A PRE-EMPTION FRAMEWORK FOR UMTS SATELLITE SYSTEM SUPPORTING MULTIMEDIA TRAFFIC

Anju Pillai, Yim-Fun Hu, Rosemary Halliwell[1]

University of Bradford, United Kingdom

ABSTRACT

The pre-emption procedure is an important part of the radio resource management when dealing with the emergency traffic. It allows resources to be allocated to higher priority connections by pre-empting lower priority connections. The provision of the pre-emption mechanism becomes much more important in the case of satellite systems such as the Inmarsat Broadband Global Area Network system, which aids in providing the communication during a catastrophe. This paper focuses on the pre-emption framework for a Universal Mobile Telecommunications System-based satellite systems. Three algorithms have been proposed, Greedy, SubsetSum and Fuzzy pre-emption algorithm. Extensive simulations are carried out for the three algorithms and their performances are compared against each other. Simulation results show that the Fuzzy pre-emption algorithm performs better than the other two algorithms.

KEYWORDS

UMTS, MATLAB Simulation, pre-emption, multimedia traffic.

1. INTRODUCTION

In general, networks are designed in a way such that the performance of a system can be maintained under specified maximum traffic load conditions. However, beyond the maximum load, the system performance starts to deteriorate and eventually leads to network failure. Under a highly congested state irrespective of the cause, whether expected or unexpected, it is necessary to minimize network failure [1]. In order to achieve such an objective, the network should alter the normal resource management procedures, in particular admission control by including pre-emption measures. A pre-emption control mechanism is based on the priorities assigned to all connections using a predetermined criteria set by the system and can be triggered to prematurely stop one or more existing connections in order to admit a new connection.

Several schemes have been proposed in the literature for pre-emption control in different networks [2-6]. A priority scaled pre-emption scheme Third Generation Partnership Project Long Term Evolution (LTE) networks using allocation and retention priority (ARP) has been proposed in [2]. This paper assumes that each bearer is mapped to a single service data flow and that the resource requirement of each bearer is fixed. The proposed technique suggested the pre-emption of the resources up to minimum QoS level from all lower priority bearers. The priority scaled pre-emption allowed the amount of resources pre-empted from the lower priority bearers to be proportional to their priorities. This

---

[1] Nee Wyatt-Millington

method was compared to the conventional pre-emption technique where the pre-emption of the resources up to minimum QoS level from all lower priority bearers is performed starting with the lowest priority bearer. The priority scaled pre-emption was shown to perform better than the conventional method in terms of the number of dropped and blocked active bearers. Although, the proposed technique has shown to outperform the conventional method under the given assumptions, the applicability of such scheme does not apply to a satellite system where several connections are tuned to one bearer along with the capability of the system supporting multimedia traffic with different QoS requirements.

In [3, 4], the pre-emption control was shown as one of the component of an adaptive bandwidth borrowing admission control scheme for cellular networks. In this scheme, the pre-emption control mechanism was triggered during the admission of the handoff calls if there was not sufficient bandwidth to borrow from the lower priority active calls. The pre-emption policy applied call pre-emption on one or more active calls in decreasing order of their allocated bandwidth, until the resource needed to admit an incoming handoff call could be satisfied.

The authors in [5] proposed two pre-emption based resource allocation schemes, last-come-first-pre-empted (LCFP) and path-prediction-based-pre-emption (PPBP); that could efficiently support multiple traffic types such as voice, video, data in an integrated heterogeneous wireless and mobile network. The authors assumed K different types of wireless and mobile networks out of which one network covers the entire service area consisting of many homogenous cells with lower bandwidth service. The remaining networks have limited coverage with one cell randomly distributed in each cell of the single wide coverage network. These networks were assumed to provide higher bandwidth service. Two types of traffic were considered: delay sensitive real-time traffic, such as voice and video, and delay tolerant non-real time traffic. An incoming real-time call, either new or handoff call could pre-empt ongoing non-real-time calls in the same cell of the network. A higher priority real-time call could be accepted in the network if there were enough resource for the call else the non-real time calls would be checked for pre-emption. In the LCFP scheme, the order of pre-emption was based on the descending order of time when ongoing non-real-time calls were accepted by the system. Therefore, the last accepted non-real-time call would be pre-empted first which allowed the earlier accepted calls to finish their service time so that the occupied bandwidth can be released more quickly. However, this scheme is not appropriate for the connections with different holding time which indicates the time the connection is in the system. An earlier accepted call may have much longer holding time than the last accepted call. On the other hand, in the PPBP scheme, the location information of the mobile user was known which was used to calculate the time, T, it took an active non-real time mobile user to reach a network providing higher bandwidth service before it moves out of the current cell. The non-real time call with the smallest value of T was pre-empted first. Such a scheme is only suitable to the given system architecture.

A threshold based pre-emption scheme for cellular network was presented in [6] . The purpose of this scheme was to guarantee a certain amount of resources to lower priority calls while allowing a higher priority call such as emergency calls an immediate access to the network. The amount of pre-emption was decided by a pre-emption threshold value which could be tuned according to the channel occupancy rate and traffic rate. If the number of channels occupied by the higher priority calls was less than the threshold, pre-emption was allowed otherwise the higher priority call would be blocked. Hence, the higher the threshold, the higher the resources used by the higher priority calls. However, the pre-emption decisions of the schemes presented above were only based on a single criterion, which might not be able to provide an optimum solution.

This paper presents three pre-emption algorithms incorporated in the connection admission control (CAC) framework, namely Greedy, SubSetSum, and Fuzzy, based on different pre-emption policies for a Universal Mobile Telecommunications System (UMTS)-based satellite system. The Greedy and the SubSetSum pre-emption algorithms are single-criterion algorithms using the resource utilization of a connection as the pre-emption criteria and differ in the order in which the connections are pre-empted. The Fuzzy pre-emption algorithm uses an intelligent algorithm based on fuzzy logic that considers multiple criteria: priority, resources utilization and the remaining time of the connection to be pre-empted in the system.

Figure 1 shows the UMTS satellite network architecture used for the implementation of the CAC framework.

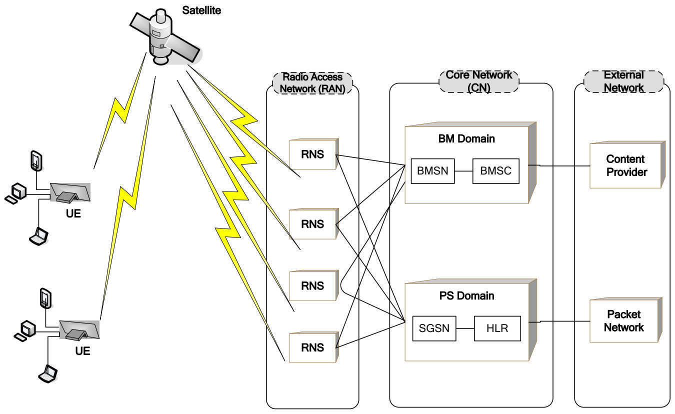

Figure 1 UMTS Satellite Network Architecture

The architecture is divided into three segments:

- User equipment (UE) segment consists of a transportable satellite modem, the mobile terminal (MT) connecting to a terminal equipment such as a personal computer or a PDA, allowing users access to UMTS services. Multiple terminal equipment can be connected to one MT such that multiple data connections can belong to one MT;

- Satellite segment consists of a multi-beam geostationary satellite system that provides a transparent link between the user equipment and the radio network controller (RNC). Multi-Frequency, Time Division Multiplex and Multi-Frequency, Time Division Multiple Access are adopted in the forward (satellite-to-user) and the reverse (user-to-satellite) links, respectively. In the forward direction, each satellite channel has a bandwidth of 200 kHz, which is termed as forward subbands.;

- Ground segment consists of the radio access network and the core network (CN). The radio access network, which handles all radio-related aspects of the ground network, consists of a number of radio network subsystems. Each radio network subsystem consists of a RNC and a radio frequency subsystem. The CAC controller resides in the RNC. The RNC interfaces to the CN for switching and routing data connections to and from external network. The CN consists of the packet switched elements such as the serving general packet radio service (GPRS) support node and the gateway GPRS support node in the packet switching domain and the multicast switched elements such as broadcast multicast service node and broadcast multicast service centre in the broadcast multicast domain. However, only packet switching domain has been considered for this study.

