

Performance Analysis and Estimation of Call Admission Control Parameters in Wireless Integrated Voice and Data Networks

MAHMOUD T. EL-HADIDI, KHALED M. ELSAYED, AND MOHAMED M. ABDALLAH

Department of Electronics & Communications Engineering, Faculty of Engineering, Cairo University, Giza, Egypt 12613
hadidi@frcu.eun.eg, khaled@ieee.org, mabdallah@glue.umd.edu

Abstract. We propose an admission control policy for wireless multimedia networks that is based on the well known threshold-based guard channel method. The new scheme deals with two different types of traffic classes; namely: voice and data. We assume two different thresholds, one for each traffic class. In addition, we propose to buffer the handoff data calls if no free channels are available rather than rejecting them. For handoff voice calls, we propose two methods, namely: a blocking method and a preemptive method. For the blocking method, we reject the handoff voice calls if no channels are available. For the preemptive method, an ongoing data call can be buffered and its channel allocated to the handoff voice call. We study the effect of the thresholds, buffer size and the application of the proposed methods on call blocking probabilities. It is shown that the new call blocking probabilities are only affected by the threshold values. Meanwhile, the data handoff blocking probability exhibited great improvement. For handoff voice calls, when the blocking method is applied, the blocking probability value increases slightly with the increase of buffer size. Meanwhile, for the preemptive method, the handoff voice call blocking probability significantly decreases as the buffer size increases. Based on these results, we develop an algorithm that uses the proposed policy to estimate the appropriate thresholds and buffer size which meet the required call blocking probabilities for each traffic type.

1 INTRODUCTION

Future wireless personal communication will provide a lot of communication services, especially multimedia, to a large number of mobile users. These services require a backbone network that can support connections with guaranteed quality of service (QoS) requirements.

One of the main design issues in wireless multimedia networks is the implementation of call admission control. Call admission control is defined by a set of actions to determine if the call request can be accepted or rejected. Based on call admission policy function, the condition for accepting a new call request is the availability of sufficient resources to guarantee the QoS parameters without affecting the existing calls. For wireless multimedia networks based on the cellular concept, the main call-level quality of service parameters are: new call blocking probability and handoff call blocking probability [1].

Several admission policies were proposed which deal with different types of traffic classes. Tekiany *et al.* [2] illustrate the concept of prioritization of handoff calls over

new calls since it is desirable to complete an ongoing call rather than accepting a new one. They propose some channel assignment techniques to realize this concept as well as the idea of buffering handoff calls in case no free channels are available. In [3] Ramjee *et al.* present a threshold-based guard channel policy for a single traffic type. In this policy, some guard channels are allocated only to handoff calls. If the number of busy channels exceeds a given threshold, new calls are blocked. The number of guard channels depends on the required values of the new and handoff call blocking probabilities. The authors illustrate three methods to optimize a linear objective function of the new and handoff blocking probabilities. In [4] Naghshineh and Schwartz propose a distributed call admission algorithm where they dynamically vary the threshold value depending on input traffic to the cell to enhance the handoff call blocking probability in case of overloading. In [5] Acampora and Naghshineh propose a class-based admission policy which satisfies the QoS requirements for each traffic class by allocating sufficient resources to each type. They propose the capability of buffering calls which can tolerate delay if no free channels are available. None of the previous works has illus-

* An earlier version of this paper has been presented at the IEEE International Conference on Communications 1999.

trated an algorithm for estimating the thresholds to meet a certain required call blocking probabilities.

In this paper, we modify the threshold-based guard channel policy to deal with distinct types of traffic classes such as voice and data. We assume that data traffic is delay-tolerant. We add the capability for buffering the (delay-insensitive) data traffic class. We study the call admission policy under different scenarios. Later on, we present an algorithm that can be used to estimate the values of the thresholds and buffers that meet the QoS requirements.

The paper is organized as follows. Section 2 describes our proposed admission policy. The mathematical model for the wireless multimedia network system that implements the proposed policy is then explained in section 3. This is followed by a performance analysis of the proposed call admission policy in section 4. An algorithm to estimate the parameters of the proposed call admission policy is described in section 5. This is followed by a numerical example to illustrate the algorithm steps. Conclusions of the paper are given in the last section.

2 CALL ADMISSION POLICY

We consider a single cell in isolation within a cellular system. The system deals with two types of traffic: voice and data. At any time, the system is able to distinguish between the two types of traffic. During call establishment, a call is assumed to declare itself either as a voice or a data call. The call processing entities of the system (for example the processing elements in the Base-Station, the Base-Station Controller, or the Mobile Switching Center) are capable of identifying the call type at any moment.

The available resources are the maximum number of channels in the cell and the buffer capacity which is used to queue handoff calls in case no channels are available. It is assumed that the number of channels are constant due to the fact that wireless resources are limited. The buffer capacity can be changed depending on the required call blocking probabilities.

The proposed admission policy deals with two sets of parameters. Firstly, quality of service parameters, which comprise:

- New call blocking probability for voice and data calls, respectively: P_{Nvoice} and P_{Ndata}
- Handoff call blocking probability for voice and data calls, respectively: P_{Hvoice} and P_{Hdata}
- Delay experienced by handoff data calls due to buffering the handoff data call request until a channel is released: D_{Hdata}

Secondly, call admission policy control parameters, which comprise:

- New_VOICE_Threshold (Th_{Nvoice}): It determines the total number of the channels allocated to voice calls after which the new voice calls are blocked

- New_DATA_Threshold (Th_{Ndata}): It determines the total number of the channels allocated to data calls after which the new data calls are blocked
- Buffer size (B): It determines the value of the buffer size to queue the handoff data calls

These parameters control the values of the QoS parameters. In particular, Th_{Nvoice} and Th_{Ndata} control the values of the new and handoff call blocking probabilities for voice and data calls, respectively. The buffer size enhances the handoff data call blocking probability on the expense of increased delay to the call connection time.

The proposed call admission policy applies the following conditions:

- For **new voice calls**, upon request of a new channel if the number of channels allocated to voice calls is less than Th_{Nvoice} and a free channel is available, the new voice call is accepted and a free channel is assigned to the call. Otherwise, the call request is rejected. Therefore, we conclude that Th_{Nvoice} controls the value of new voice call blocking probability and handoff voice call blocking probability. As Th_{Nvoice} increases, new voice call blocking probability decreases while handoff voice call blocking probability increases.
- For **new data calls**, upon request of a new channel if the number of channels allocated to data calls is less than Th_{Ndata} and a free channel is available, the call is accepted and a free channel is assigned to the call. Otherwise, the call request is rejected. As before, as Th_{Ndata} increases, new data call blocking probability decreases while handoff data call blocking probability increases.
- For **handoff data calls**, the call request is accepted whenever there is a free available channel. Otherwise if there is no free channels, the call request is buffered until a channel is released. If the buffer is full then the handoff data call is rejected. For a given Th_{Ndata} , in order to achieve certain new data call blocking probability, buffer capacity can be increased to lower handoff data call blocking probability.
- For **handoff voice calls**, two different methods are proposed which are:
 - Blocking method:** The handoff voice call is accepted whenever there is a free available channel, i.e. the call request is rejected if the number of busy channels equals the total number of channels.
 - Preemptive method:** In case free channels are available, the handoff voice call is accepted. However, in case free channels are not available, the call is accepted under the conditions that an ongoing data call exists. A data call is selected and buffered so that the handoff voice call can utilize the released channel.

Figure 1 illustrates the admissible regions for the proposed call admission policy using blocking method.

Figure 2 illustrates a flow chart for the proposed call admission policy with two different methods; blocking and preemptive.

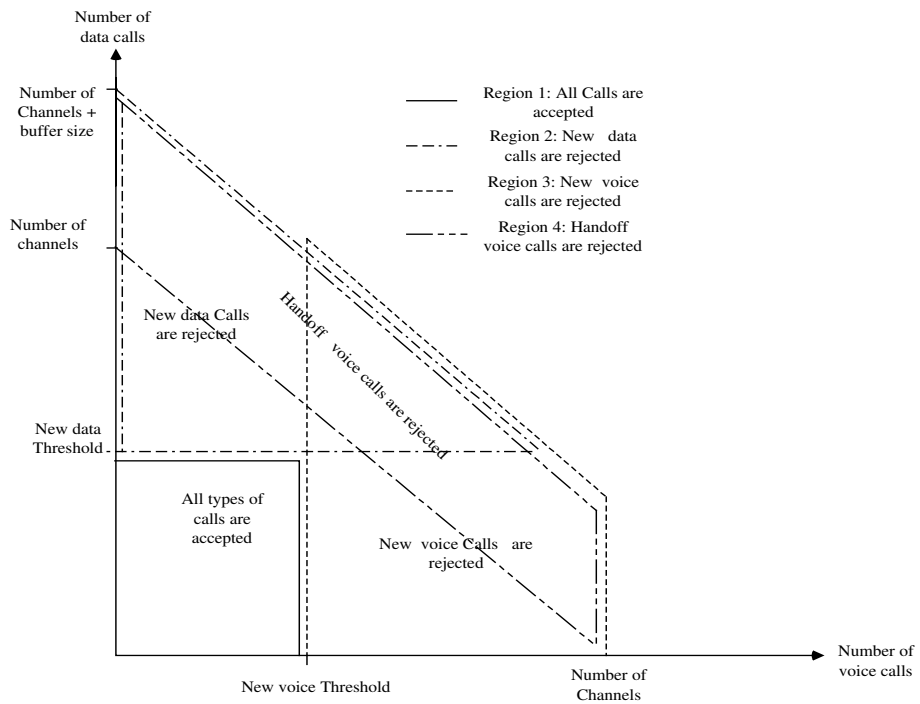


Figure 1: Admissible regions for the proposed call admission policy using blocking method.

