Blocking probability for GSM service requests during overload

Terje Jensen and Olav Østerbø

Telenor R&D, B5d, NO-1331 Fornebu, Norway
{terje.jensen1, olav-norvald.osterbo}@telenor.com

Abstract: During overload (high load) in GSM systems the product of intensity of requests and service times is higher than the capacity of the traffic channels. Prior to potentially being allocated a traffic channel, the request has to successfully get through a random access channel and a common signalling channel. Both of these latter channel types could potentially become bottlenecks during high load configurations. This paper provides a simple approximation for estimating blocking of requests prior to reaching the algorithm for allocating traffic channels.

Keywords: Mobile system, overload, blocking probability

1. INTRODUCTION

The GSM system is intriguingly designed consisting of several channel types and protocols. In particular this is relevant for the air/radio interface between the base station and the mobile station. These channel types are mapped onto different physical resources such as frequencies and time slots on the air interface. A network operator is able to configure the use of these physical resources within certain limits. However, more resource capacity to some channel types implies less to others. From the outset it is not obvious how these resources should be configured, as the better overall performance would typically depend on the traffic mixture.

Prior to allocating a traffic channel for a set-up request, the request must get through the random access channel (RACH) and the stand-alone dedicated control channel (SDCCH). After successfully utilising these two channels the mobile user’s identity is confirmed and further resources allocated if requested. This means that when a mobile user wants to make a phone call, firstly the RACH must be used and then the SDCCH must be used. Thereafter a traffic channel is allocated.

During overload, it is likely that no traffic channels are available as all are occupied with on-going calls. Still, further requests are made by the mobile users, seizing RACH and SDCCH capacity. As these requests are likely to be rejected, the mobile users would in several cases try again. These re-tries add to the original request rate and hence could result in a quite much higher load on the RACH and SDCCH. Hence, it would eventually be more difficult to successfully get through RACH and SDCCH as the effective overall request rate increases.

To illustrate this phenomenon, a measurement trace is given in Figure 1. This illustration...
shows blocking on traffic channels (TCH) and SDCCH during an interval of a number of hours. The original cause of this situation was a chain collision close to the main airport in Norway. As seen, blocking rate of TCH was up to about 80% as average during a one-hour measurement. There were 13 traffic channels installed at that GSM site. Blocking and load on SDCCH are also indicated. An additional concern is that a user also needs access to an SDCCH in order to utilise the Short Message Service (SMS). In fact a high blocking ratio on SDCCH prevents access to all GSM services.

![Figure 1 Measured traffic and congestion on a certain GSM base station during high load situation](image)

This phenomenon of rejected requests and re-tries during high load situations is well known for different systems, including PSTN/ISDN and web servers. Depending on the “persistence” of the mobile users (i.e. how eager they are to re-try attempts), the fraction of successfully treated requests could become very low. For the configuration of RACH, SDCCH and TCH, different server models apply. For example, the RACH would have some similarities with slotted ALOHA systems. Results for such, and similar, configurations are available in a range of publications, a few being [1] and [2]. More detailed studies for wireless systems also considering propagation conditions have also been conducted, e.g. [3] and [4]. Closer examinations of performance of RACH are also available, such as [5], [6] and [7].

Compared to those publications, in this paper we are looking at both effects from SDCCH and RACH blocking. Hence, a main question addressed is: What is the probability of successfully getting through RACH and SDCCH in particular during high load intervals?

An objective of this paper is to elaborate a simple approximate method for estimating this blocking probability.

A brief description of the system and the model is given in the following chapter. Then the analysis approach is presented in Chapter 3 supplemented with some results and observations in Chapter 4.
2. SYSTEM AND MODEL DESCRIPTION

A principle flow through different channel types used during a mobile user’s request is shown in Figure 2. Note that not all cases are shown, such as request collisions, retries, time-outs, etc. Moreover, a number of other channel types are not shown either, such as the one applied when acknowledging the random access request.

![Diagram](https://via.placeholder.com/150)

Figure 2 Principle flow of requests through different channel types as looked at in this model

Considering a given overload situation, a working assumption here is that all traffic channels are occupied all the time. Naturally this is an approximation, although it is assumed that the overall load is so high that each request attempt has a very small probability of finding a traffic channel available. Referring to the situation shown in Figure 1 measured blocking on the traffic channels is about 70 – 80 % in average during two subsequent one-hour intervals. Hence, during several shorter intervals, all traffic channels were occupied. An effect of this working assumption is that the traffic channels do not need to be explicitly included in the model.

A more detailed illustration of the model is given in Figure 3. A basic motivation for this model is to start with applying a number of simplifying assumptions allowing for fast assessment of performance estimates. In general the dependencies between inter-arrivals for resource groups are neglected for this initial model. The traffic situations for the two resource groups are:

- Random Access Channel (RACH): where mobile terminals are generating requests in defined slots – competing for these slots according to specified procedures. A maximum of \( n \) retries are applied when collisions occur (a modified slotted ALOHA-model). Holding time for this resource is a single slot when transfer is successful.