The proposed CAC framework focuses on the resource availability in the forward direction using a fixed number of forward subbands to admit data connections. Each MT is tuned to a particular forward subband and therefore, all the data connections associated with that MT are transmitted on one forward subband. The system supports different MT classes pertaining to the size of their antennas and operating scenario of the MT such as portable, land-vehicular, maritime or aeronautical.

The rest of the paper is organised as follows. Section 2 describes the extension of the work presented in [7-9] for adaptive admission control to support the pre-emption control. A brief description of the CAC Processor is presented followed by the detailed flowchart of the pre-emption algorithms. The simulation parameters used to analyse the system performance are presented in Section 3. Section 4 compares the three pre-emption schemes, and in Section 5, the paper is concluded.

2. CAC FRAMEWORK FUNCTIONAL MODEL

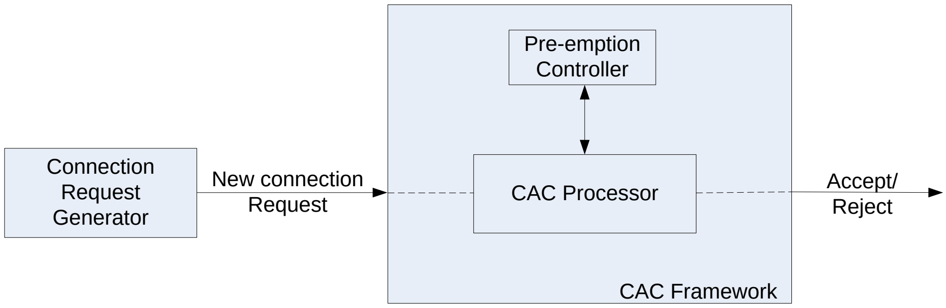

Figure 2 CAC Framework Functional Model

Figure 2 shows the CAC framework consisting of 2 different functional entities, the CAC processor and pre-emption controller. The CAC processor is central to the CAC framework. It runs an adaptive admission control algorithm when triggered by a new connection request and decides whether the new connection can be admitted. The adaptive CAC algorithm takes into account the link condition and the class of the MT while calculating the resources used by the connections on a forward subband. The link condition may vary depending on the weather condition, user mobility etc. The MT classes are categorised according to the data transfer capabilities of the MTs depending, on the size of the antenna. The pre-emption controller will be triggered by the CAC processor if all the available subbands cannot accommodate any new connection request. Based on the pre-emption policy used, one or more lower precedence ongoing connections are pre-empted in order to admit the higher precedence new connection.

2.1 Connection request generator

The Connection request generator is responsible for generating new connection requests. Three UMTS traffic classes are supported: the streaming class, the interactive class and the background class [10,11]. For the interactive traffic class, a UMTS QoS attribute known as Traffic Handling Priority (THP) is used to identify the priority of the connections within the interactive class. The THP parameter is only applicable for the interactive class and it can take three values: THP1, THP2, THP3, depending on the type of application which follows the following priority order: THP1>THP2>THP3. The system considers four types of applications; video streaming, netted voice, web browsing, and e-mail. The inter-arrival time for each connection requested is generated using Poisson distribution and the connection holding time is generated using exponential distribution. Table provides the summary of the type of applications supported by the system.

Application Type	Mode of Transmission	Traffic Class	THP	Priority	Pre-emptable
Video Streaming	Unicast	Streaming	N/A	1	No
Netted Voice	Unicast	Interactive	1	2	Yes
Web Browsing	Unicast	Interactive	2	3	Yes
Email	Unicast	Background	N/A	4	Yes

Table I Types of applications supported by the system

2.2 CAC Processor

The CAC Processor consists of four functional blocks: a *subband selector*, an *effective bandwidth estimator*, a *resource consumption estimator* and an *admission decision controller*, as shown in Figure 3[7-9].

The *subband selector* selects the forward subband for a new MT from the list of available subbands. Two methods have been proposed: (i) *MinConnSubSel* selects the forward subband with the minimum number of connections running from the list of available subbands. This method allows a basic form of load balancing, (ii) *random* method randomly selects a forward subband.

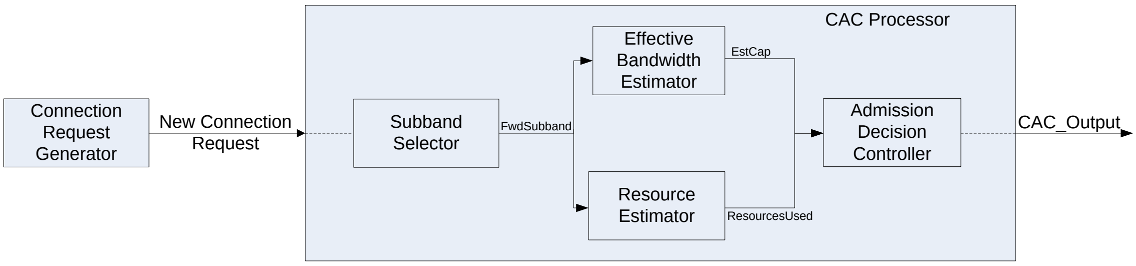

Figure 3 Functional Block Diagram of CAC Processor

The *effective bandwidth estimator* estimates the bandwidth requirement of the connection based on their statistical characteristics. All traffic sources are modelled as ON-OFF process [12]. The estimated capacity is calculated using the peak rate, R_{peak}, and the source utilization, ρ, of the connection as follows:

$$EstCap = R_{peak} * \rho \quad (1)$$

The source utilization represents the fraction of time the source is active

The function of the *resources utilization estimator* is to calculate the total resources used on the given forward subband. The adaptive CAC algorithm takes into account the link condition and the class of the MT while calculating the resources used by the connections on a forward subband. The forward frame in the physical layer carries the data on a forward subband from the RNC to MT. Each forward frame is 80ms long and consists of eight FEC blocks. The forward subband supports a range of code rates and is bounded by the lowest and the highest code rates in order to maintain a packet error rate of 10^{-3} under different radio link conditions. The code rate is a fractional number that indicates the portion of the total amount of information that is useful. Hence, the resource used on a forward subband is calculated as follows:

$$ResourcesUsed_{givenfwdsubband} = \sum_{allconnections} EstCap / coderate \quad (2)$$

where the *coderate* varies constantly in adaptive CAC algorithm.

The admission decision controller is responsible for performing one of the following actions on the new connection request: (i) admit new connection on the given forward subband, (ii) admit new connection on the given forward subband by pre-empting one or more, lower priority connections and (iii) block the new connection.