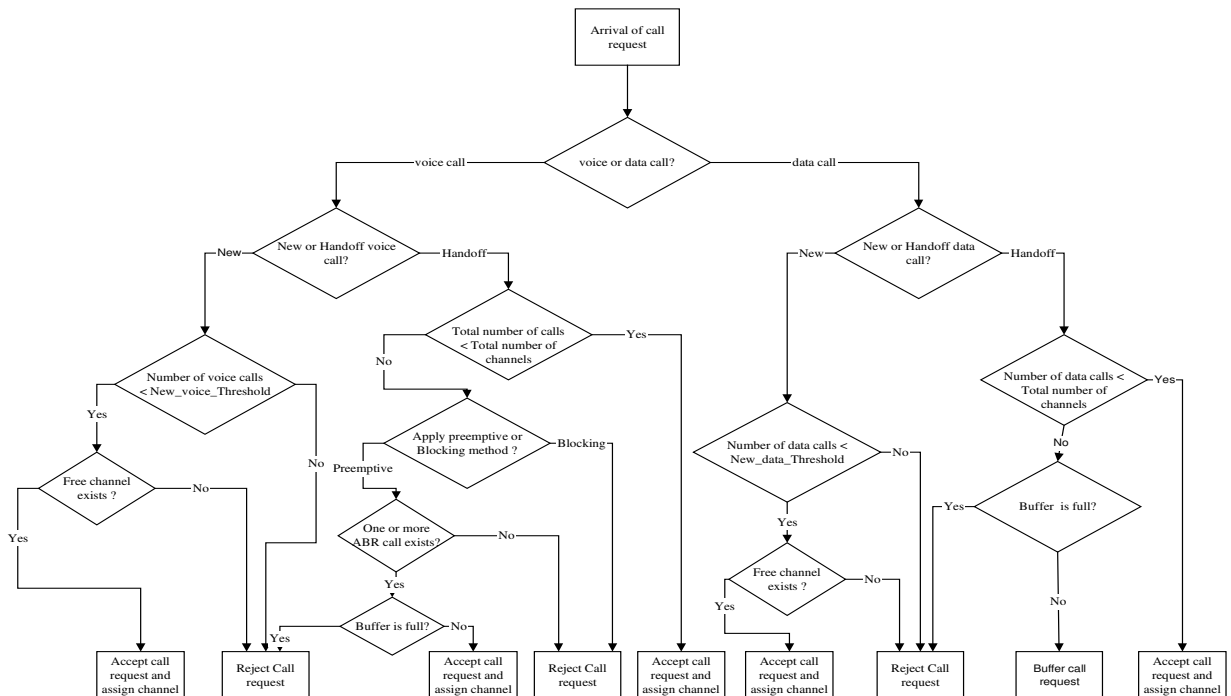


Figure 2: Flow chart for the proposed call admission policy.

3 MATHEMATICAL MODEL AND PERFORMANCE ANALYSIS

As mentioned above, the model is based on a single cell where the proposed call admission policy is applied.

We assume that the total number of channels equals C , and that the buffer is capable of buffering a number of B call requests.

It is further assumed that the arrival rates of the two input traffic types have Poisson distribution. For each of the two traffic types, voice and data, we have two types of calls, new and handoff. Therefore, we have four different arrival rates which are:

λ_{Nvoice} : Arrival rate of new voice calls.

λ_{Ndata} : Arrival rate of new data calls.

λ_{Hvoice} : Arrival rate of handoff voice calls.

λ_{Hdata} : Arrival rate of handoff data calls.

The total arrival rate for each traffic type is then:

$$\lambda_{Tvoice} = \lambda_{Nvoice} + \lambda_{Hvoice} \quad (1)$$

$$\lambda_{Tdata} = \lambda_{Ndata} + \lambda_{Hdata} \quad (2)$$

The channel holding time for each traffic type is equal and has negative exponential distribution with average service rate μ .

Since we have two types of input traffic, the system is modeled mathematically by a two-dimensional Markov chain where each state represents the number of voice and data calls in the system. Assume that the total number of calls at each state is N_{Total} , where:

$$N_{Total} = N_{voice} + N_{data} \quad (3)$$

with

N_{voice} = Number of voice calls at a certain state (N_{voice} ranges from 0 to C).

N_{data} = Number of data calls at a certain state (N_{data} ranges from 0 to $C+B$).

In Figures 3 and 4, two examples of a two-dimensional Markov chain are shown based on the proposed call admission policy using blocking and preemptive methods, respectively. The admission policy parameters have the values: $C = 5$, $B = 2$, $Th_{Nvoice} = 3$, and $Th_{Ndata} = 3$.

We now follow the standard procedure for deriving the state probabilities. In particular, we write down the steady-state equations for the state probabilities in terms of the state transition rate matrix Q and the state probability vector Π :

$$\Pi Q = 0 \quad (4)$$

$$\sum_i \pi_i = 1 \quad (5)$$

with

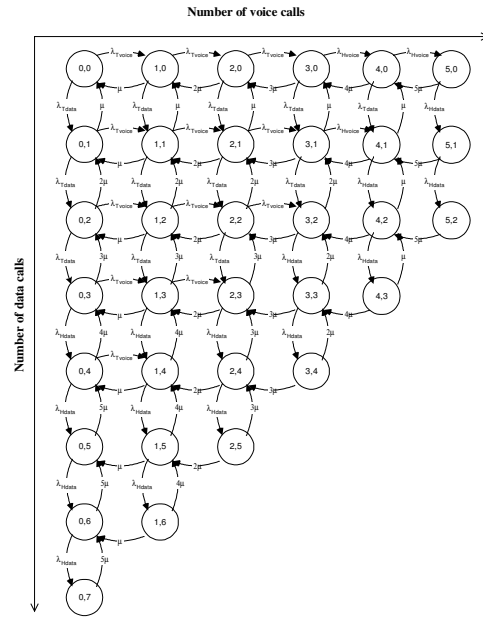


Figure 3: Two-Dimensional Markovian chain representation for a wireless network cell using the proposed call admission policy applying the blocking method ($C=5$, $B=2$, $Th_{Nvoice}=3$, $Th_{Ndata}=3$).

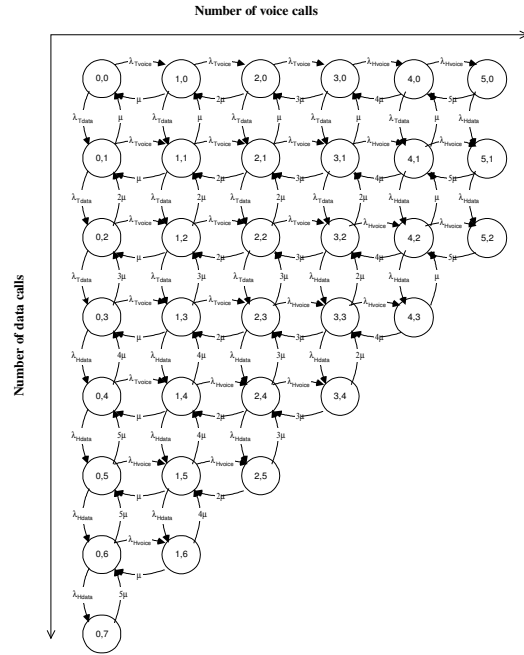


Figure 4: Two-Dimensional Markovian chain representation for a wireless network cell using the proposed call admission policy applying the preemptive method ($C=5$, $B=2$, $Th_{Nvoice}=3$, $Th_{Ndata}=3$).

$$Q^T = \begin{bmatrix} -(\lambda_{\text{voice}} + \lambda_{\text{data}}) & \mu & 0 & \dots & 0 & \mu & 0 & \dots & 0 \\ \lambda_{\text{data}} & -(\lambda_{\text{voice}} + \lambda_{\text{data}} + \mu) & 2\mu & 0 & \dots & 0 & 2\mu & 0 & \dots & 0 \\ 0 & \lambda_{\text{data}} & -(\lambda_{\text{voice}} + \lambda_{\text{data}} + 2\mu) & 3\mu & \dots & 0 & 3\mu & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

$$\Pi = \left[\pi_{0,0} \quad \pi_{0,1} \quad \pi_{0,2} \quad \dots \quad \pi_{N_{\text{voice}}, N_{\text{data}}} \quad \dots \right]$$

A program is written in C to generate these equations. Since the probability transition matrix represents a sparse matrix, the above equations can be solved using sparse 1.3 package [6], which calculates the state probability vector.