- Stand-alone Dedicated Control Channel (SDCCH): allowing mobile terminals to signal its service requests. The mobile’s identity is also conveyed during this phase. Holding time for this resource has to be specified.
3. ANALYSIS ALGORITHM

3.1 Overall flow of calculations

Neglecting the inter-arrival dependencies, the model allows for an iterative/sequential analysis, outlined as follows (illustrated in Figure 4):
1. Given the “original” arrival rates for the requests – that is service initiation rates for “fresh calls” as generated by the users.
2. Estimate “effective” resource load on random access capacity considering the load and number of retries allowed.
3. Calculate rate of “successful” requests for random access (continuing on to the SDCCH) and rate of “unsuccessful” requests.
4. Calculate rate of “successful” requests for SDCCH and rate of “unsuccessful” requests.
5. Calculate rate of retried call attempts (as controlled by the users)
6. Check for convergence in main variables (e.g. arrival rates for resource groups) – end iteration when converged, continue from step 2 when not converged.
3.2 Set of analysis formulae

The “original” arrival request rate is referred to by $\lambda$ (ref. step 1 in calculation procedure).

3.2.1 Random access capacity, RACH

The RACH “effective” request rate can be calculated as (step 2 in the calculation procedure):

$$\lambda_{\text{eff}} = \lambda \sum_{i=0}^{n-1} (i+1) \cdot b^i \cdot (1-b) + (n+1) \cdot \lambda \cdot b^n = \lambda \sum_{i=0}^{n} b^i \cdot \frac{1-b^{n+1}}{1-b} ,$$  \hspace{1cm} (1)

where $b$ indicates the probability that a request is colliding with a number of other requests (and hence not successfully “served” by the random access capacity). This might be estimated by assuming a slotted ALOHA model for this capacity.

Note that $n$ refers to number of retries; hence in maximum $(n+1)$ access attempts can be seen.

For a population of $M$ mobile stations, each having a given probability of initiating a “fresh” attempt in a random access time slot equal to $r$, the probability of a successful access, in case no retries were allowed, is found by: $M \cdot r \cdot (1-r)^{M-1}$. Given that we are looking at a particular mobile station, the probability becomes equal to $(1-r)^{M-1}$. That is, an attempt is successful when none of the others are transmitting in the same slot. However, the retries add to the effective rate.

The resulting probability that no other transmissions (“fresh” and retries) arrive in the same time slot as a given attempt is approximated by $e^{-\lambda_{\text{eff}}}$. This is introducing a Poissonian arrival model for requests. To some extent, this could be argued by a high number of mobile stations each having a certain request generating probability for that time slot. Hence, as a first approximation: $b = 1 - e^{-\lambda_{\text{eff}}}$, \hspace{1cm} (2)

As shown, from these expressions, iteration can be applied solving:

$$\lambda_{\text{eff}} = M \cdot r \cdot e^{-\lambda_{\text{eff}}} \left\{ 1 - \left( 1 - e^{-\lambda_{\text{eff}}} \right)^{n+1} \right\} .$$ \hspace{1cm} (3)

Figure 5 shows relations between “fresh requests” and the effective rate of random access attempts for different values of $n$. 

---

Figure 4 Overall flow of traffic in the modelled system
Figure 5 “Effective” request rate to RACH as function of the product \( M.r \) for some values of \( n \). Left part shows more details for lower traffic areas.

The probability that a given mobile station’s attempt gets through is then found as \( 1 - b^{n+1} \), where \( b = 1 - e^{-\lambda eff} \). This function is depicted in Figure 6, note that the effective load on the access channel is the input variable. As a worst case, the effective load on the access channel is equal to \((n+1)\) times the “fresh” attempts as seen from that resource/access channel (retries due to user impatience, user re-dials, etc. are not considered in this variable).

Values for success probability as function of rate of “fresh requests” are given in Figure 7 (ref. step 3 in procedure). As more retries are allowed the success probability has the steepest decrease as should be expected as the retries add to the overall load on the random access channel.

Figure 6 Probability of successful attempt as function of effective load on access channel (i.e. including retries) for different number of maximal retry options.
0.2
0.4
0.6
0.8
1

0.2
0.4
0.6
0.8
1

Figure 7 Probability of success as function of “fresh requests” – product $M \cdot r$ for different values of $n = 0.7$. $n = 0$ is the curve with the slowest decrease of success probability as function of increasing “fresh request” rate

3.2.2 SDCCH capacity

The performance of the SDCCHs under consideration will depend on the actual arrival process and the corresponding holding times. We assume that the losses due to lack of random access channels is negligible or that the actual thinning of the ordinary input stream due to random access loss is Poissonian. By this assumption, for step 4 in the procedure, we could analyse the SDCCHs by an M/G/s/s loss-system where $s$ is the number of channels available. Consequently the loss probability due to lack of SDCCHs may be estimated by applying Erlang’s formula. Further the “carried” rate offered to the ordinary speech channels may be estimated as:

$$\lambda_{speech} = \lambda(1-b)[1-E_1, (\lambda(1-b) t_{SDCCH})]$$

(4)

where $t_{SDCCH}$ is the mean holding time of a dedicated control channel.

