the connection phase (see Fig. 7) shows that a higher maximum throughput at the preamble phase results in a higher throughput at low loads ($k < N$) at the connection phase and that a flatter curve at the preamble phase results in a higher saturation point at the connection phase. Indeed, at low loads, both sparse sets and dense sets will use $M_i = 1$. The M-RAR PB is then equivalent to a standard PB, which as we saw in Section II cannot achieve 37% like S-ALOHA. At higher loads however, a PB solution will not be saturated and will still be able to provide an accurate load estimation. By tuning the PB parameters to choose a saturation point, the proposed scheme can effectively adapt to any traffic type. Moreover, it is robust to sudden traffic changes and does not require constant or fast update. Indeed, the saturation point will only depend on the maximum number of simultaneous users that the BS has experienced in a given burst and should not change quickly in time.

To compare this scheme with adaptive schemes such as dynamic barring schemes, we need to consider a system with varying load where the BS tries to predict the incoming load. Our scheme has access to as much information as these schemes, although the overall load estimation can be slightly less accurate for low loads (see Fig. 6). However, by instantaneously allocating as many RA resources as necessary, PB can prevent any congestion and quickly and efficiently serve users without having to use more PRACH resources. Short intense traffic bursts, such as those studied in this paper, can be served in a number of RA slots that grows logarithmically with the number of users (10 steps for 1000 users) because 35-37% of users will be served at each step. On the other hand, longer less intense bursts such as those described in [4], [7] will be instantly resolved without needing to spread users over time, provided that there are sufficient uplink data resources.

IV. Conclusions and future work

In this paper, we challenged the common practice that relies on predicting the load to optimize the RA preamble phase and instead use this phase to provide information regarding the load before sending an appropriate number of RARs at step 2. In section II, we introduced a markovian state transition model that allowed us to derive the exact success probability in a S-ALOHA system. With this, we could estimate accurately the system throughput and determined the optimal Multiple RAR parameter M. In Section III, we enhanced the Multiple RAR technique by proposing a novel probabilistic Random Access scheme, namely preamble barring, which allows instantaneous traffic load estimation under a wide range of traffic conditions.

We showed that the proposed scheme can instantaneously use near optimal resources for an unexpected burst of 1000 users with only 54 preambles, while also having a near optimal throughput at low loads. In state-of-the-art solutions, this extreme scenario would result in an immediate traffic congestion that would take time resolve and result in lesser throughput and greatly increased uplink access delay. Since the BS already sends individualized RARs for each received preamble and this scheme does not require any parameter update, the implementation costs are minimal and we believe that this is a promising solution to avoid congestion and access delay for fast random access in massive MTC.

Future work should investigate the use of more than two preamble sets and the optimization of the PB parameters in a dynamic environment with a given traffic pattern. In addition, we assumed in this paper that all users were using the same PB parameters, investigating different access probabilities for different classes of users in future work would help tackle the problem of the coexistence of H2H and MTC traffics.

Acknowledgment

We would like to acknowledge the support of the University of Surrey 5GIC (http://www.surrey.ac.uk/5gic) members for this work and NEC Telecom Modus Ltd for partly funding this research through an Industrial Cooperative Awards in Science & Technology (iCASE) studentship. This work was also partly funded by grant N62909-17-1-2114 from the US Office of Naval Research Global.

References

[1] A. Laya, L. Alonso, and J. Alonso-Zarate, "Is the Random Access Channel of LTE and LTE-A Suitable for M2M Communications? A Survey of Alternatives," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 4–16, 2014.

[2] J. Kim, J. Lee, J. Kim, and J. Yun, "M2M service platforms: Survey, issues, and enabling technologies," *IEEE Communications Surveys and Tutorials*, vol. 16, no. 1, pp. 61–76, 2014.

[3] A. Ijaz, L. Zhang, M. Grau, A. Mohamed, S. Vural, A. U. Quddus, M. A. Imran, C. H. Foh, and R. Tafazolli, "Enabling Massive IoT in 5G and Beyond Systems: PHY Radio Frame Design Considerations," *IEEE Access*, vol. 4, pp. 3322–3339, 2016.

[4] 3GPP, "Ran improvements for machine-type communications," TR 37.868, 3rd Generation Partnership Project (3GPP), Sep. 2011. V11.0.0.

[5] A. Lo, Y. W. Law, M. Jacobsson, and M. Kucharzak, "Enhanced LTE-advanced random-access mechanism for massive machine-to-machine (M2M) communications," in *27th World Wireless Research Forum (WWRF) Meeting*, pp. 1–5, WWRF27-WG4-08,, 2011.

[6] 3GPP, "Evolved Universal Terrestrial Radio Access (E-UTRA); Physical channels and modulation," TS 36.211, 3rd Generation Partnership Project (3GPP), Jun. 2018. V14.2.0.

[7] ZTE, R2-104662: MTC simulation results with specific solutions, in 3GPP TSG RANWG2 Meeting 71, Aug. 2010.

[8] N. K. Pratas, H. Thomsen, Č. Stefanović, and P. Popovski, "Code-expanded random access for machine-type communications," in *2012 IEEE Globecom Workshops*, pp. 1681–1686, Dec 2012.

[9] S. Vural, N. Wang, G. Foster, and R. Tafazolli, "Success probability of multiple-preamble based single-attempt random access to mobile networks," *IEEE Communications Letters*, pp. 1–5, 2017.

[10] J. Kim, D. Munir, S. Hasan, and M. Chung, "Enhancement of LTE RACH through extended random access process," *Electronics Letters*, vol. 50, no. 19, pp. 1399–1400, 2014.

[11] J. S. Kim, S. Lee, and M. Y. Chung, "Efficient random-access scheme for massive connectivity in 3gpp low-cost machine-type communications," *IEEE Transactions on Vehicular Technology*, vol. 66, pp. 6280–6290, July 2017.

[12] M. E. Rivero-Angeles, D. Lara-Rodriguez, and F. A. Cruz-Perez, A new EDGE medium access control mechanism using adaptive traffic load slotted ALOHA, in IEEE 54th Vehicular Technology Conference. VTC Fall 2001. Proceedings (Cat. No.01CH37211), 2001, vol. 3, pp. 13581362.

[13] C. M. Chou, C. Y. Huang, and C.-Y. Chiu, "Loading prediction and barring controls for machine type communication," in *Communications (ICC), 2013 IEEE International Conference on*, pp. 5168–5172, IEEE, 2013.