After obtaining the state probabilities, the values of the new blocking probability, handoff blocking probability and average delay time are calculated using the following relations. These relations are identical in the case of blocking method and preemptive method except for the calculation of handoff voice blocking probability.

For the new voice call blocking probability, $P_{N_{\text{voice}}}$:

$$P_{N_{\text{voice}}} = \left(\sum_{N_{\text{voice}}=Th_{N_{\text{voice}}}}^C \sum_{N_{\text{data}}=0}^{C+B-N_{\text{voice}}} \pi_{N_{\text{voice}}, N_{\text{data}}} \right) \frac{\lambda_{N_{\text{voice}}}}{\lambda_{\text{voice}} + \lambda_{\text{data}}} \quad (6)$$

For the new data call blocking probability, $P_{N_{\text{data}}}$:

$$P_{N_{\text{data}}} = \left(\sum_{N_{\text{voice}}=0}^C \sum_{N_{\text{data}}=Th_{N_{\text{data}}}}^{C+B-N_{\text{voice}}} \pi_{N_{\text{voice}}, N_{\text{data}}} \right) \frac{\lambda_{N_{\text{data}}}}{\lambda_{\text{voice}} + \lambda_{\text{data}}} \quad (7)$$

For the handoff voice call blocking probability, $P_{H_{\text{voice}}}$:

$$P_{H_{\text{voice}}} = \left(\sum_{N_{\text{voice}}=0}^C \sum_{N_{\text{data}}=C-N_{\text{voice}}}^{C+B-N_{\text{voice}}} \pi_{N_{\text{voice}}, N_{\text{data}}} \right) \frac{\lambda_{H_{\text{voice}}}}{\lambda_{\text{voice}} + \lambda_{\text{data}}} \quad (8)$$

using the blocking method, and

$$P_{H_{\text{voice}}} = \left(\sum_{N_{\text{voice}}=0}^C \pi_{N_{\text{voice}}, C+B-N_{\text{voice}}} \right) \frac{\lambda_{H_{\text{voice}}}}{\lambda_{\text{voice}} + \lambda_{\text{data}}} \quad (9)$$

using preemptive method.

For the handoff data call blocking probability, $P_{H_{\text{data}}}$:

$$P_{H_{\text{data}}} = \left(\sum_{N_{\text{voice}}=0}^C \pi_{N_{\text{voice}}, C+B-N_{\text{voice}}} \right) \frac{\lambda_{H_{\text{data}}}}{\lambda_{\text{voice}} + \lambda_{\text{data}}} \quad (10)$$

Average delay time for handoff data calls, $D_{H_{\text{data}}}$:

$$D_{H_{\text{data}}} = \frac{\sum_{N_{\text{voice}}=0}^C \sum_{N_{\text{data}}=C+1-N_{\text{voice}}}^{C+B-N_{\text{voice}}} (\pi_{N_{\text{voice}}, N_{\text{data}}} \cdot (N_{\text{voice}} + N_{\text{data}}))}{\lambda_{H_{\text{data}}} (1 - P_{H_{\text{data}}})} \quad (11)$$

4 NUMERICAL RESULTS FOR THE PROPOSED CALL ADMISSION POLICY

To illustrate the effect of the threshold values and the buffer size on $P_{N_{\text{voice}}}$, $P_{N_{\text{data}}}$, $P_{H_{\text{voice}}}$, $P_{H_{\text{data}}}$ and $D_{H_{\text{data}}}$, we applied Eqs. (6) - (11) to a numerical example where we assigned the following values for the policy parameters :

Total number of channels, $C = 25$.

Average channel holding time, $1/\mu = 3$ min.

Offered load for voice calls per channel:

$$\frac{\lambda_{\text{voice}}}{C\mu} = 0.4 \text{ Erlang/channel.}$$

Offered load for data calls per channel:

$$\frac{\lambda_{\text{data}}}{C\mu} = 0.4 \text{ Erlang/channel.}$$

Handoff offered load for voice calls per channel:

$$\frac{\lambda_{H_{\text{voice}}}}{C\mu} = 0.2 \text{ Erlang/channel.}$$

Handoff offered load for data calls per channel:

$$\frac{\lambda_{H_{\text{data}}}}{C\mu} = 0.2 \text{ Erlang/channel.}$$

Maximum buffer size, $B = 5$.

Remark: The seemingly unrealistic values for handoff traffic were intentionally chosen to study the behavior of the proposed buffering scheme under heavy handoff traffic conditions.

4.1 RESULTS FOR ADMISSION POLICY WITH BLOCKING

We will first study the effect of the control parameters which are the thresholds and buffer capacity on the QoS parameters in case of applying the proposed call admission policy using blocking method.

Figure 5 illustrates the effect of varying $Th_{N_{\text{voice}}}$ on $P_{N_{\text{voice}}}$, for a constant $Th_{N_{\text{data}}} (=10)$ and different values of B . It is clear that there is no effect of buffering on $P_{N_{\text{voice}}}$ except for high values of the threshold, where we can observe a slight increase in $P_{N_{\text{voice}}}$. However, such high $Th_{N_{\text{voice}}}$ values are not practical since giving preference to handoff voice calls over new voice calls necessitates using smaller values. Furthermore, it can be observed that $P_{N_{\text{voice}}}$ becomes constant for the values of $Th_{N_{\text{voice}}}$ where the sum of the two thresholds ($Th_{N_{\text{voice}}}$ and $Th_{N_{\text{data}}}$) exceeds the number of available channels ($C = 25$). This can be easily observed in Figure 5 where $P_{N_{\text{voice}}}$ becomes constant for the values of $Th_{N_{\text{voice}}}$ which exceeds 15. This can also be seen from Figure 6 where $P_{N_{\text{voice}}}$ is recalculated for a constant $Th_{N_{\text{data}}} (= 5)$. It is clear that $P_{N_{\text{voice}}}$ remains constant for values of $Th_{N_{\text{data}}}$ which exceeds 20.

The effect of varying Th_{Nvoice} on P_{Ndata} - under the same previous conditions - is shown in Figure 7. It is evident that varying Th_{Nvoice} has no effect on P_{Ndata} and we can conclude that P_{Ndata} depends only on the value of Th_{Ndata} . Also the effect of buffering on P_{Ndata} is negligible (similar to the effect of buffering on P_{Nvoice}).

The effect of Th_{Nvoice} on P_{Hvoice} is shown in Figure 8. As expected, P_{Hvoice} is degraded with larger Th_{Nvoice} but it becomes constant when Th_{Nvoice} equals 15. It is also clear that P_{Hvoice} increases slightly when a buffer is added to the system, but it remains constant as the buffer size increases. Therefore, we can neglect the effect of buffering on P_{Hvoice} .

Finally, the two graphs shown in Figure 9 and 10 illustrate the effect of buffering data handoff calls on its blocking probability and average delay respectively. It is evident that by buffering, P_{Hdata} is enhanced considerably. The larger the buffer size, the smaller P_{Hdata} becomes. However, this enhancement is on the expense of increasing the average delay time added to the call connection time of handoff data calls. Also, it can be seen that the average delay time depends mainly on the threshold value rather than the buffer size. Finally, it can be deduced that although P_{Ndata} does not depend on Th_{Nvoice} , P_{Hdata} is affected by varying the value of Th_{Nvoice} which is increased as Th_{Nvoice} increases.

Figures 11 to 14 illustrate the simultaneous effect of the two thresholds - Th_{Nvoice} and Th_{Ndata} - on QoS parameters for two different values of buffer size - 1 and 5 - using blocking method. Based on these figures, the minimum values for each QoS parameter can be deduced as follows:

For P_{Nvoice} and P_{Ndata} , it is clear that threshold values that minimize each of them occur at:

$$P_{Nvoice\min} \equiv \min P_{Nvoice}(Th_{Nvoice}, Th_{Ndata}, B) = P_{Nvoice}(C, 0, B) \quad (12)$$

$$P_{Ndata\min} \equiv \min P_{Ndata}(Th_{Nvoice}, Th_{Ndata}, B) = P_{Ndata}(0, C, B) \quad (13)$$

It is to be noticed that minimization is done with respect to Th_{Nvoice} and Th_{Ndata} since P_{Nvoice} and P_{Ndata} are independent of the buffer size B .