3.2.3 Retries

It is possible to extend the results above by considering the effect of retries (ref. step 5 in the calculation procedure). Assuming a fixed retry probability, $p_r$, (independent of number of retries a user has been through), the resulting input rate has to be increased to $\frac{\lambda}{1-p_r}$, however this can be adapted by just scaling the input rate due to retries. During heavy overload we believe that the increase of the input rate due to retries will be significant.
4. EXAMPLES AND OBSERVATIONS

The overall probability of success, given that this is modelled as a series of factors, is estimated by the product of the corresponding “blocking” probabilities. Again assuming independence, the average number of attempts needed is given by the inverse of the overall success probability. \( n = 4 \), that is, in total 5 attempts are assumed for all calculations in the following.

If attempts are made every 5 second per mobile station, this can then be used for estimating average time until a call gets through. Given 115 random access time slots per second, the effective load/rate for this resource group has to be scaled accordingly. Furthermore, assume 5 second “think time” between re-tries for a user facing an unsuccessful call attempt.

1000 users/mobile stations will then generate a scaled load of (i.e. neglecting the relatively few calls that are connected): \( M \cdot r = (1000/5\text{sec}) \cdot (1\text{sec}/115) = 1.74 \).

Similarly, a case of 300 users with the same behaviour results in \( M \cdot r = 0.52 \). From Figure 5 we find an effective rate (including “automatic retries” due to collisions) about 1.8. This gives probability of a successful request about 0.59. Hence, the arrival rate that the SDCCH resource group faces, in this case, is \( 0.59 \cdot (300/5\text{s}) = 35.4 \) per second. Assuming an SDDCH holding time of 5 seconds, gives an equivalent load of 177. This can be inserted into Erlang’s loss formula giving blocking probabilities for the SDCCH resource group. The overall success probability is found by adding the success of the random access and the success (= 1 – blocking probability) for the SDCCH. In this calculation example, this success probability would be rather small.

In a similar manner, a number of graphs of the overall success probability are presented in the following. For all these calculations the number of mobile users, \( M \), is assumed to be 300. When 4 SDCCH channels are present, the random access channel is assumed to have 115 slots per second. When higher numbers of SDCCH channels are used (8, 12, 16, 20 and 24) 217 slots per second are given for the random access channel.

Overall success probabilities given in the following are:

- Figure 8 - left: function of time between repeated call attempts for mobile users (two illustrations – 24 SDCCH and 4 SDCCH). SDCCH holding time is 3 seconds.
- Figure 8 - right: function of SDCCH holding time; 2 – 8 seconds. Time between call reattempts is 5 seconds (3 illustrations – 24 SDCCH, 8 SDCCH and 4 SDCCH).
- Figure 9: function of number of SDCCH – 4, 8, 12, 16, 20 and 24. SDCCH holding time is assumed to be 3 seconds. Time between call reattempts is 5 seconds in left part, while that variable is 20 seconds in right part.

During commercial operations one would commonly have a set of blocking requirements that should be met. However, overload situations would likely be exceptional allowing for lower success probabilities. The results give indications on that even fairly small groups of mobile users, \( M = 300 \) in these cases, may lead to very low success ratios.
Figure 8 Overall success probability:

Left: as function of mobile station “think time”, i.e. time between repeated call attempts – “think time” from 0 to 20 seconds. 300 mobile users. Upper curve 24 SDCCH, lower curve 4 SDCCH. SDCCH holding time 3 second

Right: function of SDCCH holding time (from 2 seconds till 8 seconds). Mobile “think time” is 5 seconds. 300 mobile users. Upper curve 24 SDCCH, intermediate curve 8 SDCCH, lower curve 4 SDCCH.

Figure 9 Overall success probability

Left: as function of number of SDCCH 4, 8, 12, 16, 20 and 24. Time between call reattempts (“think time”) is 5 seconds. SDCCH holding time 3 seconds.

Right: as function of number of SDCCH 4, 8, 12, 16, 20 and 24. Time between call reattempts (“think time”) is 20 seconds. SDCCH holding time 3 seconds.
5. CONCLUDING REMARKS

This paper describes a rough approximate model and analysis approach for estimating the probability that a request gets through the RACH and SDCCH during high load situations. Comparisons with simulations have been done for several of the results, showing that the accuracy well below 10 % deviation, although not included here. It also turned out that a cause for any discrepancy mainly stemmed from the SDCCH calculations. Hence, some modifications to the Markovian arrival process could looked into (e.g. applying Hayward’s approximation). However, this requires that more moments are know for the traffic flow after passing the RACH.

Still, the rough modelling/analysing approach presented here allows for estimating the blocking probabilities of even reaching the state of requesting a traffic channel or transmitting an SMS.

REFERENCES