Modelling the time-varying cell capacity in LTE networks

Bart Sas · Elena Bernal-Mor · Kathleen Spaey · Vicent Pla · Chris Blondia · Jorge Martinez-Bauset

Keywords LTE · Time-varying cell capacity · Quality of Service · Admission control optimisation

Abstract In wireless orthogonal frequency-division multiple access (OFDMA) based networks like Long Term Evolution (LTE) or Worldwide Interoperability for Microwave Access (WiMAX) a technique called

Bart Sas
PATS Research Group
University of Antwerp/IBBT
2020 Antwerp, Belgium
E-mail: bart.sas@ua.ac.be

Elena Bernal-Mor
GIRBA Research Group
Universitat Politècnica de València/ITACA
46022 Valencia, Spain
E-mail: elbermo@upvnet.upv.es

Kathleen Spaey
PATS Research Group
University of Antwerp/IBBT
2020 Antwerp, Belgium
E-mail: kathleen.spaey@ua.ac.be

Vicent Pla
GIRBA Research Group
Universitat Politècnica de València/ITACA
46022 Valencia, Spain
E-mail: vpla@dcom.upv.es

Chris Blondia
PATS Research Group
University of Antwerp/IBBT
2020 Antwerp, Belgium
E-mail: chris.blondia@ua.ac.be

Jorge Martinez-Bauset
GIRBA Research Group
Universitat Politècnica de València/ITACA
46022 Valencia, Spain
E-mail: jmartinez@upvnet.upv.es

adaptive modulation and coding (AMC) is applied. With AMC, different modulation and coding schemes (MCSs) are used to serve different users in order to maximise the throughput and range. The used MCS depends on the quality of the radio link between the base station and the user. Data is sent towards users with a good radio link with a high MCS in order to utilise the radio resources more efficiently while a low MCS is used for users with a bad radio link. Using AMC however has an impact on the cell capacity as the quality of a radio link varies when users move around; this can even lead to situations where the cell capacity drops to a point where there are too little radio resources to serve all users. AMC and the resulting varying cell capacity notably has an influence on admission control (AC). AC is the algorithm that decides whether new sessions are allowed to a cell or not and bases its decisions on, amongst others, the cell capacity. The analytical model that is developed in this paper models a cell with varying capacity caused by user mobility using a continuous-time Markov chain (CTMC). The cell is divided into multiple zones, each corresponding to the area in which data is sent towards users using a certain MCS and transitions of users between these zones are considered. The accuracy of the analytical model is verified by comparing the results obtained with it to results obtained from simulations that model the user mobility more realistically. This comparison shows that the analytical model captures the varying cell capacity very accurately; only under extreme conditions differences between the results are noticed.

The developed analytical and simulation models are then used to investigate the effects of a varying cell capacity on AC. The analytical and simulation models are also used to study an optimisation algorithm that adapts the parameter of the AC algorithm which de-

termines the amount of resources that are reserved in order to mitigate the effects of the varying cell capacity. Updating the parameter of the AC algorithm is done by reacting to certain triggers that indicate good or bad performance and adapt the parameters of the AC algorithm accordingly. Results show that using this optimisation algorithm improves the quality of service (QoS) that is experienced by the users.

1 Introduction

Contemporary wireless networks such as Long Term Evolution (LTE), the next generation of cellular networks, or Worldwide Interoperability for Microwave Access (WiMAX), a standard for wireless broadband internet access, try to optimise performance by employing adaptive modulation and coding (AMC) [5]. AMC allows different modulation and coding schemes (MCSs) to be used at different points in time depending on the signal quality. By using different MCSs data can be sent to the users at higher or lower bit rates as they move around in a cell. Since the capacity of a cell is determined by the number of active users and the throughput that can be achieved by each of these users, changing the MCS will cause the cell capacity to vary as the number of active users changes (sessions are started and stopped) and as users move around. This varying of the cell capacity might influence the decisions taken by admission control (AC), which is the algorithm that decides if a session is admitted to the cell or not. It bases its decisions on the availability of the resources that are needed to guarantee the quality of service (QoS) of the new session and the already accepted sessions. Due to the time-varying cell capacity, it is for instance possible that at one point in time there is sufficient capacity to provide the desired QoS to all users while at another point in time variations in the cell capacity cause the cell capacity to drop below the required amount. When designing AC algorithms it is important to take this possibility into account.

In this paper we will create an analytical model for studying the effects of varying cell capacity on AC. We consider a single cell in which users move around. When a new session arrives it is subjected to AC which will decide if the user is admitted to the cell or not. Only one, real-time, service is considered. Although for our purposes it is not of great importance which traffic direction is considered, we consider only the downlink traffic direction. Users that are accepted remain active for a certain amount of time. In order to maintain a service with a fixed bitrate, active users require a varying number of resources depending on their location in the cell. The modelling assumptions will then be verified using simulations that model the mobility of the users and the session duration more accurately.

The developed models will afterwards be used to evaluate the performance of a simple AC scheme and the performance of an AC optimisation algorithm that adapts the parameters of the AC algorithm in order to guarantee QoS for existing sessions depending on the network load in order to obtain better performance.

Analytical models for studying AC in cellular networks have already been studied in several papers, but most of them do not consider the variation of the cell capacity due to user movement. For example, in [18] and [10] the efficiency of several AC policies in cellular networks is studied. An AC algorithm where users requesting admission are gradually admitted is suggested in [6] and an algorithm for finding the optimal thresholds within the class threshold type AC policies is proposed in [20].

An analytical model considering varying cell capacity is described in [8], three AC policies for capacity-varying networks are discussed in [21] and an AC algorithm that takes into account the mobility of the users is proposed in [9]. Studying the effects of varying cell capacity on AC using simulations has already been done in a number of papers. In [19] an AC scheme for WiMAX is proposed and its performance is evaluated and [22] studies self-optimisation of an AC scheme for LTE.

Algorithms that optimise the parameters of radio resource management (RRM) algorithms have already been the subject of a number of research papers [17,23, 22,11] and projects such as SOCRATES [3], MONOTAS [2] and E^3 [1]. These papers and projects however used simulations instead of analytical models. Industrial lobbies like Next Generation Mobile Networks (NGMN) [16] promote the development of self-optimising networks (SONs). The contributions of this paper are that the mobility assumptions assumed in the analytical model proposed considering varying cell capacity are checked using simulations, the cases and the reason why both models differs are identified and the analytical model is used to study the performance of a dynamic AC algorithm proposed.

This paper is structured as follows: Section 2 explains how AMC influences the cell capacity in wireless networks. In Section 3 the general modelling assumptions as well as the AC and AC optimisation algorithms that are studied are presented. Section 4 discusses the analytical model. The simulation model that is used to validate the mobility and session duration assumptions made in the analytical model is described in Section 5. In Section 6 compares and discusses the numerical re-

sults for different AC policies obtained with both the analytical and the simulation models. Section 7 concludes this paper and mentions some future work.

2 Varying cell capacity

2.1 Adaptive modulation and coding

In this paper we will consider AMC as it is implemented in LTE, although the same principles also apply to AMC in other network technologies. LTE defines 15 different MCSs, which are obtained by combining a number of code rates with 3 modulation schemes: quadrature phase-shift keying (QPSK), and quadrature amplitude modulation (QAM) with 16 and 64 states per symbol.The modulation scheme determines how binary data (bits) is converted to a signal. The code rate represents the ratio of information bits to the total amount of bits (information bits + forward error correction (FEC) bits) that is used to transmit the data. The lower the signal quality, the lower the code rate has to be, (i.e. more FEC bits are needed per information bit) in order to tackle the high number of erroneous bits. An overview of the MCSs that are used in LTE can be found in Table 1.

The choice of which MCS is used is based on measurements of the signal quality that are provided by the user equipment (UE) to the eNodeB (eNB). Based on the modulation scheme and the code rate, the theoretical maximum bit rate for each MCS can be calculated. Using the attenuated Shannon bound [5] the minimum signal to interference ratio (SIR) that is needed to be able to use a certain MCS can be determined. The SIR is the ratio (in the linear domain) of the received signal strength (S) at the UE of the source eNodeB (SeNB) to the total strength of the signals of the interfering neighbouring eNodeBs (NeNBs) (I_i): $SIR = \frac{S}{\sum_i I_i}$. The higher the SIR, the better the quality of the signal. The strengths of the received signals S and I_i in turn depend on a number of factors. First of all there is the power at which the signals are transmitted. The most dominant factor in the reduction of the transmitted signal power is the pathloss. Pathloss is the decreasing of the signal strength because of spatial dispersion and is linked to the distance between the transmitter and the receiver. The received signal strength is also influenced by other factors like shadow fading and multi-path fading. As the pathloss has the biggest impact on the received signal strength and thus also on the SIR, the SIR will especially depend on the relative distance from the SeNB and the interfering NeNBs. If we consider a site that is surrounded by a number of interfering sites, consider-

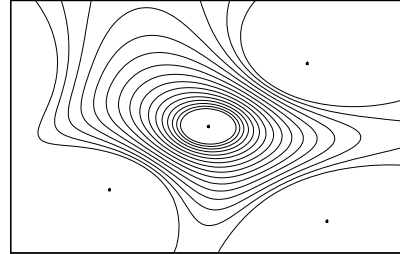ing only pathloss, the lines with equal SIR will be cycles around the SeNB (see Fig. 1).