2.3 Pre-emption controller

The functionality of the pre-emption controller is to find the connections which can be pre-empted according to pre-defined pre-emption criteria, such that the new connection can be admitted on the given forward subband. Three pre-emption algorithms have been proposed; Greedy, SubSetSum and Fuzzy pre-emption algorithms.

2.3.1 Greedy pre-emption. The Greedy pre-emption algorithm pre-empts the connections with the lowest resource usage. This algorithm performs by pre-empting the lower priority connections in the ascending order of the resources consumed by the connections and in doing so, it greedily pre-empts more connections than required. Figure 4 shows the flowchart for the Greedy pre-emption algorithm.

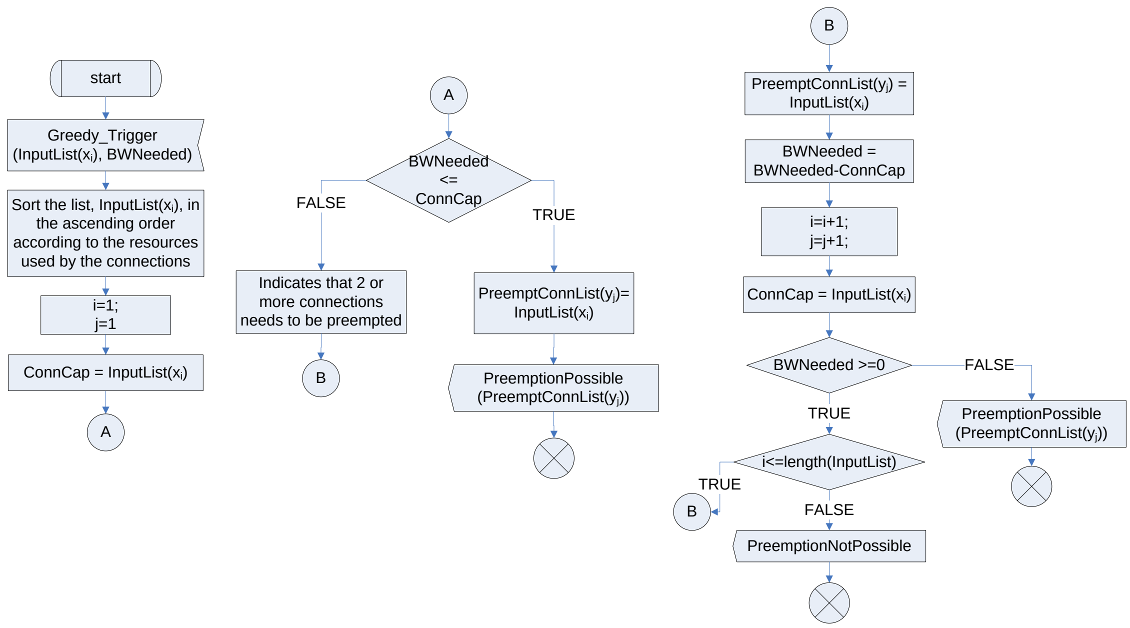

Figure 4 Flow chart of Greedy pre-emption algorithm

2.3.2 SubSetSum pre-emption. The SubSetSum pre-emption algorithm selects the pre-emptable connections in an optimum manner. The algorithm is based on a SubSetSum problem [13] which states that given a set A of positive integers such that $A = [a_1, a_2, \ldots\ldots, a_n]$ and a positive integer called the target sum, s, where $s \le \sum_{i=1}^{n} a_i$ there exists a column vector $X = [x_1, x_2, \ldots \ldots, x_n]^T, x_i \in [0,1]$, such that AX is as large as possible but not greater than s. Figure 5 shows the flowchart for the SubSetSum pre-emption algorithm.

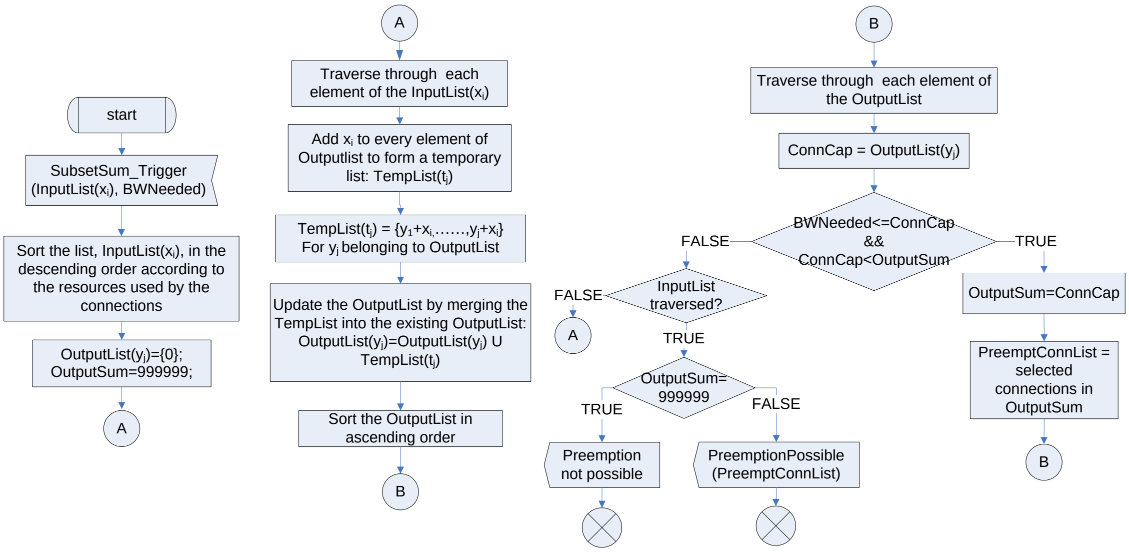

Figure 5 Flow chart of SubSetSum pre-emption algorithm

Adapting the SubSetSum problem to the pre-emption problem is equivalent to finding a subset of connections belonging to a set of pre-emptable connections, $A = [a_1, a_2, \ldots\ldots, a_n]$ where n is the total number of pre-emptable connections on a forward subband and a_j is the resource consumed by connection j, such that the sum of this subset of A, i.e. AX where X is defined as before, is equal to or great than the resource, r, where $r \le \sum_{i=1}^{n} a_i r$, requested by a new connection. In this case, the target sum is no longer a fixed value, but a value with a minimum bound r and a maximum bound $s \le \sum_{i=1}^{n} a_i$.

Because there can be more than one solution for X, the one which gives the minimum value of AX but greater than r will be adopted. Any element x_i equalling to 1 in the chosen solution of X will lead to connection i being dropped. Hence, the algorithm drops one or more lower priority connections such that the total resource consumption of these connections is just enough to accommodate the new connection request. This enables high priority connection requests to be admitted without dropping more than necessary existing lower priority connections, thus minimizing the bandwidth released by the pre-emptable connections and also reducing the number of connections pre-empted. Hence, SubSetSum pre-emption algorithm is an improvement over the Greedy pre-emption algorithm.

2.3.3 *Fuzzy pre-emption.* The Fuzzy pre-emption algorithm is proposed to overcome the shortcomings of the SubSetSum algorithm and to further enhance the performance of the system. Although, SubSetSum algorithm provides an optimum solution by minimizing the bandwidth released by the pre-emption of the connections and by reducing the number of connections to be pre-empted, however, it is mathematically complex and requires more computation time. Also, it does not consider other factors such as priority of the connection when deciding which connections are to be pre-empted [14, 15].