Considering handoff blocking probabilities on the other hand, one observes that the minimum values of P_{Hvoice} and P_{Hdata} depend on both thresholds and the buffer size. Therefore, it is clear from Figures 13 and 14 that minimum values occur at:

$$P_{Hvoice\min} \equiv \min P_{Hvoice}(Th_{Nvoice}, Th_{Ndata}, B) = P_{Hvoice}(0, 0, 0) \quad (14)$$

$$P_{Hdata\min} = \min P_{Hdata}(Th_{Nvoice}, Th_{Ndata}, B) = P_{Hdata}(0, 0, \infty) = 0 \quad (15)$$

It is clear that the minimum value for P_{Hdata} is zero for infinite buffer size; however, this is realized on the expense of infinite delay value.

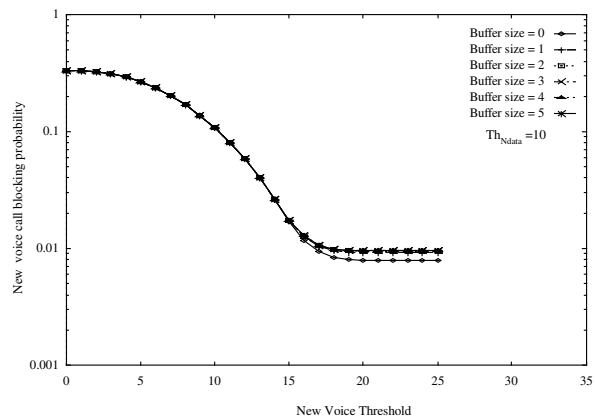


Figure 5: P_{Nvoice} versus Th_{Nvoice} for different values of buffer size B using blocking method - $Th_{Ndata} = 10$.

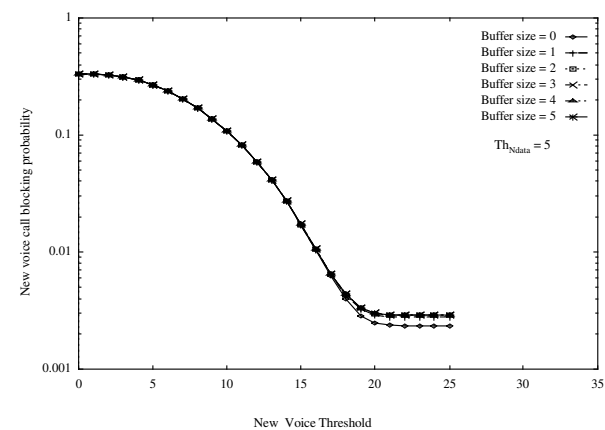


Figure 6: P_{Nvoice} versus Th_{Nvoice} for different values of buffer size B using blocking method - $Th_{Ndata} = 5$.

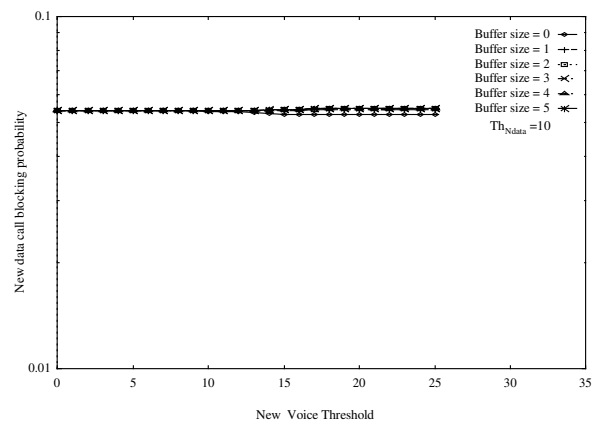


Figure 7: P_{Ndata} versus Th_{Nvoice} for different values of buffer size B using blocking method - $Th_{Ndata} = 10$.

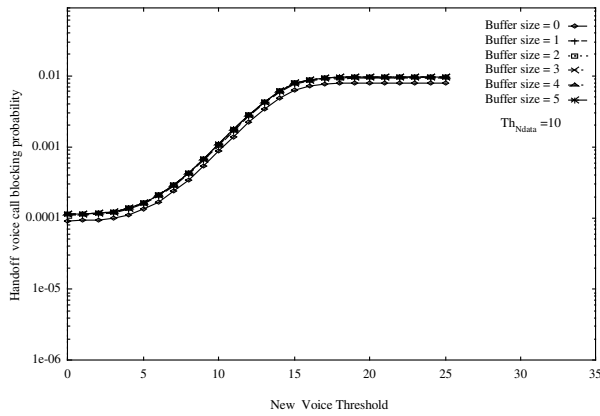


Figure 8: P_{Hvoice} versus Th_{Nvoice} for different values of buffer size B using blocking method - $Th_{Ndata} = 10$.

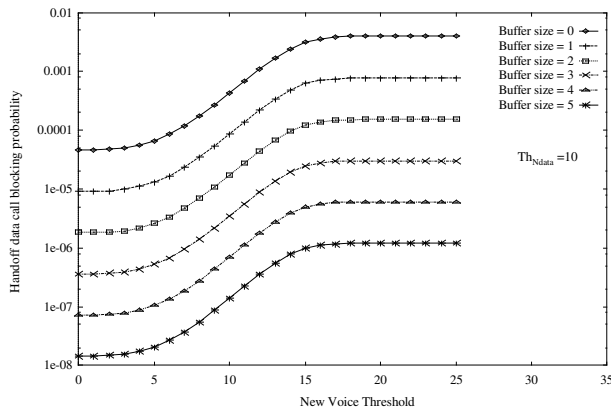


Figure 9: P_{Hdata} versus Th_{Nvoice} for different values of buffer size B using blocking method - $Th_{Ndata} = 10$.

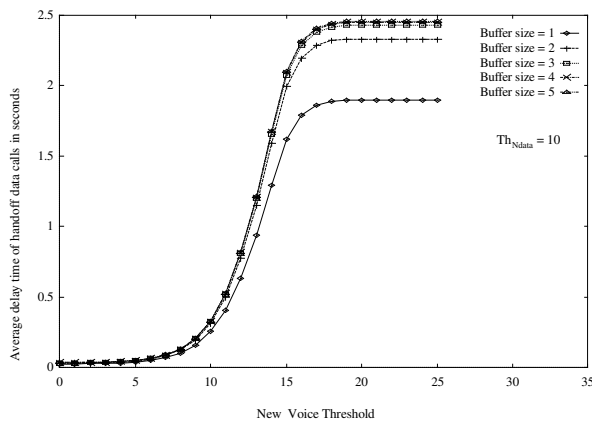


Figure 10: D_{Hdata} versus Th_{Nvoice} for different values of buffer size B using blocking method.

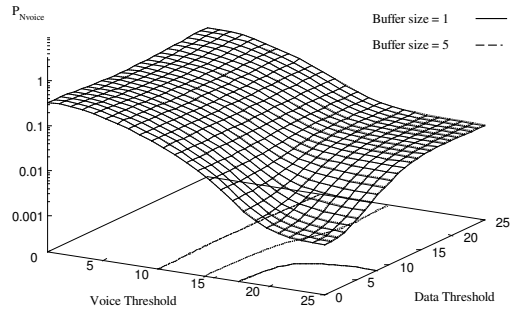


Figure 11: Simultaneous effect of two thresholds - Th_{Nvoice} and Th_{Ndata} - on P_{Nvoice} using the blocking method.

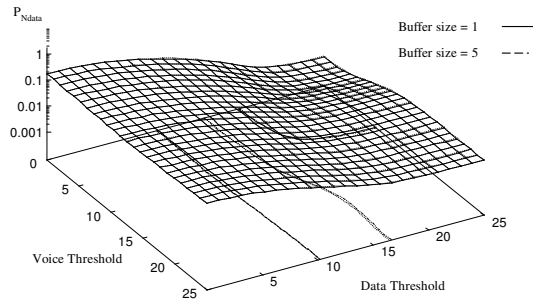


Figure 12: Simultaneous effect of two thresholds - Th_{Nvoice} and Th_{Ndata} - on P_{Ndata} using blocking method.