Fig. 1 Lines of equal interference surrounding a site.

Table 1 The different MCSs used by LTE [4].

#	Modulation	Bits/symbol	Code rate
1	QPSK	2	78/1024
2	QPSK	2	120/1024
3	QPSK	2	193/1024
4	QPSK	2	308/1024
5	QPSK	2	449/1024
6	QPSK	2	602/1024
7	16-QAM	4	378/1024
8	16-QAM	4	490/1024
9	16-QAM	4	616/1024
10	64-QAM	6	466/1024
11	64-QAM	6	567/1024
12	64-QAM	6	666/1024
13	64-QAM	6	772/1024
14	64-QAM	6	873/1024
15	64-QAM	6	948/1024

2.2 Determining the cell capacity

LTE uses orthogonal frequency-division multiple access (OFDMA). The frequency domain is divided into non-overlapping sub-channels of 180 kHz and the time domain in slots with a duration of 1 ms. These subdivisions of time and frequency are referred to as scheduling resources (SRs) in this paper. The SRs are distributed among all ongoing sessions by a scheduling algorithm which is implemented at the eNB. In this paper we assume a simple scheduler that makes optimal use of the available resources (i.e. SRs will only be left unused if there is no data to send). In case there are less SRs

available than there are required, the scheduler divides the available SRs among the users, proportional to the amount of SRs they require. The number of SRs a user needs depends on its location. This means that when a user moves around in a cell, the number of SRs that are needed to send a certain amount of bits towards that user will change.

The total amount of SRs that are needed for all active users can be calculated as follows. Suppose r_k is the required bitrate by user k and b_k is the amount of bits that can be sent towards that user in a single SR. The average number of SRs per second (n_k) that are needed by user k is given by: $n_k = \frac{r_k}{b_k}$. The total amount of SRs that are needed in order to serve all users is: $R = \sum_k n_k = \sum_k \frac{r_k}{b_k}$. Note that, using the scheduler described above, the cell capacity can be estimated as follows. If the required number of SRs per second is smaller than the available amount of SRs per second ($R < R_A$), the factor $\frac{R_A}{R}$ has been added to take into account the capacity of unassigned SRs. In this case each user will be given the bit rate it requires and the cell capacity is defined as $C = \frac{R_A}{R} \sum_k r_k$. If there are insufficient SRs the scheduler will divide the available amount of resources among the users, proportionally to the number of SRs they require. I.e. the amount of SRs a user k receives will be $\frac{n_k}{R} R_A$ and therefore the cell capacity is given by $C = \sum_k \frac{R_A}{R} b_k n_k = \frac{R_A}{R} \sum_k r_k$. So, in either case, the cell capacity is given by: $C = \frac{R_A}{R} \sum_k r_k$.

3 General Model

In order to model that the cell capacity varies depending on the location of the users we consider a single cell that is divided into N concentric rings called zones, where zone 1 is the outermost zone (see Fig. 2). Denote by ρ_i the bits transmitted per SR to a user in zone i (with $\rho_1 < \rho_2 < \cdots < \rho_N$). The cell has R_A SRs available per second and an active user requires a bitrate of $r_k = r$ bits per second to fulfil its QoS requirements. Therefore, the number of SRs per second that a user needs to achieve its required bitrate r in zone i is: $s_i = \frac{r}{\rho_i}$.

For evaluating the performance of the models, two performance measurements are defined. These are the total blocking probability (P_T) and the low QoS probability (P_{QoS}). The total blocking probability is the probability that a session that arrives is blocked by the AC. The low QoS probability is the fraction of time that active users experience a low QoS, i.e. the bitrate they receive is lower than r.
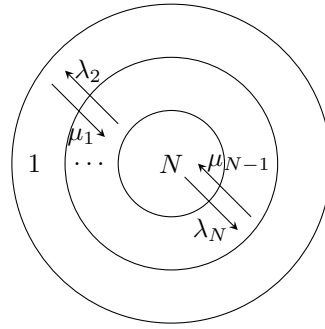

Fig. 2 Different zones with transition rates.

3.1 Admission Control Policy

In order to guarantee a minimum QoS (i.e. a minimum bitrate of r) for the users that are in the system, the acceptance of new arrivals is controlled by an AC policy. Let $f \in [0, 1]$ denote the AC threshold that determines which fraction of resources under which new sessions are accepted by the AC algorithm. A new session is accepted if after accepting the session there would still be more than $(1 - f)R_A$ SRs available.

Therefore, to decide on the acceptance of a new session in zone i, the following decisions are taken:

$$R + s_i \begin{cases} \leq fR_A & \text{session accepted} \\ > fR_A & \text{session blocked} \end{cases} \qquad (1)$$

Once a user is admitted to the system we assume that it cannot be removed before its session ends. If a user makes an outward zone-transition, it can happen that more SRs than there are available are needed ($R > R_A$) since users need more SRs in the destination zone than in the zone where they came from to maintain their bitrate. In this case, the scheduler will give all the users a share of the available SRs that is proportional to the amount they requested, as is mentioned in Section 2.2. So all users will receive a lower bitrate than the required r, hence all users will be served with a lower QoS.

3.2 AC optimisation algorithm

In this paper we will also study the effects of an algorithm (which will be referred to as the AC optimisation algorithm) that tunes the parameter f of the AC algorithm. The goal of this algorithm is to dynamically adapt the parameters of the AC algorithm to changes in the environment. When for instance the load is high more resources will be reserved for ongoing sessions while when the load is low, more new sessions will be admitted to the cell.

The optimisation algorithm used in this paper will, at certain time instances, check the current load of the system. Based on this load the AC optimisation algorithm decides how it will update the parameter f. If, on the time instances the load is checked, the load is high, f will be lowered in order to reserve more resources for ongoing sessions while blocking more new sessions. If on the other hand the load is considerably low, meaning that only little resources are in use, f will be raised as to allow more sessions to the system and have a higher resource utilisation.

The load is considered to be high when there are insufficient resources available to serve all sessions that are admitted to the cell with their required bitrate r, i.e., when $R > R_A$. The load is considered to be considerably low when the total amount of resources that are in use is less than a certain fraction g of the fraction f of the resources under which new sessions are accepted by the AC algorithm, i.e., when $R < fgR_a$. The fraction g is a system parameter with a predefined value. The raising and lowering of f is done in discrete, evenly sized steps denoted by the parameter Δf. The parameter f is also bounded by a lower and upper limit, denoted f_m and f_M respectively. The number of different discrete values that f can take is denoted by n_f. If f_m denotes the lower limit of f, the upper limit of f is $f_M = f_m + (n_f - 1)\Delta f$.

The algorithm that decides whether f is raised or lowered is given by:

$$\frac{R}{R_A} \begin{cases} > 1 & f \leftarrow \max(f - \Delta f, f_m) \\ < fg & f \leftarrow \min(f + \Delta f, f_M) \\ \text{otherwise} & \text{leave } f \text{ unchanged} \end{cases} \quad (2)$$

4 Analytical Model

We first present the analytical model for the static AC policy, later we will present an extension of this model considering the AC optimisation algorithm.

4.1 Static AC algorithm

We model the proposed system with the static AC policy using a multidimensional continuous-time Markov chain (CTMC), where the system state vector is described by the N-tuple $\boldsymbol{x} = (x_1, \ldots, x_N)$, where x_i represents the number of users in zone i. Since the number of bits that can be transmitted per SR is the highest for a user in zone N, the maximum number of sessions M that can be present in the system is determined by the maximum number of sessions accepted in this zone

when there are no active users in the other zones. The set of feasible states is thus given by:

$$W := \left\{ \boldsymbol{x} : x_i \in \mathbb{N}; \quad \sum_{i=1}^{N} x_i s_N \leq f R_A \right\}. \quad (3)$$

The total number of SRs that are needed per second to serve all the users at the required bitrate r when the system is in state $\boldsymbol{x}$ is represented by $R(\boldsymbol{x}) = \sum_{i=1}^{N} x_i s_i$.