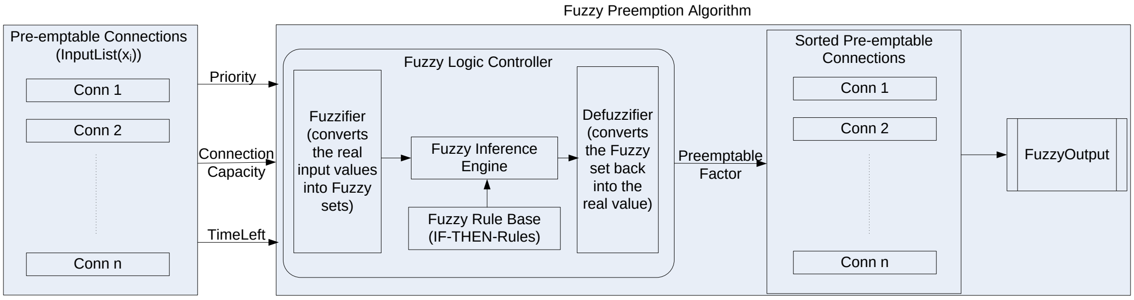

Figure 6 Block diagram of Fuzzy Pre-emption Algorithm

Figure 6 shows the block diagram describing the methodology and criteria used in the Fuzzy pre-emption algorithm. The main idea of the algorithm is to produce the output, *PreemptableFactor* for each pre-emptable connection using a given number of input criteria. *PreemptableFactor* indicates the odds of a connection to be pre-empted; the higher the value of *PreemptableFactor*, the greater the chance of the connection to be pre-empted and vice-versa. Once the *PreemptableFactor* is calculated for each pre-emptable connection, the list is sorted according to the value of *PreemptableFactor*. The sorted list is then sent to the *FuzzyOutput* procedure which selects the connections for pre-emption. The following three input criteria have been used to compute *PreemptableFactor*:

- *Priority* – indicates the priority of the connection which in turn depends on the type of traffic.
- *ConnectionCapacity* – indicates the resource utilized by the connection on the given forward subband.
- *TimeLeft* – indicates the remaining service time of the connection.

Table II shows the range used for the input and the output variables while designing the Fuzzy pre-emption algorithm. The range has been selected such that the design remains suitable for different traffic classes with varying QoS requirements.

The core of the Fuzzy pre-emption algorithm is the Fuzzy logic controller (FLC) [15]. It collects the input variables for each pre-emptable connection and based on that information, it produces the *PreemptableFactor* as an output. The FLC operates by converting the real or crisp values to the corresponding linguistic values of the fuzzy sets, which can be described using membership functions. A membership function can be represented by a curve or a line. Some of the most common shapes used for membership functions are Guassian, Trapezoidal and Triangular. The given input and output linguistic variables are assumed to have either triangular or trapezoidal membership functions which are described in the succeeding text:

- The triangular curve function, *trimf(x,[a,b,c])*, is a function of vector x and depends on three scalable factors, *a*, *b* and *c*. The parameters, *a* and *c* locate the 'feet' of the triangle, and the parameter b locates the peak.
- The trapezoidal curve function, *trapmf(x,[a,b,c,d])*, is a function of vector *x* and depends on four scalable factors, *a*, *b*, *c* and *d*. The parameters, *a* and *d*, locate the 'feet' of the trapezoid, and the parameters *b* and *c* locate the 'shoulders'.

Type of variable	Variables	Range
Input	*Priority*	2, 3, 4
Input	*ConnectionCapacity*	1 – 10 Kbits
Input	*TimeLeft*	0 – 1200 s
Output	*PreemptableFactor*	0 – 1

Table II Range of input and output variables used in Fuzzy pre-emption algorithm

The Fuzzy variables assumed for the input linguistic variables, *Priority*, *TimeLeft* and *ConnectionCapacity*; and for the output linguistic variable, *PreemptableFactor* are defined respectively as:

$$T(Priority) = \{High, Medium, Low\}$$

$$T(TimeLeft) = \{T1, T2, T3, T4\}$$

$$T(ConnectionCapacity) = \{Low, Medium, High\}$$

$$T(PreemptableFactor) = \{PF1, PF2, PF3, PF4, PF5, PF6, PF7, PF8, PF9\}$$

Figures 7–10 show the membership functions of the input variables.

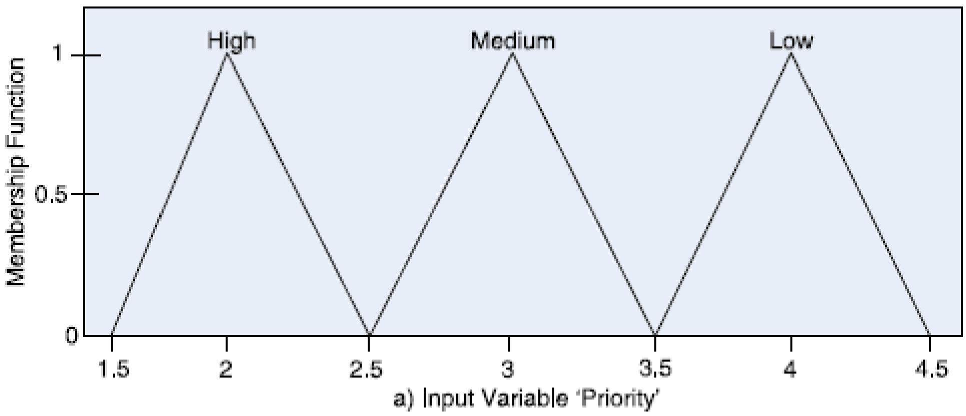

Figure 7: Membership function plot for Input variable, *Priority*.

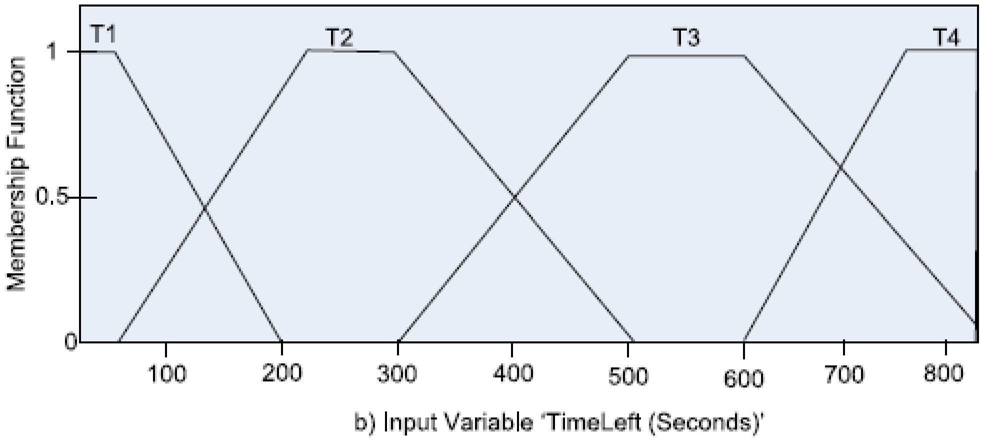

Figure 8: Membership function plot for Input variable, *TimeLeft.*

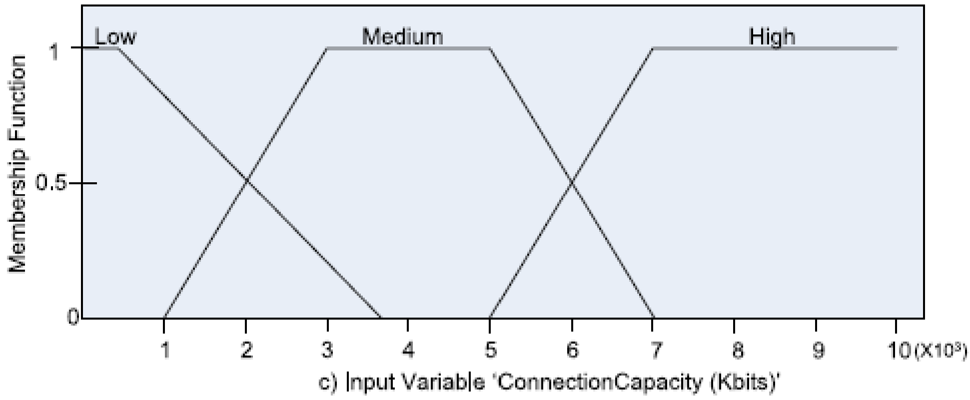

Figure 9: Membership function plot for Input variable, *ConnectionCapacity*.