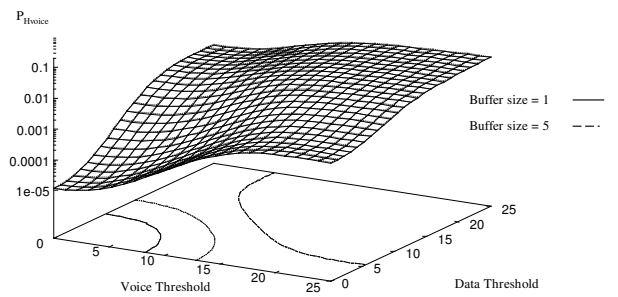


Figure 13: Simultaneous effect of two thresholds - Th_{Nvoice} and Th_{Ndata} - on P_{Hvoice} using the blocking method.

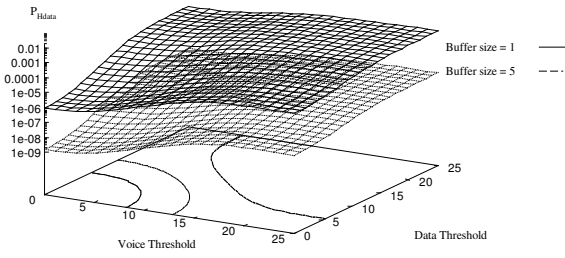


Figure 14: Simultaneous effect of two thresholds; Th_{Nvoice} and Th_{Ndata} on P_{Hdata} using the blocking method.

4.2 RESULTS FOR ADMISSION POLICY WITH PREEMPTION

We next study the effect of control parameters on QoS parameters for the proposed call admission policy using preemptive method under the same previous traffic conditions.

It is clear by comparing Figures 5 and 15, which illustrate the effect of thresholds and buffer size on P_{Nvoice} for both methods - blocking and preemptive, that both methods have the same effect on P_{Nvoice} with respect to behavior and value. Similarly, P_{Ndata} does not depend on the method used as shown in Figures 6 and 16.

However, as mentioned before, the preemptive method affects P_{Hvoice} as it depends on the buffer size. As the buffer size increases, P_{Hvoice} decreases as illustrated in Figure 17. This is realized on the expense of degrading P_{Hdata} values as can be deduced from Figures 9 and 18. For $Th_{Nvoice} = 0$ and $B = 5$, $P_{Hdata} \cong 1e-08$ in the blocking method and $P_{Hdata} \cong 3e-07$ in the preemptive method. Also, the delay values increase when using the preemptive method. This can be deduced by comparing Figures 10 and 19.

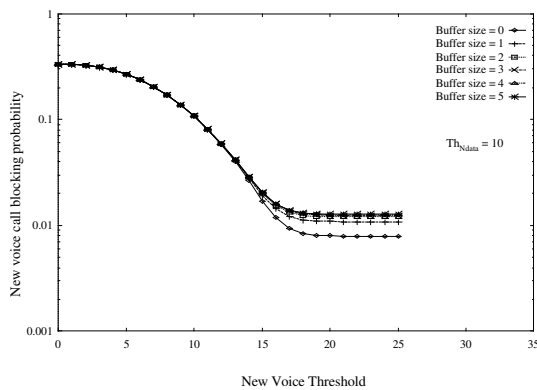


Figure 15: P_{Nvoice} versus Th_{Nvoice} for different values of buffer size B using the preemptive method.

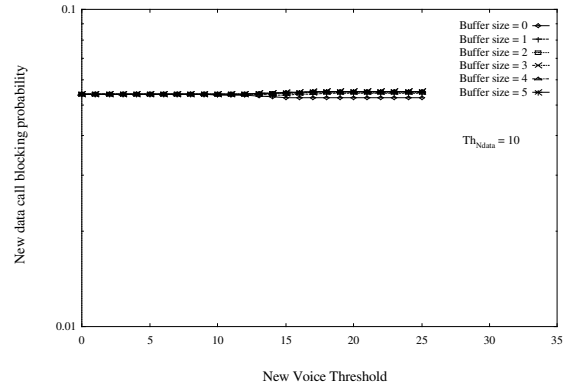


Figure 16: P_{Ndata} versus Th_{Nvoice} for different values of buffer size B using the preemptive method.

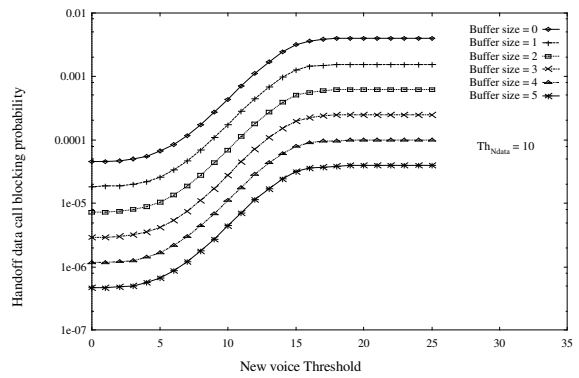


Figure 17: P_{Hvoice} versus Th_{Nvoice} for different values of buffer size B using the preemptive method.

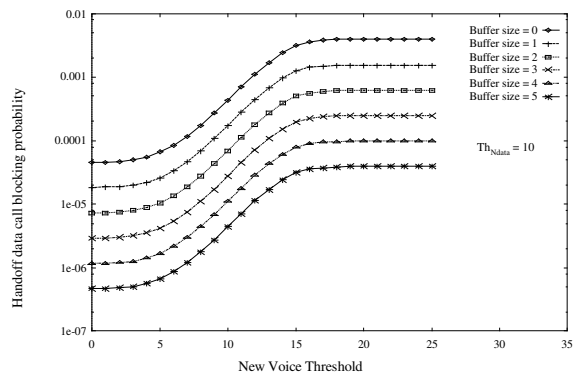


Figure 18: P_{Hdata} versus Th_{Nvoice} for different values of buffer size B using the preemptive method.

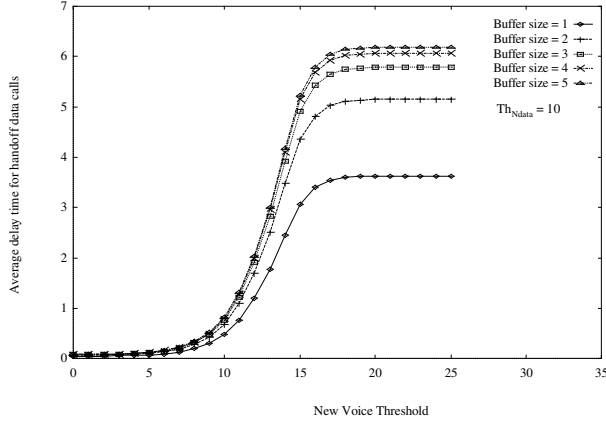


Figure 19: D_{Hdata} versus Th_{Nvoice} for different values of buffer size B using the preemptive method.

Figures 20 to 23 illustrate the simultaneous effect of the two thresholds for two different buffer values using the preemptive method. It can be easily seen from Figures 20 and 21 that the minimum values for P_{Nvoice} and P_{Ndata} do not depend on the buffer size and that their values occur at the same threshold values obtained in the blocking method.

Meanwhile, one observes from Figure 22 that the minimum value of P_{Hvoice} changed due to the application of preemptive method because it now depends on the buffer size. Therefore, the threshold values and the buffer size that minimize P_{Hvoice} are:

$$\begin{aligned} P_{Hvoice\min} &\equiv \min P_{Hvoice}(Th_{Nvoice}, Th_{Ndata}, B) \\ &= P_{Hvoice}(0, 0, \infty) = 0 \end{aligned} \quad (16)$$

Therefore, the minimum value of P_{Hvoice} depends on which method is used, blocking or preemptive. More specifically, the absolute minimum value of zero will be realized if the preemptive method is used along with an infinite buffer.

It is also clear from Figure 23 that the minimum value for P_{Hdata} will not change and it will occur at the same thresholds and buffer size values as in equation (15).

Finally, it is to be noted that contour plots are shown in the 3-dimensional surfaces, which are depicted at different blocking probability values. Each contour is specifically function of the following:

Contour \equiv K(Type of blocking probability, Th_{Nvoice} , Th_{Ndata} , B , Method type {Blocking or Preemptive}).

These contour plots will be utilized later in the paper.

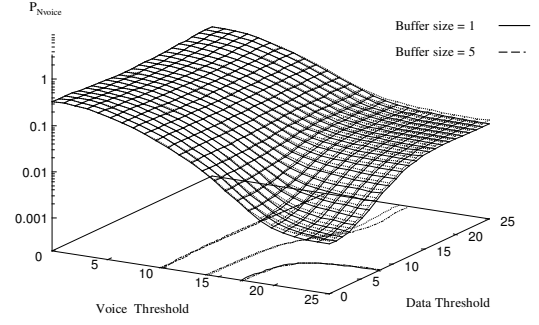


Figure 20: Simultaneous effect of two thresholds - Th_{Nvoice} and Th_{Ndata} - on P_{Nvoice} using the preemptive method.