For the sake of mathematical tractability we make the common assumptions that new sessions arrive according to a Poisson arrival process with arrival rate ϵ and exponentially distributed session durations with rate γ. Assuming uniformly distributed traffic, if A_T is the total area of the cell and A_i is the area corresponding to zone i (i.e. $A_T = \sum_i A_i$), we can consider that the arrival rate for new sessions in zone i is $\epsilon_i = \frac{A_i}{A_T}\epsilon$. We checked this assumption with simulations for the random walk mobility model that will be described in Section 5.

The zone residence time, i.e the time that a user stays in a certain zone before entering another one, is also assumed to be exponentially distributed with rate λ_i for transitions from zone i to zone $i-1$ where $i = 2, \ldots, N$ and μ_i for transitions from zone i to zone $i+1$ where $i = 1, \ldots, N-1$ (see Fig. 2). We assume that the users start their sessions inside the cell and do not leave the cell until their session is finished.

The function $a_i(\boldsymbol{x})$ denotes whether a session that arrives in zone i when the system is in state $\boldsymbol{x}$ is accepted by the AC algorithm or not ($a_i(\boldsymbol{x}) = 1$ means that the session is accepted and $a_i(\boldsymbol{x}) = 0$ means that the session is blocked).

Figure 3 shows a CTMC for $N = 2$ as an example. The notation has been simplified as $a_i(\boldsymbol{x}) = a_i$. This system has only two zones $i = 1, 2$ and therefore one incoming rate μ_1 and one outgoing rate λ_2. Recall that M is the maximum number of users that can be active in the system. If we define phases as the number of users in zone $i = 2$ and levels as the number of users in zone $i = 1$, we can study the model as a finite level-dependent quasi-birth-and-death (QBD) process [14] with $M+1$ levels where level h ($h = 0, \ldots, M$) has $M+1-h$ phases. Therefore we can construct the transition rate matrix $\boldsymbol{Q}$ with a block-tridiagonal form, see (4). The first row of blocks corresponds to level $h = 0$, the second row of blocks to level $h = 1$, etc., where blocks $\boldsymbol{Q_1^h}$ correspond to transitions between phases in level h, blocks $\boldsymbol{Q_0^h}$ to transitions from level h to level $h+1$ and blocks $\boldsymbol{Q_2^h}$ to transitions from level h to level $h-1$.
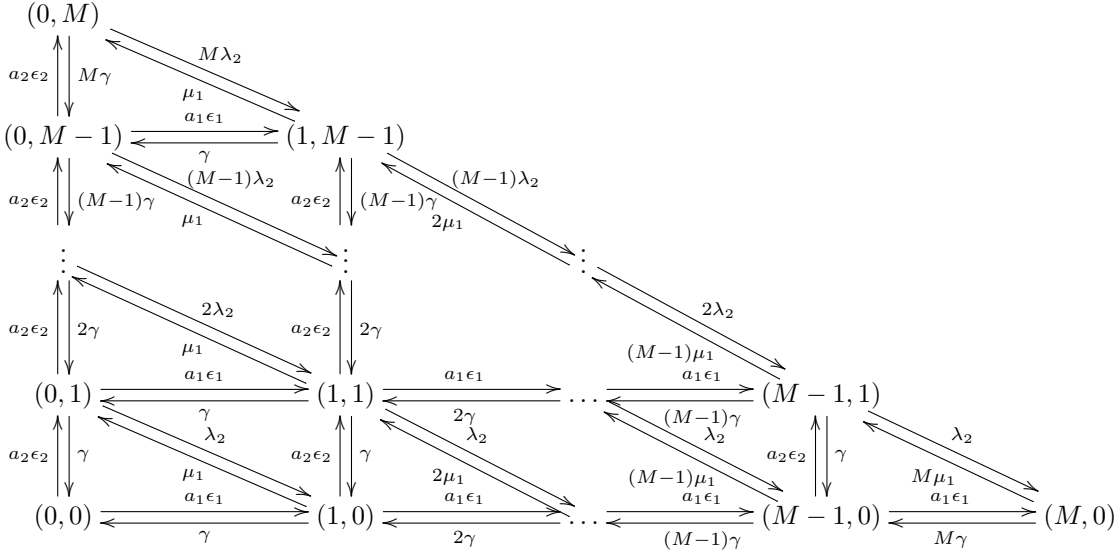
Fig. 3 Transition diagram of the bi-dimensional model.

$$Q = \begin{bmatrix} Q_1^0 & Q_0^0 & 0 & 0 & 0 & \cdots \\ Q_2^1 & Q_1^1 & Q_0^1 & 0 & 0 & \cdots \\ 0 & Q_2^2 & Q_1^2 & Q_0^2 & 0 & \cdots \\ & & \ddots & \ddots & \ddots & \\ \cdots & 0 & 0 & Q_2^{M-1} & Q_1^{M-1} & Q_0^{M-1} \\ \cdots & 0 & 0 & 0 & Q_2^M & Q_1^M \end{bmatrix} \qquad (4)$$

Note that the blocks of Q have different sizes for different levels h. The total size of the transition rate matrix Q for $N = 2$ is:

$$M_T^2 = \sum_{h=0}^M M + 1 - h = \frac{M^2 + 3M + 2}{2}.$$

In the case of $N = 3$ zones, we can define a layered-level structure with phases and 2 level layers, where phases are defined as the number of users in zone $i = 3$, low-levels as the number of users in zone $i = 2$ and high-levels as the number of users in zone $i = 1$. The model is a three-dimensional finite QBD process where the transition rate matrix Q again follows the structure of (4). Moreover the block matrices Q_0^h, Q_1^h and Q_2^h are also constructed with a block-tridiagonal form, see (5), (6) and (7).

$$Q_1^h = \begin{bmatrix} A_1^{h,0} & A_0^{h,0} & 0 & \cdots \\ A_2^{h,1} & A_1^{h,1} & A_0^{h,1} & \cdots \\ \ddots & \ddots & \ddots & \\ \cdots & 0 & A_2^{h,M-h} & A_1^{h,M-h} \end{bmatrix} \qquad (5)$$

$$Q_0^h = \begin{bmatrix} B_1^{h,0} & 0 & 0 & \cdots \\ B_2^{h,1} & B_1^{h,1} & 0 & \cdots \\ \ddots & \ddots & & \\ \cdots & 0 & 0 & B_2^{h,M-h} \end{bmatrix} \qquad (6)$$

$$Q_2^h = \begin{bmatrix} C_1^{h,0} & C_0^{h,0} & 0 & \cdots \\ 0 & C_1^{h,1} & C_0^{h,1} & \cdots \\ \ddots & \ddots & & \\ \cdots & 0 & C_1^{h,M-h} & C_0^{h,M-h} \end{bmatrix} \qquad (7)$$

Note that the blocks have different sizes for different levels h and l. Blocks $A_1^{h,l}$ correspond to transitions among phases inside high-level h and low-level l, blocks $A_0^{h,l}$ correspond to transitions from low-level l to low-level $l+1$ inside high-level h and blocks $A_2^{h,l}$ correspond to transitions from low-level l to low-level $l-1$ inside high-level h. Blocks $B_1^{h,l}$ correspond to transitions from high-level h to high-level $h+1$ with low-level l, blocks $B_2^{h,l}$ correspond to transitions from high-level h to high-level $h+1$ and from low-level l to low-level $l-1$. Blocks $C_1^{h,l}$ correspond to transitions from high-level h to high-level $h-1$ with the same low-level l and blocks $C_0^{h,l}$ correspond to transitions from high-level h to high-level $h-1$ and from low-level l to low-level $l+1$. For more details see Appendix A. The total size of the transition rate matrix Q is:

$$M_T^3 = \sum_{h_1=0}^M \sum_{h_2=0}^{M-h_1} M - h_1 + 1 - h_2 =$$
$$= \frac{2M^3 + 12M^2 + 22M + 12}{12}.$$

This block design can be generalised to any number of zones N by constructing matrix blocks inside matrix blocks with $N-1$ different level layers.

To solve the finite QBD Markov process and obtain the steady state vector π we use the linear level reduction (LLC) algorithm [13]. Basically, this algorithm has two stages. First, the state space is reduced by removing one high-level at each step until there is a Markov process on the last high-level left. That Markov process is solved and the stationary vector is constructed in the second stage by adding back one high-level at each step. Note that despite the rate matrices being large, their sparseness makes the computations feasible. The complexity of the algorithm grows with N as the size of the transition matrix is $O(M^N)$.