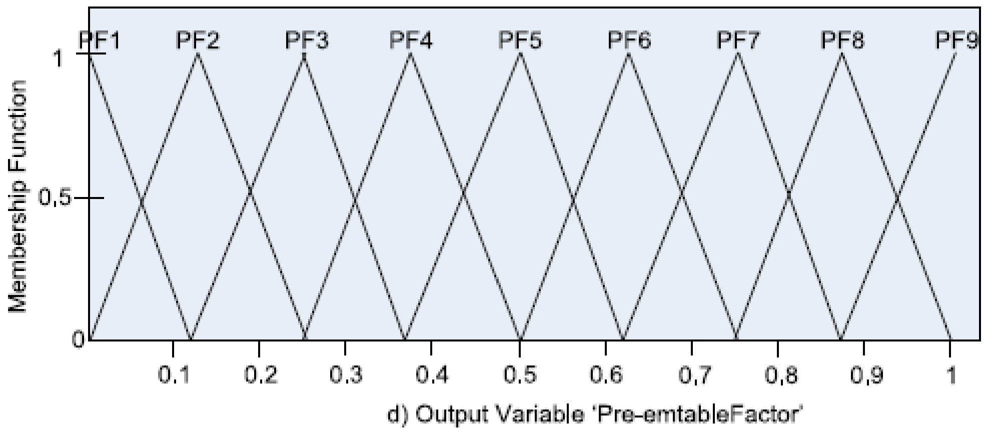

Figure 10: Membership function plot for Output variable, *PreemptableFactor*.

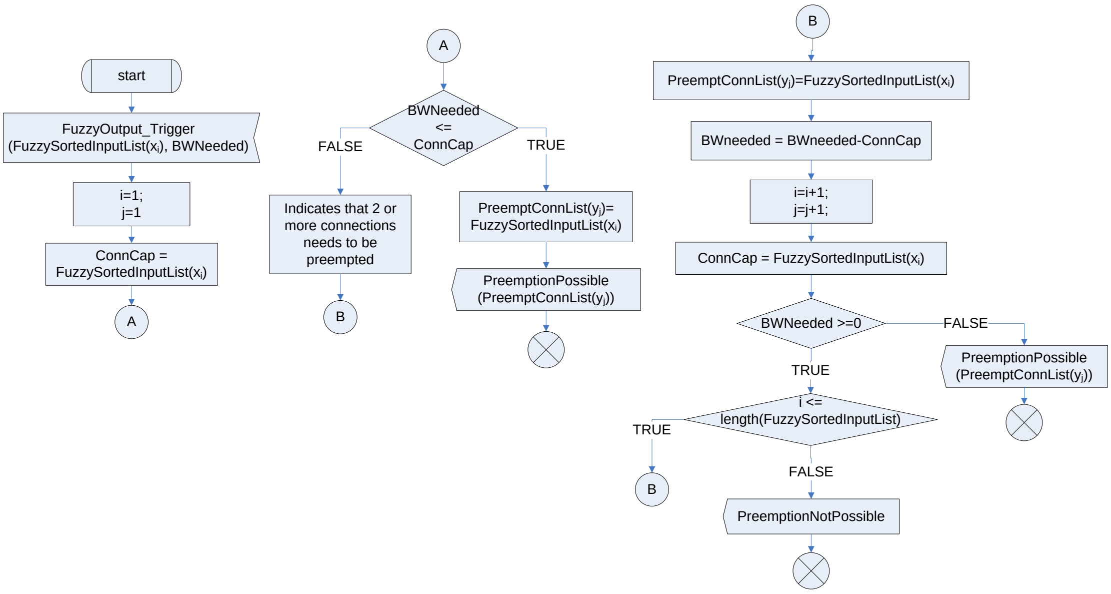

Figure 11 Flow chart of Fuzzy pre-emption algorithm

Figure 11 shows the flowchart for the Fuzzy pre-emption algorithm.

1. The FLC block runs for each pre-emptable connection in *InputList*(x_i) as shown in Figure 6 and calculates the corresponding *PreemptableFactor*. Upon completion of the FLC block, the pre-emptable connection list is sorted according to the *PreemptableFactor* to form the *FuzzySortedInputList*(x_i), where $i = 1 \ldots n$ and n is the number of existing pre-emptable connections. The *FuzzyOutput_Trigger* carrying the *FuzzySortedInputList*(x_i) and *BWNeeded* parameters is sent to the *FuzzyOutput* procedure shown in Figure 6, where *BWNeeded* is the required bandwidth for the new connection.

2. Set i = 1 and j=1, where j is the index of the connections selected for pre-emption, *PreemptConnList*(y_j). Define *ConnCap* as the bandwidth utilized by the next available connection in the *FuzzySortedInputList*(x_i), then when i=1:

$$ConnCap = FuzzySortedInputList(x_1)$$

3. If ($BWNeeded$ <=*ConnCap*), then

$$PreemptConnCap(y_j) = FuzzySortedInputList(x_i)$$

   The pre-emption process is completed and stops at this step.

4. If ($BWNeeded$ >*ConnCap*), more than one connection is required to be dropped in order to admit the new connection.

5. Set

$$PreemptConnCap(y_j) = FuzzySortedInputList(x_i)$$

$$BWNeeded = BWNeeded - ConnCap$$

6. Set $i = i + 1$ and $j = j+1$.

$$ConnCap = FuzzySortedInputList(x_i)$$

   If $BWNeeded$ < 0, the pre-emption process is completed and stops at this step.

If $BWNeeded$ >= 0,

If $i \leq n$,

$BWNeeded = BWNeeded - ConnCap$

Goto step 5

If $i > n$, then pre-emption is not possible and the process stops here.

3. SIMULATION PARAMETERS

The following simulation parameters have been defined to analyse the system performance under different scenarios:

- Blocking ratio: When a new connection arrives and finds no resources available or no lower priority connection to pre-empt, the connection will be blocked. The blocking ratio is calculated as the ratio of the number of connections rejected/blocked to the total number of connection request made.
- Dropping ratio: The lower priority connections are pre-empted in order to admit a higher priority connection. The dropping ratio is calculated as the ratio of the number of connections dropped/pre-empted to the total number of connection request made.
- Number of successful connections: The number of admitted connections which finish their service time such that the connections are not dropped.
- Pre-emptable data size: This indicates the total data size released by all the pre-empted connections over a period of simulation time.
- Computation time: This is used to measure the performance of the pre-emption algorithms. It indicates the time MATLAB takes to run an algorithm. For a given simulation, the amount of time to run the algorithm each time the algorithm is triggered in a given simulation time, is measured. The computation time for the given algorithm is the average of the measured times over a period of the simulation time.
- Revenue generation: This is also used to measure the performance of the pre-emption algorithms. A time-based charging mechanism has been applied to calculate the revenue. A tariff of w pence/sec is used for calculating the charge for the session the connection is active in the system. The value of w depends on the type of the traffic and is proportional to the priority of the application. Hence, a higher priority connection generates higher revenues and a lower priority connection generates lower revenues.