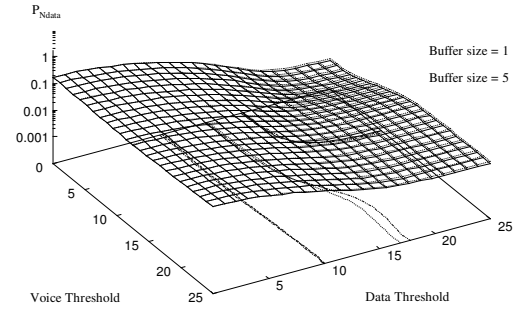


Fig. 21: Simultaneous effect of two thresholds - Th_{Nvoice} and Th_{Ndata} - on P_{Ndata} using the preemptive method.

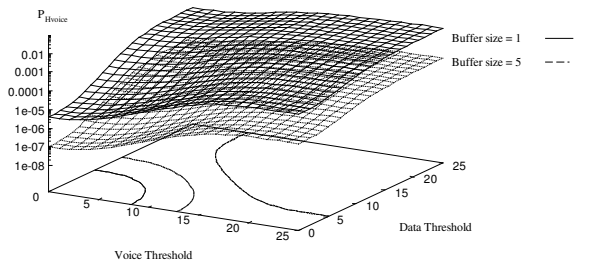


Fig. 22: Simultaneous effect of two thresholds - Th_{Nvoice} and Th_{Ndata} - on P_{Hvoice} using the preemptive method.

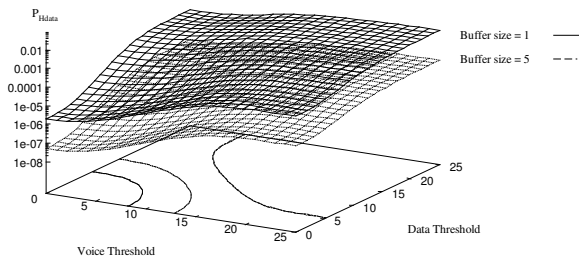


Figure 23: Simultaneous effect of two thresholds - Th_{Nvoice} and Th_{Ndata} - on P_{Hdata} using the blocking method.

5 ALGORITHM FOR ESTIMATING PARAMETERS OF THE PROPOSED CALL ADMISSION POLICY

In this section, an algorithm is proposed to evaluate the control parameters for the proposed call admission policy for given QoS requirements. Due to the limited number of available control parameters; namely : Th_{Nvoice} , Th_{Ndata} and buffer size B, the number of QoS parameters which can be guaranteed using the proposed call admission policy are also limited. We select the following set as the controllable QoS parameters:

- Maximum new call blocking probability, P_{NMAX} : It is the maximum new call blocking probability guaranteed for the two types of traffic classes i.e. $P_{Nvoice} \leq P_{NMAX}$ and $P_{Ndata} \leq P_{NMAX}$.
- Maximum handoff call blocking probability, P_{HMAX} : It is the maximum handoff call blocking probability guaranteed for the two types of traffic classes, i.e. $P_{Hvoice} \leq P_{HMAX}$ and $P_{Hdata} \leq P_{HMAX}$.

Meanwhile, other QoS parameters (delay or utilization) may be minimized or maximized as much as possible though they cannot be guaranteed to be below/above a certain value.

In addition to the QoS parameters, the number of channels C will be given as an input to the parameter estimation algorithm. The solution is obtained by depicting the contour plot for each QoS parameter at the required value and estimating the region of intersection for the contour plots which will be realized at a certain buffer size. The region of intersection will define the feasible region of thresholds for Th_{Nvoice} and Th_{Ndata} .

The algorithm steps will be described and illustrated through a numerical example which is based on the same traffic conditions and number of channels as specified in section V. The values for P_{NMAX} and P_{HMAX} are set, respectively, at 0.1 and $5e-05$. (These values are specifically chosen to illustrate the concepts not to reflect values used in actual design of cellular networks).

Step 1: Verification

The objective of this step is to verify that the specified number of channels can satisfy the specified QoS parameters. It must be justified that the maximum specified blocking probability for each traffic type is higher than the minimum value for each QoS parameter. Therefore the following relations must be satisfied:

- $P_{Nvoicemin}$ and $P_{Ndatamin}$ are lower than P_{NMAX} where $P_{Nvoicemin}$ and $P_{Ndatamin}$ are the minimum values for new voice and data call blocking probabilities which are defined before in relations (13) and (14).
- $P_{Hvoicemin}$ and $P_{Hdatamin}$ are lower than P_{HMAX} where $P_{Hvoicemin}$ and $P_{Hdatamin}$ are the minimum value of handoff voice and data call blocking probability. As illustrated before in relations (15) and (16), these probabilities can theoretically reach zero.

Based on the numerical example are, these values are given by: $P_{Nvoicemin} = 0.00208512$, $P_{Ndatamin} = 0.00104256$, $P_{Hvoicemin} = P_{Hdatamin} = 0$.

It is clear that these values are lower than the specified QoS parameters. If this condition is not satisfied, then the number of channels must be increased (unfeasible solution).

Step 2: Finding the Region Satisfying New Call Blocking Probability

In this step, the regions for the two thresholds which satisfy the condition of P_{Nvoice} and P_{Ndata} lower than P_{NMAX} are identified. Based on the 3-dimensional curves of Figures 12 and 13, the contour plots are depicted at the specified value of P_{NMAX} . The corresponding regions of Th_{Nvoice} and Th_{Ndata} are shown in Figure 24. Thus, the region, which will be named Region '1', is bounded by the following contours and lines:

Contour 1: $K_{Nvoice} = K(P_{Nvoice}, Th_{Nvoice}, Th_{Ndata})$.

Contour 2: $K_{Ndata} = K(P_{Ndata}, Th_{Nvoice}, Th_{Ndata})$.

Line 1: $Th_{Nvoice} = 25$.

Line 2: $Th_{Ndata} = 25$.

It is to be remarked that P_{Nvoice} and P_{Ndata} are independent of both the buffer size and whether blocking or preemptive is used.

Step 3: Finding the Region Satisfying Voice Handoff Call Blocking Probability

The objective of this step is to define the region of thresholds, buffer capacity and method (blocking or preemptive) in order to satisfy the following two conditions:

- $P_{Hvoice} < P_{HMAX}$.
- The region of thresholds obtained must intersect with region '1' defined in step 2 which guarantees the values of new call blocking probabilities.

Firstly, we try to use the blocking method since it introduces lower delay values (for the data calls) than the preemptive method. Contour plots at the specified P_{HMAX} value using the 3-dimensional curve in Figure 14 are redrawn in the 2-dimensional plane in Figure 25. The buffer size selected equals zero since it gives the minimum value

for P_{Hvoice} as defined in relation (15). The resulting region is bounded by the following contours and lines:

- Contour: $K_{Hvoice} = K(P_{Hvoice}, Th_{Nvoice}, Th_{Ndata}, B, \text{blocking method})$ where buffer size equals zero.
- Line 1: $Th_{Nvoice} = 0$.
- Line 2: $Th_{Ndata} = 0$.

It is clear in Figure 25 that for our given example there is no intersection between the obtained shaded region and region '1'. Therefore, the blocking method cannot satisfy the required QoS value and the preemptive method must be used.

Contour plots are depicted for buffer size = 1 while preemptive method is applied. It is clear that there is still no intersection as shown in Figure 26. Therefore, the buffer size is increased until intersection occurs (at $B = 5$) as shown in Figure 27 and the region obtained (named region '2') will be bounded by the following contours and lines:

- Contour: $K_{Hvoice} = K(P_{Hvoice}, Th_{Nvoice}, Th_{Ndata}, B, \text{preemptive method})$ where buffer size equals 5 in our example.
- Line 1: $Th_{Nvoice} = 0$.
- Line 2: $Th_{Ndata} = 0$.

Finally, our solution will be the intersection of region '1' and region '2' which will result in a new region; named: region '3'. Region '3' will be bounded by contours and lines based on our example:

- Contour 1: $K_{Nvoice} = K(P_{Nvoice}, Th_{Nvoice}, Th_{Ndata})$.
- Contour 2: $K_{Ndata} = K(P_{Ndata}, Th_{Nvoice}, Th_{Ndata})$.
- Contour3: $K_{Hvoice} = K(P_{Hvoice}, Th_{Nvoice}, Th_{Ndata}, B, \text{preemptive method})$ where buffer size equals 5 in our example.
- Line: $Th_{Nvoice} = 25$.

Step 4: Finding the Region Satisfying Data Handoff Call Blocking Probability

The objective of this step is to update the values of thresholds and buffer size to:

- specify a region where $P_{Hdata} < P_{HMAX}$.
- guarantee that an intersection exists between that region and the previously specified region in step 3.