Let us denote by P_i the blocking probability for new arrivals in zone i and the total blocking probability in the system by P_T. Then:

$$P_i = \sum_{\boldsymbol{x} \in W} (1 - a_i(\boldsymbol{x}))\pi(\boldsymbol{x}); \quad P_T = \frac{\sum_{i=1}^{N} \epsilon_i P_i}{\epsilon}. \tag{8}$$

Let $I(\boldsymbol{x})$ denote the indicator function which takes the value 1 when $R(\boldsymbol{x}) > R_A$. The low QoS probability is then given by:

$$P_{QoS} = \sum_{\boldsymbol{x} \in W} I(\boldsymbol{x})\pi(\boldsymbol{x}). \tag{9}$$

4.2 AC optimisation algorithm

The system with the AC optimisation algorithm is also modelled using a multidimensional CTMC. In this case the system state vector is described by the $(N+1)$-tuple $\boldsymbol{x} = (x_1, \ldots, x_N, f)$. In this system state vector x_i represents the number of users in zone i and f represents the value of the AC threshold which can take the values $f_m, f_m + \Delta f, f_m + 2\Delta f, \ldots, f_M$. The intervals between two optimisations, i.e., the intervals after which Eq. (2) is checked and the appropriate action is taken, are considered to be exponentially distributed with mean $1/\eta$.

As with the static algorithm, the number of bits that can be transmitted per SR is the highest for a user in zone N and the maximum number of sessions M in the system is determined by the maximum number of sessions accepted in zone N when there are no active users in the other zones and $f = f_M$, the upper limit of f. The set of feasible states is thus given by:

$$W := \left\{ \boldsymbol{x} : \quad x_i \in \mathbb{N}, \quad \sum_{i=1}^{N} x_i s_N \leq f_M R_A; \right.$$
$$\left. f \in \{f_m, f_m + \Delta f, \ldots, f_M\} \right\}.$$

We use the same system parameters and make the same assumptions as when the static AC algorithm is used.

Again, we can study the model as a finite level-dependent QBD process. In the case of $N = 3$ zones, the model is a four dimensional finite QBD process where the transition rate matrix $\boldsymbol{Q}$ follows the structure of (4). But in this case, we define a layered-level structure with phases and 3 level layers, where the phases are defined as the value of f, low-levels as the number of users in zone $i = 3$, medium-levels as the number of users in zone $i = 2$ and high-levels as the number of users in zone $i = 1$. Thus, the block matrices $\boldsymbol{Q}_0^h$, $\boldsymbol{Q}_1^h$ and $\boldsymbol{Q}_2^h$ also follow the block design defined in (5), (6) and (7). Moreover, the matrices $\boldsymbol{A}_0^{h,m}$, $\boldsymbol{A}_1^{h,m}$ and $\boldsymbol{A}_2^{h,m}$ follow again a block design, as defined in (10), (11) and (12) respectively, where $p = h + m$. For more details see Appendix B.

$$\boldsymbol{A}_1^{h,m} = \begin{bmatrix} \boldsymbol{D}_1^{h,m,0} & \boldsymbol{D}_0^{h,m,0} & \boldsymbol{0} & \cdots \\ \boldsymbol{D}_2^{h,m,1} & \boldsymbol{D}_1^{h,m,1} & \boldsymbol{D}_0^{h,m,1} & \cdots \\ \ddots & \ddots & \ddots & \\ \cdots & \boldsymbol{0} & \boldsymbol{D}_2^{h,m,M-p} & \boldsymbol{D}_1^{h,m,M-p} \end{bmatrix} \tag{10}$$

$$\boldsymbol{A}_0^{h,m} = \begin{bmatrix} \boldsymbol{E}_1^{h,m,0} & \boldsymbol{0} & \boldsymbol{0} & \cdots \\ \boldsymbol{E}_2^{h,m,1} & \boldsymbol{E}_1^{h,m,1} & \boldsymbol{0} & \cdots \\ \ddots & \ddots & & \\ \cdots & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{E}_2^{h,m,M-p} \end{bmatrix} \tag{11}$$

$$\boldsymbol{A}_2^{h,m} = \begin{bmatrix} \boldsymbol{F}_1^{h,m,0} & \boldsymbol{F}_0^{h,m,0} & \boldsymbol{0} & \cdots \\ \boldsymbol{0} & \boldsymbol{F}_1^{h,m,1} & \boldsymbol{F}_0^{h,m,1} & \cdots \\ \ddots & \ddots & & \\ \cdots & \boldsymbol{0} & \boldsymbol{F}_1^{h,m,M-p} & \boldsymbol{F}_0^{h,m,M-p} \end{bmatrix} \tag{12}$$

The matrix $\boldsymbol{A}_1^{h,m}$ is a square matrix of size $n_f(M+1-p) \times n_f(M+1-p)$, the size of $\boldsymbol{A}_0^{h,m}$ is $n_f(M+1-p) \times n_f(M-p)$ and the size of $\boldsymbol{A}_2^{h,m}$ is $n_f(M+1-p) \times n_f(M+2-p)$. Note that $\boldsymbol{A}_0^{h,m}$ and $\boldsymbol{A}_2^{h,m}$

are not square matrices. The total size of the transition rate matrix $\boldsymbol{Q}$ is:

$$M_T^A = \sum_{h_1=0}^{M} \sum_{h_2=0}^{M-h_1} n_f(M - h_1 + 1 - h_2) =$$

$$= n_f \cdot \frac{2M^3 + 12M^2 + 22M + 12}{12}.$$

This block design can be generalised to any number of zones N by constructing matrix blocks inside matrix blocks with N different level layers.

The blocking probability P_i for new arrivals in zone i, the total blocking probability in the system P_T and the low QoS probability P_{QoS} are again given by (8) and (9).

5 Simulation model

In order to verify the mobility and session duration modelling assumptions made in the analytical model and to verify the results obtained with the analytical model, simulations that model the mobility and the session duration of the users more realistically are performed. In the simulations, sessions are generated according to a Poisson process with arrival rate ϵ. The duration of a session is, unlike in the analytical model, chosen from a log-normal distribution as this distribution more realistically models the duration of sessions [12]. The mean of this log-normal distribution is taken equal to $1/\gamma$ and the variance to $1/\gamma^2$, i.e. the log-normal parameters are chosen such that this log-normal distribution has the same mean and variance as the exponential distribution considered in the analytical model. When a session is generated, it is placed uniformly in the cell and is subjected to AC. If it is admitted to the cell, it starts moving around. Users move around according to a random walk mobility model [7]. This means that when a session is started, it chooses a direction ϕ (in radians) uniformly distributed in the interval $[0, 2\pi[$ and starts moving in the chosen direction at a fixed velocity v. After the user has travelled over a fixed distance d, it again chooses a direction and starts moving in the newly chosen direction. This is repeated until the session finishes and the user is removed from the system. When a user reaches the boundary of the cell, it bounces against the circular edge and continues its path in the reflection direction. Thus, the residence time in each zone is not modelled using random distributions. Instead, transitions between zones occur when a user crosses the border of a zone.

The session blocking probability (P_T) is calculated by counting the total number of generated sessions and

the number of sessions that are dropped by AC and dividing the latter by the former. The low QoS probability is calculated by recording the time that the system has a low QoS and dividing it by the total simulation time.

In the simulation model of the AC optimisation algorithm, the optimisation algorithm, when enabled, will check the load at regular time instances which are multiples of $1/\eta$. At these time the test of Eq. (2) will be performed and, if necessary, the appropriate action will be taken. In contrast to the analytical model, the simulation model uses fixed optimisation intervals; this is because fixed-length intervals are more commonly used in reality than exponentially distributed ones.

6 Results

In this section we present the results of a comparative study between the analytical and the simulation models. We also discuss the performance results of the AC scheme presented in Section 3.1 and the AC optimisation algorithm presented in Section 3.2. Different scenarios were simulated for various settings of the parameters. In these simulations the mean session inter-arrival time ($1/\epsilon$), mean session duration ($1/\gamma$), distance travelled by the users in a single leg of the mobility model (d) and, in case of the static algorithm, AC threshold (f) were varied.

In the simulation users move around with a certain velocity and distance travelled in a single leg of the mobility model. The rates λ_i and μ_i that are used in the analytical model only depend on those parameters and the size of the zones. In order to determine these transition rates, simulations with only one user were executed. In these simulations a single user walks around without starting any sessions. Each time the user crosses the border of a zone, the event is recorded and at the end of the simulation, the mean transition rates are calculated.