4. SIMULATION SCENARIOS AND RESULTS

The purpose of the simulation is to test the performance of the system using proposed pre-emption algorithms. The pre-emption procedure is activated when the system is congested. The selection of the simulation parameters in the given scenario is such that the system is heavily congested very quickly. For this purpose, the system is configured with only one available forward subband supporting a data rate of 512 Kbps under good link condition. Table III summarizes the MATLAB simulation parameters for the given scenarios. Parameters used in the simulation scenarios are based on [16].

Common Simulation Parameters		Values
Total No. of connections		32
Number of MTs		20
Number of available forward subbands		1
Number of connections of each traffic type	Video Streaming	8
	Web Browsing	8
	Netted Voice	8
	E-mail	8
Data rate (kbps)	Video Streaming	32
	Netted Voice	60
	Web Browsing	32
	E-mail	120
Avg. Holding time (sec)	Video Streaming	300
	Netted Voice	240
	Web Browsing	200
	E-mail	150
Source Utilization	Video Streaming	0.8
	Netted Voice	0.3
	Web Browsing	0.2
	E-mail	0.2

Table III Simulation Parameters for Scenarios

Four types of unicast traffic with different traffic parameters [16] are considered; video streaming, netted voice, Web browsing and E-mail, where each traffic type generates eight connection requests. These scenarios were chosen as representative applications considered in the Inmarsat Broadband Global Area Network System.[2] The number of MTs is taken as 20. The admission control algorithm and the subband selection method are chosen as the *adaptive* and the *MinConnSubSel*, respectively.

4.1 Comparison between different pre-emption algorithms

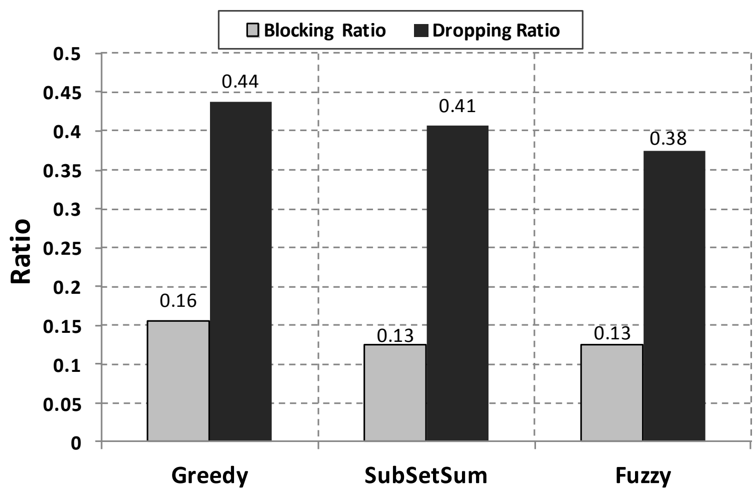

Figure 12 Comparison of the blocking and the dropping ratio for different pre-emption algorithms

[2] Private conversation with Mr Paul Febvre from Inmarsat.

Figure 12 compares the blocking and the dropping ratio for different pre-emption algorithms. As can be seen, the blocking ratio is slightly higher for the *Greedy* algorithm than the *SubSetSum* and the *Fuzzy* algorithms which have the same blocking ratio. Also, the dropping ratio is the highest for the *Greedy* algorithm followed by the *SubSetSum* and the *Fuzzy* algorithms. This is expected as the *Greedy* algorithm admits the higher priority connections by pre-empting as many lower priority connections as required starting with the lowest resource using connection. In doing so, a large number of lower priority connections are dropped. Hence, the dropping ratio is the highest. This also results in increased blocking of the new higher priority connections as there may not be enough lower priority connections to drop. The *SubSetSum* algorithm drops the connections in an optimum way by selecting the connection from the list of pre-emptable connections such that its resource consumption is just enough to admit a higher priority connection. Hence, as compared to *Greedy*, it drops fewer connections. The *Fuzzy* algorithm considers multiple criteria such as priority, remaining service time, and the connection capacity, while deciding the connections to be pre-empted. The rules are set such that the connection with the lowest priority, longest remaining time and smallest connection capacity has the highest chance of being pre-empted. Such rules ensure that the connections with short remaining service time are less likely to be dropped allowing more successful departures from the system, which in turn increases the possibility of admitting more connections and hence reduces the dropping ratio.

Figure 13 compares the number of successful connections for different pre-emption algorithms. As can be seen, all 8 video streaming connections and no email connections are admitted for each pre-emption algorithms. This is expected as the video streaming are the highest priority connections and the email are the lowest priority connections. For the netted voice and web browsing connections, the *Fuzzy* algorithm admits the maximum connections followed by *SubSetSum* and *Greedy*. This is in accordance with the increased dropping ratio shown in Figure 12 in the same order. As more connections are dropped, fewer connections are successfully admitted.

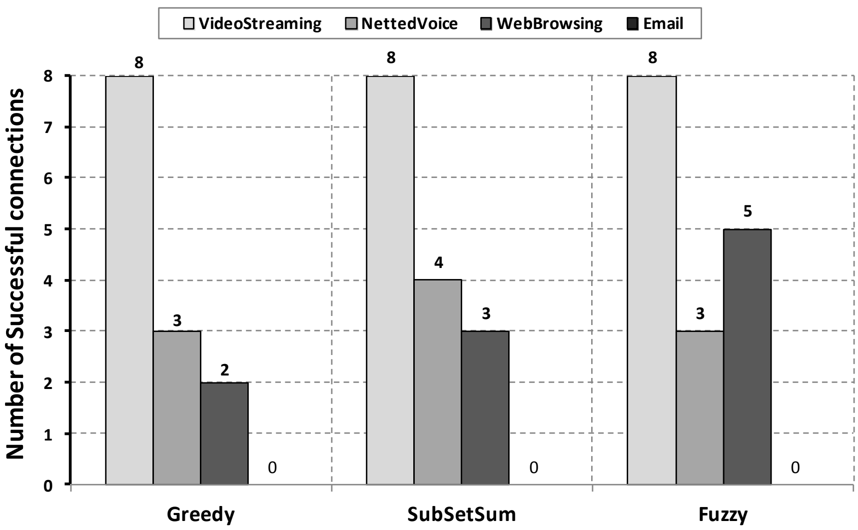

Figure 13 Comparison of number of successful connections for different pre-emption algorithms