The method (blocking or preemption) has been specified in step 3, so the contour plot are depicted at the specified P_{HMAX} using the 3-dimensional curves for either blocking or preemptive method as shown in Figures 14 and 23. If the obtained region intersects with region (3) then region (3) and the buffer size selected is the solution to guarantee the required QoS values. If no intersection exists, the buffer size is increased until intersection exists. In our example, the region; named: region '4' obtained to guarantee P_{Hdata} is lower than P_{HMAX} is defined by the following contours and lines:

- Contour: $K_{Hdata} = K(P_{Hdata}, Th_{Nvoice}, Th_{Ndata}, B, \text{preemptive method})$ where buffer size equals 5 in our example.
- Line 1: $Th_{Nvoice} = 0$.

- Line 2: $Th_{Ndata} = 0$.

It is clear that region '3' is a subset of region '4' shown in Figure 28, therefore region '3' and the buffer size selected which equals 5 in our example is the desired solution.

A flow chart for the estimation algorithm of call admission control parameters is shown in Figure 29.

After obtaining the range of thresholds, we need to obtain the optimum values of thresholds. We can apply any one of the following criteria to obtain the optimum values.

Minimum new call blocking probability: If it is desired to obtain the lowest possible minimum new call blocking probability for each traffic type, the maximum value of the range of each threshold in region '3' is selected. However, this will be on the expense of increasing delay time.

Minimum delay: In order to obtain the minimum possible delay needed, the minimum values of thresholds should be selected based on Figure 30 which shows that the delay decreases as the values of thresholds decrease.

Maximum utilization: Another parameter which can be considered is the utilization of voice calls and data calls. The relations which calculate the utilization are as follows:

$$\text{Voice_Util} = \frac{\text{Average number of voice calls in the system}}{\text{Total number of channels}} \quad (17)$$

$$\text{DATA_Util} = \frac{\text{Average number of data calls in the system}}{\text{Total number of channels}} \quad (18)$$

Figures 31 and 32 illustrate the utilization of voice and data calls under the same previous traffic conditions while applying the two methods; blocking and preemptive. It is clear that utilization increases as thresholds increases so maximum utilization is achieved for maximum values of thresholds. It can be deduced from Figure 32 that the two surfaces representing the utilization of voice and data calls are coincident which means that their values are nearly equal if preemptive method is applied.

Consequently, the values of thresholds are determined based on which parameter is to be selected.

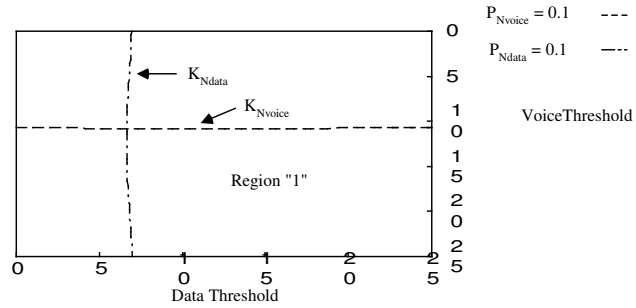


Figure 24: Regions of Th_{Ndata} and Th_{Nvoice} that guarantee P_{Nvoice} and $P_{Ndata} < P_{NMAX}$.

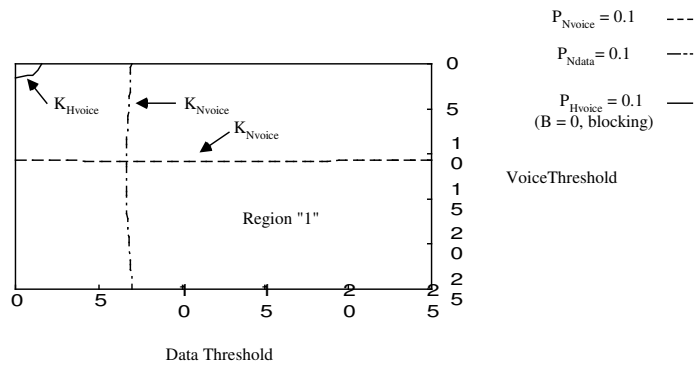


Figure 25: Illustrating that there is no intersection between region '1' and shaded region using the blocking method that guarantee QoS parameters (P_{Nvoice} , P_{Ndata} , P_{Hvoice}).

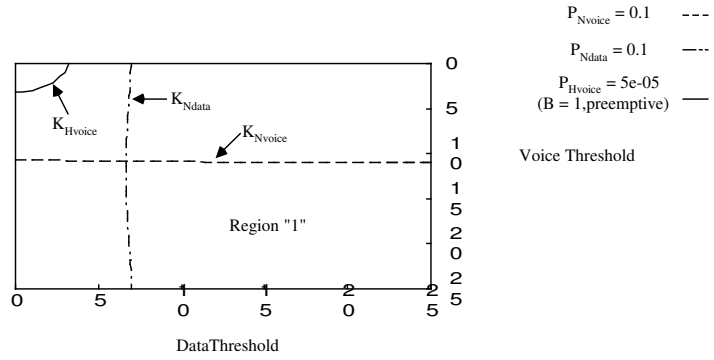


Fig. 26: Illustrating that there is no intersection between region '1' and shaded region using the preemptive method at buffer size = 1 that guarantee QoS parameters (P_{Nvoice} , P_{Ndata} , P_{Hvoice}).

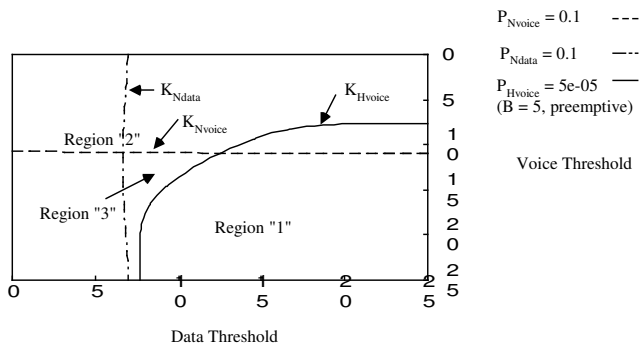


Figure 27: Illustrating the regions for Th_{Nvoice} and Th_{Ndata} using preemptive method at buffer size = 5 that guarantee QoS parameters (P_{Nvoice} , P_{Ndata} , P_{Hvoice}).

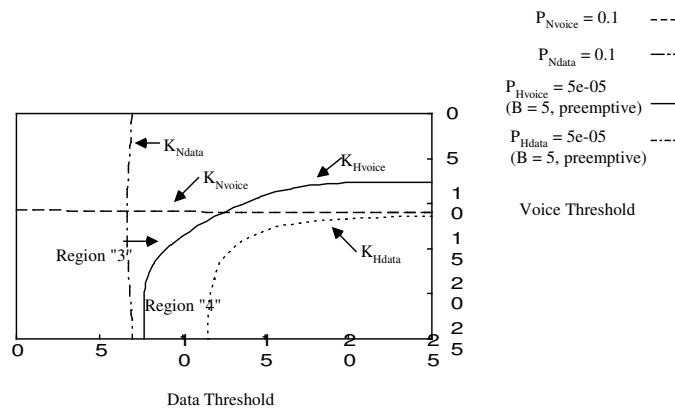


Figure 28: Illustrating the common regions for Th_{Nvoice} and Th_{Ndata} using the preemptive method at buffer size = 5 that guarantee QoS parameters (P_{Nvoice} , P_{Ndata} , P_{Hvoice} , P_{Hdata}).

6 CONCLUSIONS

In this paper, we proposed a new call admission policy for wireless multimedia networks with two different types of classes; namely: voice traffic class with low delay bounds and data traffic class with higher delay bounds. This policy is based on modifying the guard channel method and introducing a buffer for queuing handoff data calls. We select to buffer handoff data calls because they can tolerate delay and because it is desirable to accept an ongoing call. We studied the effect of buffering on the QoS parameters for each traffic class. Our study leads to the following conclusions:

- The value of new and handoff voice call blocking probability (P_{Nvoice} , P_{Hvoice}) do not depend on the buffer size. They are only controlled by Th_{Nvoice} . Also the same result is applicable to new data call blocking probability which depends on Th_{Ndata} .
- The value of handoff voice call blocking probability can be enhanced by applying the concept of preemption of an existing data call in case of no channels available to serve the handoff voice call rather than blocking it. At the same time, preemption will not degrade the performance of new call blocking probabilities.
- The value of handoff data call blocking probability (P_{Hdata}) can be enhanced by increasing the buffer size. However, this increases the delay added to the call connection time.

Finally, based on these features, we proposed a parameter estimation algorithm which can estimate the values of thresholds and buffer size based on a given QoS parameters. Since, in general, one obtains a region of feasible solutions, these control parameters can be further optimized by specifying an additional performance index

such as delay or utilization. These concepts have been demonstrated by means of a numerical example.