6.1 Model parameters

The parameters that are fed into the analytical and simulation models are based on the evaluation scenarios described in [15]. The carrier frequency f_c is chosen to be 2 GHz, the pathloss model that is associated with this frequency is $L = 37.6 \log_{10}(D) + 128.1$, where D is the distance between the eNB and the UE. The operating bandwidth is 5 MHz which means that there are 25 sub channels of 180 kHz (plus guard band), resulting in 25000 SRs per second. We model the traffic as a fluid flow with a bitrate of 128 kbit/s for scenarios involving

only the static AC algorithm and 256 kbit/s for scenarios involving the AC optimisation algorithm. Unless otherwise indicated, we consider $v = 30$ km/h, $d = 30$ m $1/\gamma = 300$ s, $1/\epsilon = 3$ s and $f = 1$. The radii of the zones can be determined from the maximum bitrates in the zones. Using the attenuated Shannon bound [5], the minimum SIR that is needed to achieve the bitrate corresponding to a MCS can be calculated according to (13) where β_i is the bitrate per Hertz in zone i and α is the attenuation factor which is 0.6 [5]:

$$\text{SIR}_i = 2^{\frac{\beta_i}{\alpha}} - 1. \tag{13}$$

Combining this equation with the pathloss model from [15] mentioned above, not taking noise into account and assuming that for every direction the interference comes from a single source at a distance D_{s2s} from the SeNB with the same transmit power (P_{Tx}) as the SeNB, an expression for the SIR (in the logarithmic domain) at a given distance from the SeNB (D) can be constructed:

$$\begin{aligned} \text{SIR} &= P_{\text{Tx}} - L_S - (P_{\text{Tx}} - L_I) = L_I - L_S \\ &= 37.6 \log_{10}(D_{\text{s2s}} - D) + 128.1 \\ &\quad - (37.6 \log_{10}(D) + 128.1) \\ &= 37.6 \log_{10}\left(\frac{D_{\text{s2s}}}{D} - 1\right) \end{aligned} \tag{14}$$

where L_S is the pathloss of the signal and L_I the pathloss of the interference. The site-to-site distance D_{s2s} is set to 500 m, this inter-site distance is related to cells in urban environments. As can be seen from (14) the SIR at a given distance from the SeNB does not depend on the transmit power of the eNBs, using the assumptions made in this paper. From (14) the radius D_i of zone i, given the SIR can be calculated:

$$D_i = \frac{D_{\text{s2s}}}{10^{\frac{\text{SIR}_i}{37.6}} + 1}. \tag{15}$$

By combining (13) and (15) the radius of a zone given the bitrate of the MCS can be calculated.

Ideally, 15 zones should be considered, each corresponding to a single MCS. Since using all 15 different MCSs would produce too much computational overhead, the cell was instead divided in 3 zones with equal areas (i.e. the size of each zone is $\frac{1}{3}$ of the size of the cell), where the ρ_i of each zone is calculated considering the zones that correspond to the different MCSs. The cell border itself coincides with the circle on which the SIR is 0 dB. As the 0 dB border is in the middle between the SeNB and a NeNB, this border lies on $\frac{D_{\text{s2s}}}{2} = 250$ m and it is inside the zone that corresponds to MCS 10. The number of bits per SR that is used in each of these

zones is the weighted average number of bits per SR of the areas that overlap with the 3 different zones.

For the AC optimisation algorithm Δf is set to 0.1 and f_m and f_M are set to 0.5 and 1 respectively. This means that the value of f can take the values 0.5, 0.6, 0.7, 0.8, 0.9 and 1. The mean optimisation interval $1/\eta$ is set to 60 s. The value of the optimisation threshold g is set to 0.8.

A summary of all the model parameters is given in Table 2.

Table 2 A summary of the model parameters

Parameter	Symbol	Value
Site-to-site distance	D_{s2s}	500 m
Available resources	R_A	2500 SR/s
Carrier frequency	f_c	2 GHz
Operating bandwidth	f_{BW}	5 MHz
Attenuation factor	α	0.6
Traffic source rate	r	128 kbit/s
		256 kbit/s
Mean session duration	$1/\gamma$	300 s
Mean session i/a time	$1/\epsilon$	3 s
AC threshold	f	1
Optimisation threshold	g	0.8
Minimum AC threshold	f_m	0.5
Maximum AC threshold	f_M	1
AC threshold step	Δf	0.1
User velocity	v	30 km/h
Mobility distance	d	30 m
Mean optimisation interval	$1/\eta$	60 s
Radius zone 1	D_1	250 m
Relative area zone 1	a_1	33 %
Bits/SR zone 1	ρ_1	154.81
Radius zone 2	D_2	204.12 m
Relative area zone 2	a_2	33 %
Bits/SR zone 2	ρ_2	349.87
Radius zone 3	D_3	144.33 m
Relative area zone 3	a_3	33 %
Bits/SR zone 3	ρ_3	466.59

6.2 Numerical results

In this section numerical results obtained with both the analytical and simulation models are presented and compared. All simulation results have been obtained by running 10 simulations per point. Fig. 4 shows the blocking probabilities P_T while varying the session inter-arrival time ($1/\epsilon$). The results of the analytical model are represented using a solid line while the results of the simulations are represented using crosses.

Increasing the mean inter-arrival time will cause the blocking probability P_T to decrease. When the mean inter-arrival time is low, more sessions will be started in a shorter amount of time. As there are only a fixed number of resources available, this will cause more sessions to be blocked by the system resulting in a higher blocking probability. As can be seen in Fig. 4, the results obtained with the analytical model and the simulations are very similar.
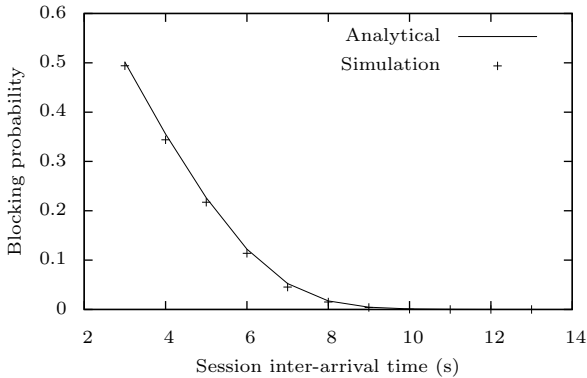

Fig. 4 Blocking probabilities for various mean session inter-arrival times.

Also the low QoS probability P_{QoS} obtained with the analytical model and with the simulations are very similar as can be seen in Fig. 5. As with the blocking probabilities, the QoS is worse when the session inter-arrival time is low than when the session inter-arrival time is high. The reason that the probability of a low QoS is higher when the session inter-arrival time is low is a consequence of the varying cell capacity: as the AC threshold f is set to 1 in these results, the system will be filled up until all SRs are used, implying that whenever the variations in the cell capacity cause more resources to be needed, the system will no longer be able to fulfil the QoS needs of the users.

Fig. 6 shows the total blocking probability when the mean session duration is varied. When sessions are longer (high $1/\gamma$) more sessions will be blocked as accepted sessions will be active in the system (and thus require resources) for a longer time. For the same reason the low QoS probabilities are higher when the session duration is longer as can be seen in Fig. 7. As with the case in which the mean inter-arrival time is varied, the results of the analytical model and the simulations are very similar in both figures.

Fig. 8 shows the same as Fig. 6 but this time the blocking probabilities of the individual zones are shown instead of the blocking probability for the entire cell (the legend does not discriminate between the analytical and the simulation model for presentation purposes,
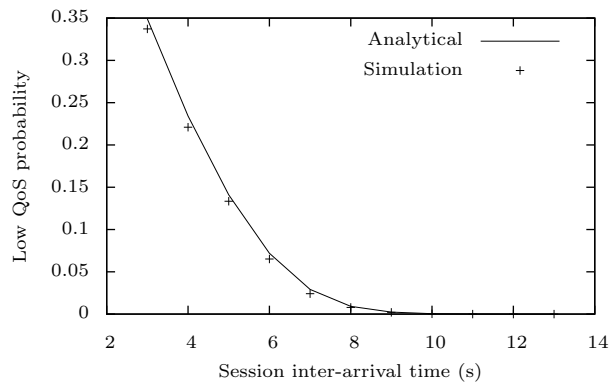

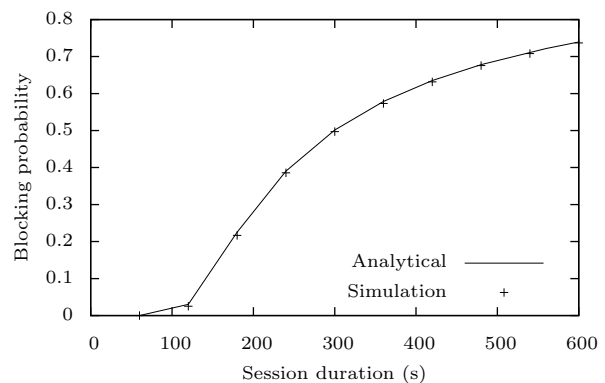**Fig. 5** Low QoS probability for various mean session inter-arrival times.