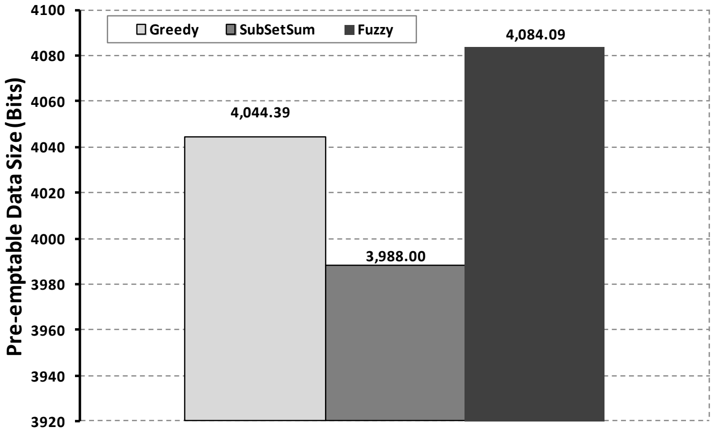

Figure 14 Comparison of pre-emptable data size for different pre-emption algorithms

Figure 14 compares the pre-emptable data size in bits for different pre-emption algorithms. As can be seen, the *SubSetSum* has the lowest pre-emptable data size followed by the *Greedy* and the *Fuzzy*. This is expected as the *SubSetSum* algorithm tries to minimize the pre-emptable data size by selecting the connections to be pre-empted such that its resource consumption is just enough to admit a higher priority connection. The *Greedy* algorithm also tries to keep the pre-emptable data size minimum by pre-empting as many lower priority connections as required starting with the lowest resource using connection. However, it does not do it in an optimum manner and hence, the pre-emptable data size is slightly higher than the *SubSetSum*. For the *Fuzzy* algorithm, although the rules ensure that the connections with low connection capacity have the highest chance of being pre-empted which helps to minimize the pre-emptable data size, however, at the same time the algorithm also tries to keep the dropping ratio to a minimum and to maximise the number of successfully admitted connections. In doing so, the pre-emptable data size is higher than both the *Greedy* and the *SubSetSum* algorithms.

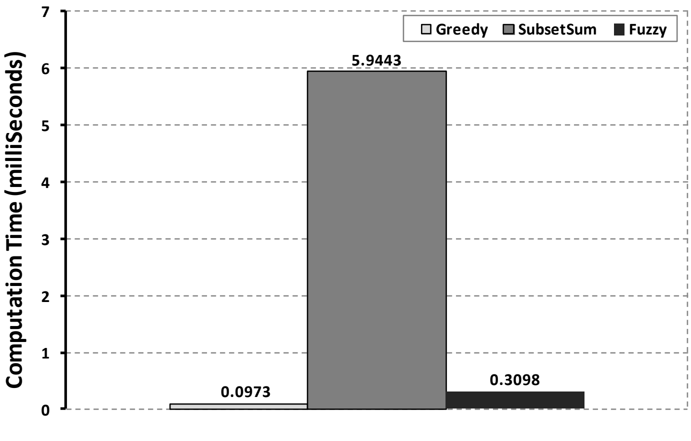

Figure 15 Comparison of computation time for different pre-emption algorithms

Figure 15 compares the computation time for different pre-emption algorithms. As can be seen, the *SubSetSum* algorithm has the highest computation time followed by the *Fuzzy* and the *Greedy* algorithms. This is expected since the *SubSetSum* algorithm is based on the SubSetSum problem which is computationally heavy and requires a lot of loop iterations to find the optimum solution. The *Fuzzy* algorithm also provides the optimum solution. However, it is easy to understand and takes much less time to compute as compared to *SubSetSum*. Although the *Greedy* algorithm gives higher blocking and dropping ratio, it is easy to implement and computationally efficient.

4.2 Comparison between pre-emption and no pre-emption

In this scenario, the performance of the system is compared under the following three conditions:

- *Pre-emption allowed* – All the connections are configured to be pre-emptable. The fuzzy pre-emption algorithm has been used.
- *Pre-emption not allowed* – No connections are configured to be pre-emptable.
- *Pre-emption randomly allowed* – Some connections are randomly selected to be pre-emptable.

Figure 16 shows the effect of the pre-emption procedure on the blocking and the dropping ratio. As can be seen, the blocking ratio is the highest for *pre-emption not allowed* and lowest for *pre-emption allowed*. However, the reverse is true for the dropping ratio. Such results are expected. With the pre-emption procedure enabled, fewer number of connections are blocked as the connection of a higher priority are admitted into the system by dropping of one or more existing lower priority connections during congestion. This in turn implies a higher dropping ratio. On the other hand, with *no pre-emption allowed*, the connections are blocked irrespective of the priority once the given forward subband is fully occupied. Hence, the blocking ratio is very high and since there is no dropping of the connections, the dropping ratio is zero.

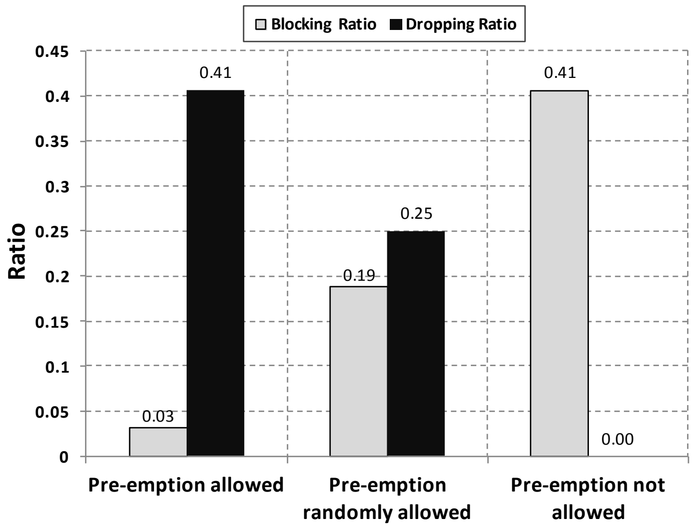

Figure 16 Effect of the pre-emption procedure on the blocking and the dropping ratio

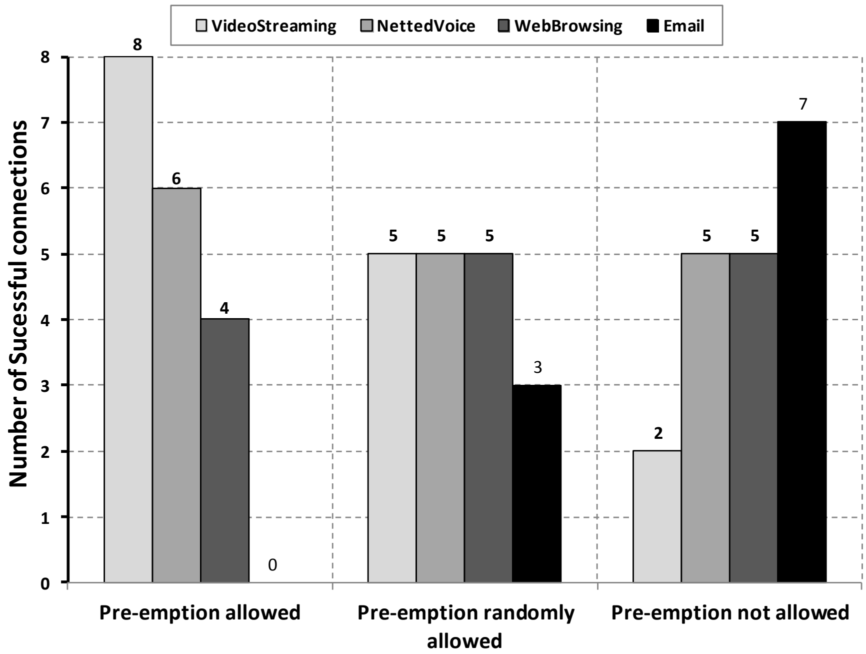

Figure 17 Effect of pre-emption procedure on the number of successful connections

Figure 17 shows the effect of the pre-emption procedure on the number of successful connections. As can be seen for *pre-emption allowed*, all the video streaming connections are admitted since they are the highest priority connections. As the priority drops for the different traffic in the given order, netted voice > web browsing > email, the number of successful connections also reduces as most of these lower priority connections are dropped. Since email connections have the lowest priority, they are dropped for all other higher priority connections and hence, there are no successful email connections. On the other hand for *pre-emption not allowed*, the connections are not admitted according to their priority resulting in only 2 video streaming connections being admitted as opposed to 8 in the case when pre-emption is allowed. Under the condition, *pre-emption randomly allowed*, only 5 video streaming connections are successfully admitted and the rest have been blocked since there were not enough pre-emptable lower priority connections.