We have experimented with other cases (which are not reported in this paper for space limitations) and the general conclusions presented here, also applies to them. It is to be noted that the work in this paper is not intended for real-time implementation. However, the knowledge gained here can be utilized to design an efficient real-time admission control policy for integrated-services wireless networks.

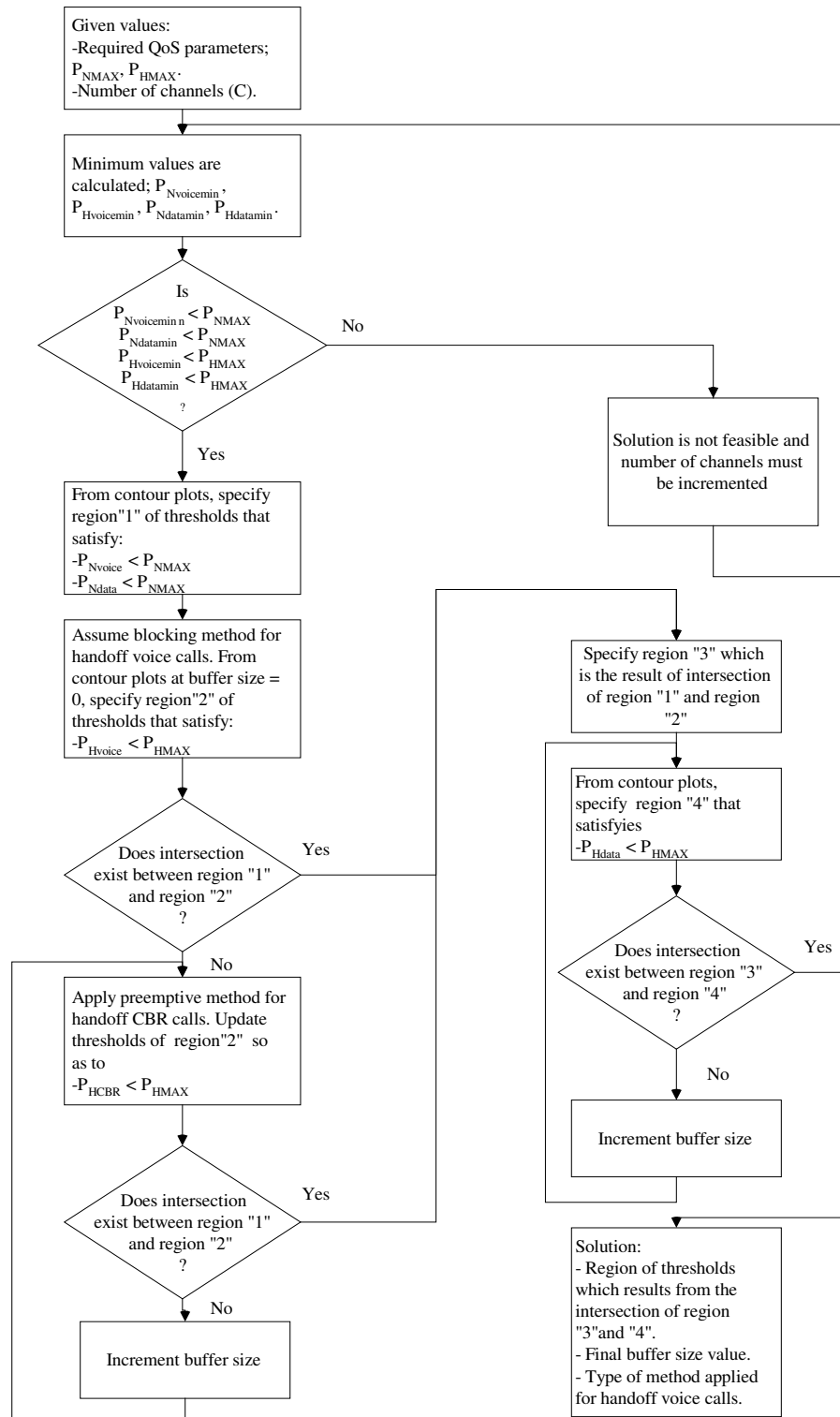


Figure 29: Flow chart for the algorithm used to estimate parameters of the of proposed call admission policy parameters.

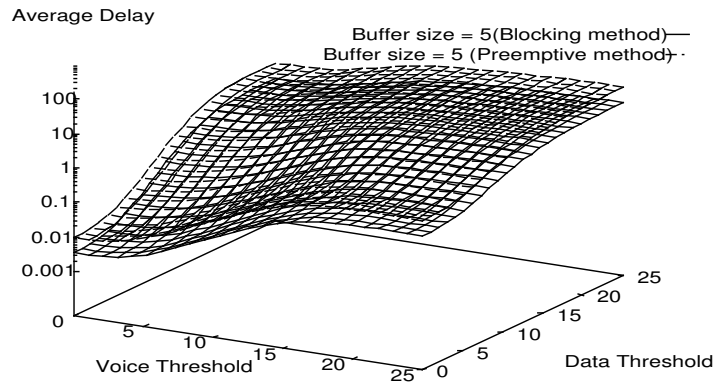


Figure 30: Average delay versus two thresholds for two types of methods at buffer size = 5.

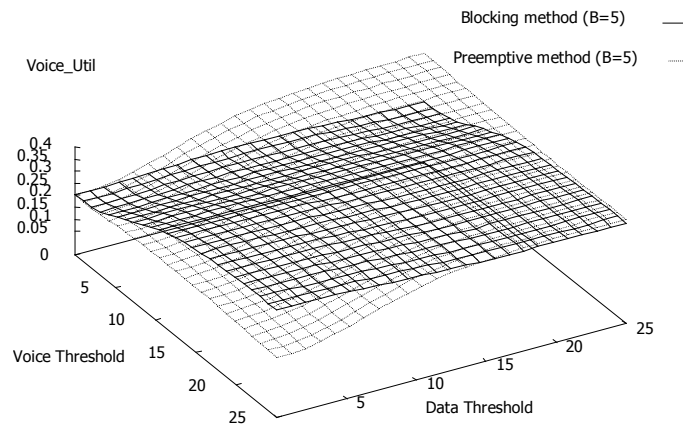


Figure 31: Utilization of voice calls versus Th_{voice} and Th_{data} using the blocking and the preemptive method at buffer size = 5.

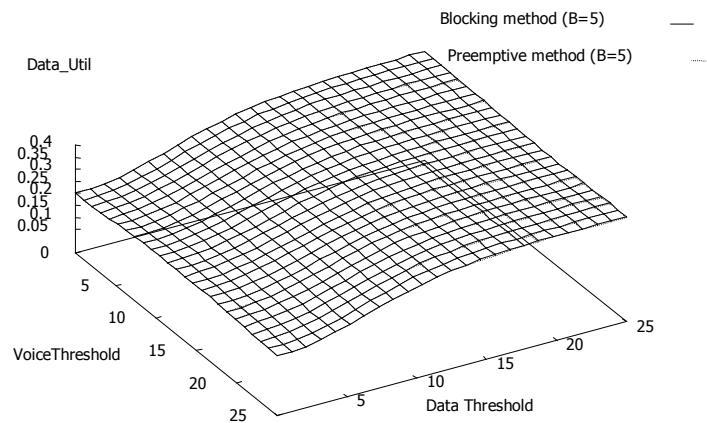


Figure 32: Utilization of data calls versus Th_{voice} and Th_{Ndata} using the blocking and the preemptive method at buffer size = 5.

Manuscript received on May 15, 1999

REFERENCES

- [1] M. Sidi and D. Starobinski, New Call Blocking versus Handoff Blocking in Cellular Networks, *Wireless Networks*, Vol. 3, No. 1, 1997.
- [2] S. Tekiany and B. Jabbari, Handover and Channel Assignment in Mobile Cellular Networks, *IEEE Communications Magazine*, Vol. 30, No. 11, pages 42-46, November 1991.
- [3] R. Ramjee, D. Towsley, R. Nagarajan, On Optimal Call Admission Control in Cellular Networks, *Wireless Networks*, Vol. 3, March 1997.
- [4] M. Naghshineh and M. Schwartz, Distributed Call Admission Control in Mobile/Wireless Networks, *IEEE Journal on Selected Areas in Communications*, Vol. 14, no. 4, pages 711-717, May 1994.
- [5] A. Acampora and M. Naghshineh, Control and Quality-of-Service Provisioning in High-Speed Microcellular Networks, *IEEE Journal on Selected Areas in Communications*, Vol. 1, No. 2, pages 36-43, 1994.
- [6] K. Kundert and A. Sangiovanni-Vincentelli, *Sparse 1.3 Package : A Sparse Linear Equation Solver*, Version 1.3a, April 1988.