Fig. 6 Blocking probabilities for various mean session durations.
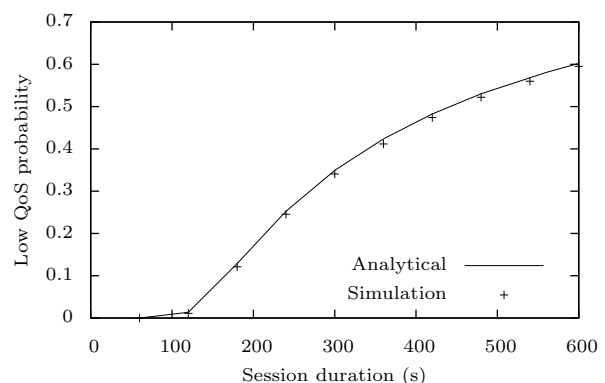

Fig. 7 Low QoS probabilities for various mean session durations.

but again the lines correspond to the results of the analytical model and the crosses to the results of the simulation model). As can be seen in this figure the analytical and simulation results for the individual zones again match very accurately, meaning that the analytical model is also accurate for the individual zones and not only for the entire cell.
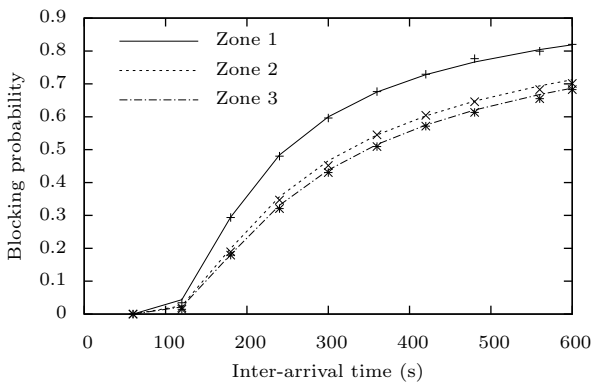
Fig. 8 Low QoS probabilities in the individual zones for various mean session durations.

6.3 Deviations between the analytical model and the simulations

Although the results obtained with the analytical and simulation models fit very well, there are occasions where there are deviations between both models. One of the cases in which the results of the analytical model and the simulations differ is when the distances travelled by the users in a single leg of the random walk mobility model (d) are either very short or very long in comparison to the radii of the zones. An example of this can be seen in Fig. 9. In this figure, the distance is
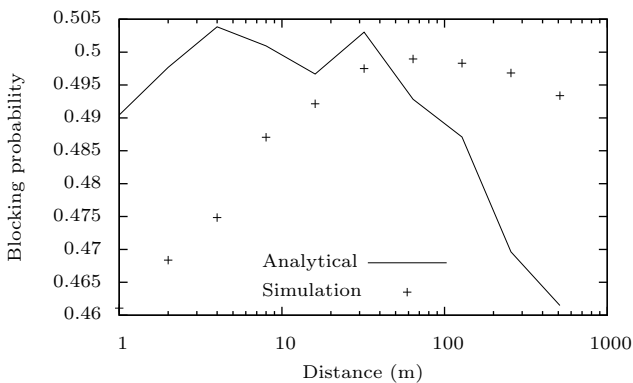

Fig. 9 The blocking probabilities for various distances travelled by the users.

varied between 1 m and 512 m (note that the cell radius is 250 m). As can be seen in this figure, the differences between the results obtained from the analytical model and the simulations are similar when the distance d lies between 10 m and 100 m. When d is either shorter or longer the deviations between both models become bigger. In the case of short d, this deviation is because users that are close to the border of a zone will cross the border of the zone many times while users that are further away from the border will most likely already

have finished their session before they make a transition because the short distance causes them to remain at nearly the same location. In the case of long d, this means that the distance that is travelled by the users is bigger than the radii of the zones and that the users will cross the zones more than once before choosing a new direction. This in turn has as a consequence that the time between entering a zone and leaving it again is bounded by the minimum and maximum distances that can be travelled in a zone in a straight line, divided by the velocity of the users. The time between entering and leaving a zone thus can no longer be either very long or very short as users will travel in a straight line through the zone and will not be able to change direction within a zone. This will have an influence on the distribution of the transition times. Figure 10 shows the distribution of the time users spend in zone 2 before going to zone 1. The times are measured in simulations where the distance travelled in a single leg are respectively 1 m, 32 m and 512 m. The plots also contain the probability density function (PDF) of the exponential distribution that is used in the analytical model to model the time that a user stays in that zone. As can be seen in the figures, the distribution of the times resembles the exponential distribution best when the travelled distance is around 30 m.

6.4 Analysis of the AC algorithm

In this section we briefly evaluate the AC algorithm that was used in this paper using the developed analytical and simulation models. Fig. 11 shows the blocking probabilities for various values of the AC threshold f. As can be seen in this figure the blocking probability decreases as the AC threshold increases. When the AC threshold increases, more sessions will be allowed to the cell, causing the blocking probability to decrease.
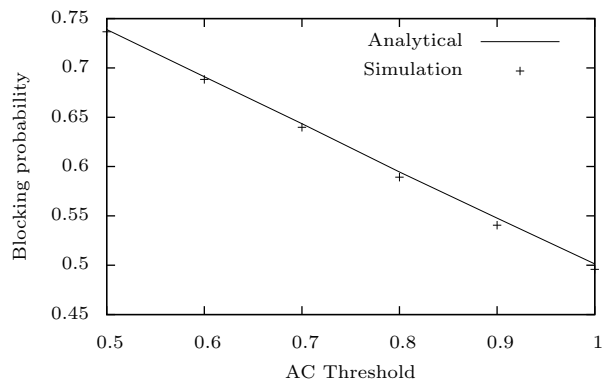

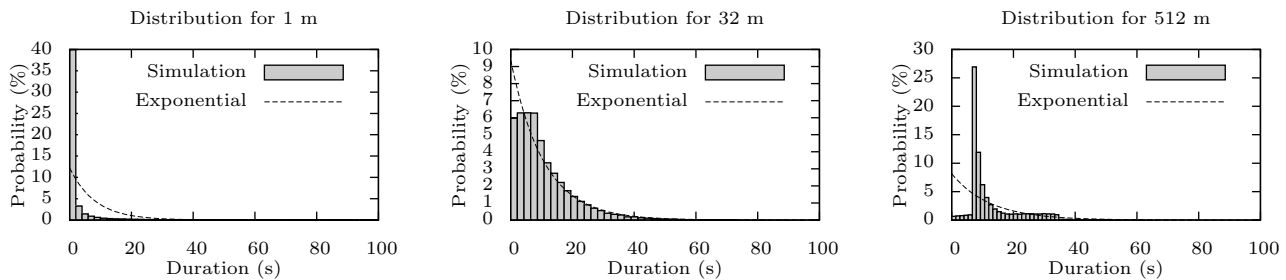**Fig. 11** Blocking probabilities for various AC thresholds f.

Fig. 10 Distribution of the time spent in zone 2.

When looking at the low QoS probability in Fig. 12 the QoS remains good until f reaches a value of more than 80 %. When f is higher than this value, the varying cell capacity causes the cell capacity to drop below the required capacity resulting in a low QoS. This shows the effects of the time-varying cell capacity on the QoS experienced by the active users.
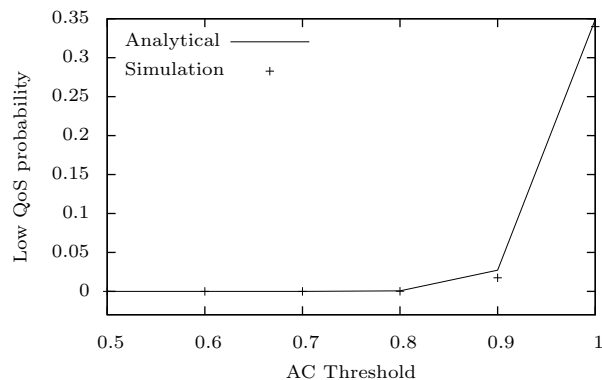

Fig. 12 Low QoS probabilities for various AC thresholds f.

various values of f. The slope of the blocking probability curve obtained with the AC optimisation algorithm is steeper than the slopes of the blocking probability curves of the static algorithm for short session durations. For long session durations the slope of the blocking probability curve obtained with the AC optimisation algorithm is flatter than the slope of the blocking probability curves obtained with the static algorithm.
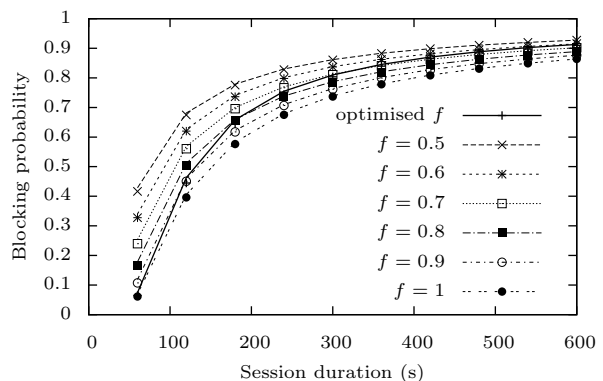

Fig. 13 Blocking probabilities for the AC optimisation algorithm.