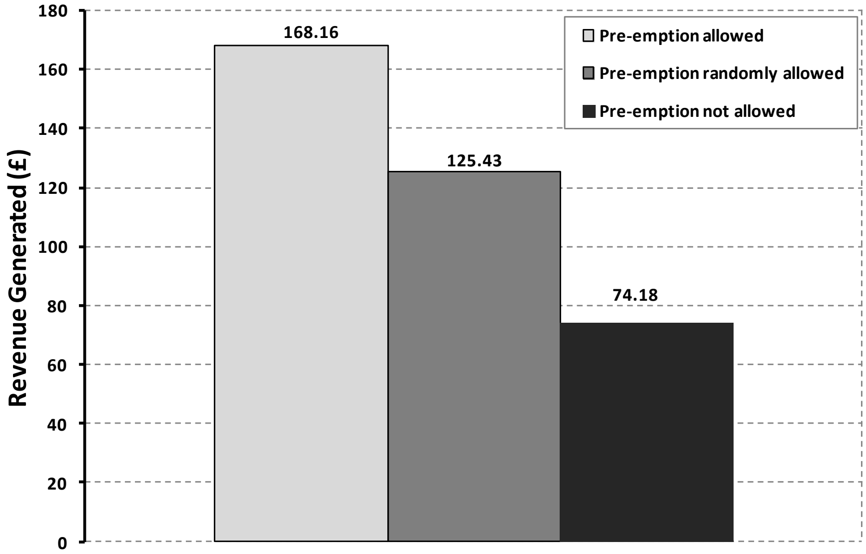

Figure 18 Effect of pre-emption procedure on the revenue generation

Figure 18 shows the effect of pre-emption procedure on the revenue generated by the network. As can be seen, the revenue generation is the highest for *pre-emption allowed* and lowest for *pre-emption not allowed*, and lies in between for *pre-emption randomly allowed*. This is directly related to the number of successful connections for each priority connections. For example, the pre-emption procedure tries to maximize the higher priority connections which have the highest revenues whereas with no pre-emption, the highest numbers of successful connections are email connections as shown in Figure 17 and hence, *pre-emption not allowed* has the lowest revenues.

5. CONCLUSIONS

In this paper, a pre-emption controller as a part of CAC framework has been presented for UMTS satellite systems. The pre-emption controller allows the possibility of pre-empting an existing connection of lower priority in order to admit a connection of higher priority. Three pre-emption control algorithms have been proposed; Greedy, SubSetSum and Fuzzy. The system supports mixed types of traffic such as video, Web browsing, netted voice and E-mail. Comparing the Greedy and the SubSetSum algorithms, the simulation results demonstrate that the SubSetSum algorithm performs 7% and 18% better than the Greedy pre-emption algorithm in terms of the dropping and blocking ratio, respectively. Hence, higher numbers of connections are successful using the SubSetSum algorithm. However, the computation time is 98% higher than the Greedy pre-emption algorithm because SubSetSum algorithm is mathematically complex and requires more computation time. As a result, a further improvement over the SubSetSum pre-emption algorithm has been proposed by the use of an intelligent Fuzzy pre-emption algorithm. It makes use of the expert system knowledge to provide a better system performance as compared to the Greedy and the SubSetSum pre-emption algorithms. The results indicate 7% improvement in the dropping ratio resulting in a slight increase in the number of successful connections but 94% shorter computation time as compared with the SubSetSum algorithm is achieved. In addition, the Fuzzy pre-emption algorithm also results in higher revenue generation. This work will be extended in future to support different classes of multicast traffic. The performance of the proposed pre-emption algorithms will be analysed under different multicast and unicast traffic conditions.

ACKNOWLEDGEMENT

This paper is based on the work that was performed as part of the BGAN extension project, which is funded by Inmarsat and the European Space Agency (ESA), under the framework of the ARTES (Advanced Research in Telecommunications Systems) programme. We would like to thank Paul Febvre from Inmarsat Global Ltd, and Juan R.Castro from ESA, for their support.

REFERENCES

1. Kordyboch K, Nousianinen S. Radio Resource Management in WCDMA-based Networks in Emergency Situations. *Proceedings of 18th International Teletraffic Congress* 2003.
2. Chadchan SM, Akki CB. Priority scaled preemption of radio resources for 3GPP LTE networks. *International Journal of Computer Theory and Engineering* 2011; 3 : 743-749.
3. Khanjari SA, Arafeh B, Day K, Alzeidi N. An adaptive bandwidth borrowing-based Call Admission Control scheme for multi-class service wireless cellular networks. *International Conference on Innovations in Information Technology* 2011; 375-380.
4. Khanjari SA, Arafeh B, Day K, Alzeidi N. An adaptive Call Admission Control scheme with complete bandwidth sharing for multi-class traffic in wireless networks. *International Conference on Software, Telecommunications and Computer Networks* 2010; 265-269.
5. Shen W, Zeng QA. Resource Allocation Schemes in integrated heterogeneous wireless and mobile networks. *Journal of Networks* 2007.
6. Zhou J, Beard C. A controlled Pre-emption scheme for emergency applications in cellular networks. *IEEE Transactions on Vehicular Technology* 2009; 3753-3764.
7. Pillai A, Chan PML, Hu YF. Adaptive Connection Admission Control for MBMS Compatible Satellite Systems. *14th Ka and Broadband Communications Conference* 2008.
8. Pillai A, Hu YF, Chan PML, Millington RW, Febvre P, Rivera J. Adaptive Admission Control for MBMS Satellite Systems. *International Workshop on Satellite and Space Communications* 2009; 326-330.
9. Pillai A, Hu YF, Halliwell R. An Adaptive Connection Admission Control Algorithm for UMTS based satellite system with variable capacity supporting multimedia services. *4th International ICST conference on Personal Satellite Services* 2012.
10. Nasser N, Hassanein H. Multi-Class Bandwidth Allocation Policy for 3G Wireless Networks. *Local Computer Networks Proceedings* 2003; 203-209.
11. 3GPP. End-to-End QoS Concept and Architecture (Release 6). TS 23.007 2005.

12. Wang H, Prasad D, Teyeb O, Schwefel HP. Performance Enhancements of UMTS networks using end-to-end QoS provisioning. *Wireless Personal Multimedia Communications* 2005; 18-22.

13. Chartrand MR. *Satellite Communications for the nonspecialist.* SPIE-The International society for optical engineering 2004; 1st Ed.

14. Klir GJ, Yuan B. *Fuzzy sets and fuzzy logic: Theory and Applications*. New York: Prentice-Hall Inc 1994.

15. Ross TJ. *Fuzzy Logic with Engineering Applications* 2nd ed.: John Wiley and Sons Ltd. 2004.