6.5 Analysis of the AC optimisation algorithm

In this section we use both the analytical and simulation models to examine the performance of the AC optimisation algorithm. For this study, the blocking probability and low QoS probability are plotted against different values of the session duration. The results obtained with the AC optimisation algorithm are then compared against results obtained with the static algorithm for various values of the AC threshold f in order to assess the benefits of an optimisation algorithm relative to a static one. The values of f that are considered for the static algorithm are the values that can be assigned to f by the optimisation algorithm, i.e., $f \in \{0.5, 0.6, 0.7, 0.8, 0.9, 1\}$.

Fig. 13 shows the blocking probability for both the AC optimisation algorithm and the static algorithm for

When looking at the low QoS probability shown in Fig. 14 the AC optimisation algorithms even shows a more aberrant behaviour: in contrast to the static algorithm for which the low QoS probability rises as the session duration becomes longer, the bad QoS probability of the AC optimisation algorithm has a maximum at a mean session duration of 120 s and decreases again for longer session durations. The reason for the maximum around a session duration of 120 s is that in case the session duration is relatively short, high loads are unlikely and the optimisation algorithm will raise the threshold f to a high value. In the rare cases that the load does become too high (i.e. $R > R_A$), it will take the AC optimisation algorithm longer to react to this situation as the threshold f should be lowered starting from a high value while when the load is higher, the threshold will already have a lower value and the algorithm will react

swifter. When looking at the session inter-arrival time instead of the session duration (Fig. 15) a similar kind of behaviour can be observed. The trend in this plots is however reversed as long inter-arrival times mean that the load is lower while a high session duration means that the load is higher.
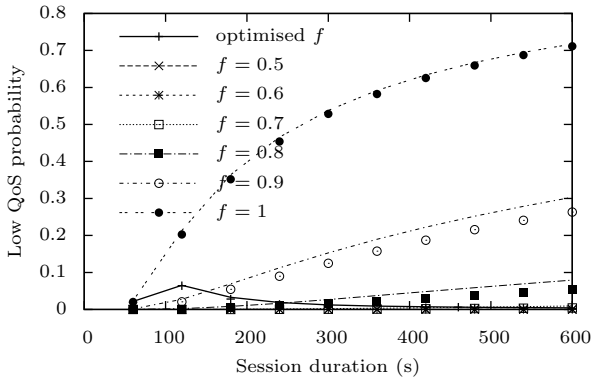

Fig. 14 Low QoS probabilities for various session duration for the AC optimisation algorithm.
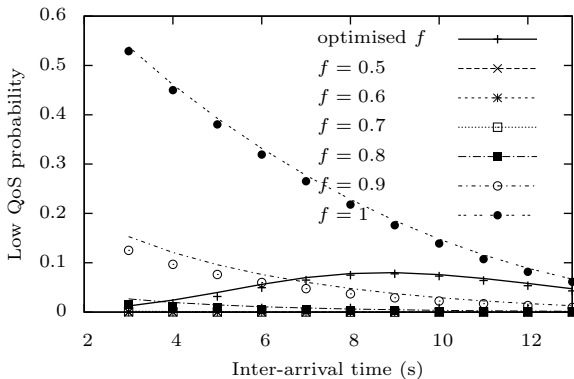

Fig. 15 Low QoS probabilities for various inter-arrival time for the AC optimisation algorithm.

The benefits of using an optimisation algorithm can also be seen in Figs. 13, 14 and 15: the blocking probability and low QoS probability are low regardless of the system parameters like the session duration or the session inter-arrival time. For instance, in Fig. 14, the low QoS probability of the adaptive algorithm stays low for high session durations in contrast to the static algorithm with $f = 0.9$ or $f = 1$ while for low session durations the blocking probability is low for both the optimisation algorithm and the static algorithm with $f = 0.9, 1$. Meanwhile, the blocking probabilty for the static algorithm for lower values of f is much higher than that of the optimisation algorithm.

7 Conclusions

In this paper an analytical model that models the time-varying cell capacity of cellular networks in general and LTE networks in particular was described. The cell capacity in these systems changes over time as the MCS that is used to send data to the users is changed depending on the signal quality. User movement causes the signal quality and thus the MCS to change over time. The time-varying cell capacity in this paper is modelled by dividing a cell in multiple concentric zones in which a certain bitrate can be achieved when sending data to the users in that zone. By assuming that the times between users changing from one zone to another are distributed according to an exponential distribution and the session duration is exponentially distributed, the system can be modelled using a CTMC. In order to verify these assumptions, results obtained with the analytical model in various scenarios were compared to simulation results that were obtained from a simulator that models the user mobility and the session duration more realistically. Results show that the analytical model captures the user mobility very accurately and that the assumption of exponentially distributed session duration is also accurate. The developed models were also used to investigate the performance of a simple AC scheme and that of an algorithm that optimises the parameter of this AC scheme. The results obtained from this study show that using an algorithm that optimises the parameters of the AC algorithm has better performance than using fixed parameters. The analytical model can further be used to study the impact of a varying cell capacity on the QoS experienced by the user and for system design issues such as resource dimensioning.

7.1 Future work

The analytical that was developed in this paper can be further extended with handovers to and from neighbouring cells. This can be done by considering transitions to and from the outermost zones coming from and going to outside the cell. The model can also be used to investigate the influence of the varying cell capacity on other AC schemes and for evaluating the ability of other AC optimisation algorithms to deal with these variations.

A Static AC algorithm

In this appendix the matrices of the system with the static AC algorithm where $N = 3$ are described. The whole analytical model for this case is described in Section 4.1. Remember that

the function $a_i(\boldsymbol{x})$ denotes whether a session that arrives in zone i when the system is in state $\boldsymbol{x}$ is accepted by the AC algorithm or not, $a_i(\boldsymbol{x}) = 1$ means that the session is accepted and $a_i(\boldsymbol{x}) = 0$ means that the session is blocked. The notation has been simplified as $a_i(\boldsymbol{x}) = a_i$. The block matrices that were not described in Section 4.1 are listed below. In these matrices $p = h + l$ and the values of δ_i equal the opposite of the sum of the other elements of the same row to make the elements of each row of the transition rate matrix $\boldsymbol{Q}$ sum to 0.

$$\boldsymbol{A_1^{h,l}} = \begin{bmatrix} \delta_0 & a_3\epsilon_3 & 0 & 0 & \cdots \\ \gamma & \delta_1 & a_3\epsilon_3 & 0 & \cdots \\ 0 & 2\gamma & \delta_2 & a_3\epsilon_3 & \cdots \\ & \ddots & \ddots & \ddots & \\ \cdots & 0 & 0 & (M-p)\gamma & \delta_{M-p} \end{bmatrix}$$

where the size of $\boldsymbol{A_1^{h,l}}$ is $(M+1-p) \times (M+1-p)$.

$$\boldsymbol{A_0^{h,l}} = \begin{bmatrix} a_2\epsilon_2 & 0 & 0 & \cdots \\ \lambda_3 & a_2\epsilon_2 & 0 & \cdots \\ 0 & 2\lambda_3 & a_2\epsilon_2 & \cdots \\ & \ddots & \ddots & \\ \cdots & 0 & 0 & (M-p)\lambda_3 \end{bmatrix}$$

where the size of $\boldsymbol{A_0^{h,l}}$ is $(M+1-p) \times (M-p)$.

$$\boldsymbol{A_2^{h,l}} = \begin{bmatrix} l\gamma & l\mu_2 & 0 & 0 & \cdots \\ 0 & l\gamma & l\mu_2 & 0 & \cdots \\ & \ddots & \ddots & \ddots & \\ \cdots & 0 & 0 & l\gamma & l\mu_2 \end{bmatrix}$$

where the size of $\boldsymbol{A_2^{h,l}}$ is $(M+1-p) \times (M+2-p)$.

$$\boldsymbol{B_1^{h,l}} = \begin{bmatrix} a_1\epsilon_1 & 0 & 0 & \cdots \\ 0 & a_1\epsilon_1 & 0 & \cdots \\ & \ddots & \ddots & \\ \cdots & 0 & 0 & a_1\epsilon_1 \\ \cdots & 0 & 0 & 0 \end{bmatrix} \qquad (16)$$

where the size of $\boldsymbol{B_1^{h,l}}$ is $(M+1-p) \times (M-p)$.

The matrix $\boldsymbol{B_2^{h,l}}$ is a diagonal matrix where the values of the diagonal are equal to $l\lambda_2$ and its size is $(M+1-p) \times (M+1-p)$.

$$\boldsymbol{C_1^{h,l}} = \begin{bmatrix} h\gamma & 0 & 0 & \cdots \\ 0 & h\gamma & 0 & \cdots \\ & \ddots & \ddots & \\ \cdots & 0 & h\gamma & 0 \end{bmatrix} \qquad (17)$$

where the size of $\boldsymbol{C_1^{h,l}}$ is $(M+1-p) \times (M+2-p)$.

The matrix $\boldsymbol{C_0^{h,l}}$ is a diagonal matrix where the values of the diagonal are equal to $h\mu_1$ and its size is $(M+1-p) \times (M+1-p)$.

B AC optimisation algorithm

When the AC optimisation algorithm described in Section 3.2 is considered, the analytical model has one more level and hence one more block level than the analytical model for the static AC algorithm with the same number of zones N, see Section 4.2. The functions $t_1(\boldsymbol{x})$ and $t_2(\boldsymbol{x})$ denote whether the parameter f must be changed or not when the system is in state $\boldsymbol{x}$ according to the AC optimisation algorithm in Eq. (2). $t_1(\boldsymbol{x}) = 1$ and $t_2(\boldsymbol{x}) = 0$ means that the parameter f must be increased, $t_1(\boldsymbol{x}) = 0$ and $t_2(\boldsymbol{x}) = 1$ means that the parameter f must be decreased and $t_1(\boldsymbol{x}) = 0$ and $t_2(\boldsymbol{x}) = 0$ means that f does not change. Note that $t_1(\boldsymbol{x})$ and $t_2(\boldsymbol{x})$ cannot be 1 at the same time. The notation has been simplified as $t_1(\boldsymbol{x}) = t_1$ and $t_2(\boldsymbol{x}) = t_2$. Remember that n_f is the number of the different discrete values that f can take and the intervals after which the optimisation is performed are exponentially distributed with mean $1/\eta$. Also remember that $p = h + m$. The block matrices for $N = 3$ that were not described in Section 4.2 are listed below. Again, the values of δ_i equal the opposite of the sum of the other elements on the same row to make the elements of each row of the transition rate matrix $\boldsymbol{Q}$ sum to 0.

$$\boldsymbol{D_1^{h,m,l}} = \begin{bmatrix} \delta_0 & t_1\eta & 0 & 0 & \cdots \\ t_2\eta & \delta_1 & t_1\eta & 0 & \cdots \\ 0 & t_2\eta & \delta_2 & t_1\eta & \cdots \\ & \ddots & \ddots & \ddots & \\ \cdots & 0 & 0 & t_2\eta & \delta_{n_f} \end{bmatrix}$$

where the size of $\boldsymbol{D_1^{h,m,l}}$ is $n_f \times n_f$. Note that $\boldsymbol{D_1^{h,m,l}}$ does not depend on the levels, i.e. it is equal for all the (h, m, l) levels.

The matrices $\boldsymbol{D_0^{h,m,l}}$, $\boldsymbol{D_2^{h,m,l}}$, $\boldsymbol{E_1^{h,m,l}}$, $\boldsymbol{E_2^{h,m,l}}$, $\boldsymbol{F_1^{h,m,l}}$ and $\boldsymbol{F_0^{h,m,l}}$ are diagonal matrices with size $n_f \times n_f$; the values on the diagonal are $a_3\epsilon_3$, $l\gamma$, $a_2\epsilon_2$, $l\lambda_3$, $m\gamma$ and $m\mu_2$ respectively.

$$\boldsymbol{B_1^{h,m}} = \begin{bmatrix} a_1\epsilon_1 & 0 & 0 & \cdots \\ 0 & a_1\epsilon_1 & 0 & \cdots \\ & \ddots & \ddots & \\ \cdots & 0 & 0 & a_1\epsilon_1 \\ \cdots & 0 & 0 & 0 \\ & \vdots & \vdots & \vdots \\ \cdots & 0 & 0 & 0 \end{bmatrix}$$

where the number of rows with all elements 0 is n_f and the size of $\boldsymbol{B_1^{h,m}}$ is $(n_f(M+1-p)) \times (n_f(M-p))$.

The Matrix $\boldsymbol{B_2^{h,m}}$ is a diagonal matrix where the values of the diagonal are $m\lambda_2$ and its size is $(n_f(M+1-p)) \times (n_f(M+1-p))$.

$$\boldsymbol{C_1^{h,m}} = \begin{bmatrix} h\gamma & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & h\gamma & 0 & 0 & 0 & \cdots & 0 \\ & \ddots & \ddots & \ddots & \ddots & \\ \cdots & 0 & h\gamma & 0 & 0 & \cdots & 0 \\ \cdots & 0 & 0 & h\gamma & 0 & \cdots & 0 \end{bmatrix}$$

where the number of columns with all elements 0 is n_f and the total size of $\boldsymbol{C_1^{h,m}}$ is $(n_f(M+1-p)) \times (n_f(M+2-p))$.

The matrix $\boldsymbol{C_0^{h,m}}$ is a diagonal matrix where the values of the diagonal are $h\mu_1$ and its size is $(n_f(M+1-p)) \times (n_f(M+1-p))$.

References

1. E3. ict-e3.eu.

2. MONOTAS. http://www.macltd.com/monotas.

3. SOCRATES. www.fp7-socrates.eu.

4. 3GPP. *3GPP TR 36.213: Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Physical layer procedures*, June 2010.

5. 3GPP. *3GPP TR 36.942: Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Radio Frequency (RF) system scenarios*, September 2010.

6. Al-Rawi, M. and Jäntti, R. Call admission control with active link protection for opportunistic wireless networks. *Telecommunication System*, 41(1):13–23, 2009.

7. Camp, T., Boleng, J., and Davies, V. A survey of mobility models for ad hoc network research. *Wireless Communications and Mobile Computing*, 2(5):483–502, 2002.

8. Duk Kyung Kim, David Griffith, and Nada Golmie. A novel ring-based performance analysis for call admission control in wireless networks. *IEEE Communications Letters*, 14(4):324–326, 2010.

9. Elayoubi, S. -E. and Chahed, T. Admission control in the downlink of WCDMA/UMTS. *Mobile and Wireless Systems*, LNCS 3427(5):136–151, 2005.

10. Garcia, D., Martinez, J., and Pla, V. Admission control policies in multiservice cellular networks: Optimum configuration and sensitivity. *Lecture Notes in Computer Science*, Wireless Systems and Mobility in Next Generation Internet, Gabriele Kotsis and Otto Spaniol (eds.)(3427):121–135, 2005.

11. Hossain, M., Hassan, M, and Sirisena, H. R. Adaptive resource management in mobile wireless networks using feedback control theory. *Telecommunication Systems*, 24(3-4):401–415, 2004.

12. Junqiang Guo, Fasheng Liu, and Zhiqiang Zhu. Estimate the call duration distribution parameters in GSM system based on K-L divergence method. In *International Conference on Wireless Communications, Networking and Mobile Computing, Shangai, China*, pages 2988 – 2991, September 2007.

13. Latouche, G. and Ramaswami, V. *Introduction to Matrix Analytic Methods in Stochastic Modeling*. ASA-SIAM, 1999.

14. Neuts, M. *Matrix-geometric Solutions in Stochastic Models: An Algorithmic Approach*. The Johns Hopkins University Press, 1981.

15. NGMN. *NGMN Radio Access Performance Evaluation Methodology*, January 2008.

16. NGMN. NGMN Recommendation on SON and O&M Requirements. December 2008.

17. Prehofer, C. and Bettstetter, C. Self-organization in communication networks: principles and design paradigms. *Communications Magazine, IEEE*, 43(7):78 – 85, 2005.

18. Ramjee, R., Nagarajan, R., and Towsley, D. On optimal call admission control in cellular networks. *Wireless Networks Journal (WINET)*, 3(1):29–41, 1997.

19. Sang Soo Jeong, Jeong Ae Han, and Wha Sook Jeon. Adaptive connection admission control scheme for high data rate mobile networks. In *Vehicular Technology Conference, 2005. VTC-2005-Fall. 2005 IEEE 62nd*, volume 4, pages 2607 – 2611, 2005.

20. Shalabh Bhatnagar and I. Bala Bhaskar Reddy. Optimal threshold policies for admission control in communication networks via discrete parameter stochastic approximation. *Telecommunication Systems*, 29(1):9–31, 2005.

21. Siwko, J. and Rubin, I. Call admission control for capacity-varying networks. *Telecommunication Systems*, 16(1-2):15–40, 2001.

22. Spaey, K., Sas, B., and Blondia, C. Self-optimising call admission control for LTE downlink. In *COST 2100 TD(10)10056, Joint Workshop COST 2100 SWG 3.1 & FP7-ICT-SOCRATES, Athens, Greece*, February 2010.

23. Spilling, A.G., Nix, A.R., Beach, M.A., and Harrold, T.J. Self-organisation in future mobile communications. *Electronics & Communication Engineering Journal*, 12 Issue:3:133–147, June 2000.