A TCP Driven CAC Scheme: Efficient Resource Utilization in a Leaky HAP-Satellite Integrated Scenario

M. LUGLIO
University of Rome "Tor Vergata"

G. THEODORIDIS
Aristotle University of Thessaloniki

C. ROSETI
University of Rome "Tor Vergata"

N. PAVLIDOU
Aristotle University of Thessaloniki

An integrated high altitude platform (HAP)-satellite communication system appears to be very suitable for a large set of scenarios including emergency situations, exceptional events, etc. In fact, the satellite capability to provide a broadband and ubiquitous access can be enhanced by the deployment of HAP that allows the use of low-power consuming, cost-efficient, and portable terminals.

To obtain an optimum utilization of radio resource, without renouncing to QoS satisfaction, a suitable call admission control scheme must be implemented. Nevertheless, transmission control protocol (TCP) behavior, mainly affected by the high latency and shadowing events, can impact call admission control (CAC) performance. Therefore, it would be desirable that the CAC scheme takes into account also the TCP congestion window real evolution.

We present an innovative CAC scheme that uses TCP statistics as one of its inputs and is able to manage different classes of users. Results show that CAC performance is significantly improved by introducing TCP statistics about network congestion as an input parameter.

Manuscript received June 9, 2006; revised April 30, 2007; released for publication June 6, 2008.

IEEE Log No. T-AES/45/3/933979.

Refereeing of this contribution was handled by M. Ruggieri.

This work was supported by the European IST-FP6 project: "SatNEx II-Satellite Communications Network of Excellence II."

Authors' addresses: M. Luglio and C. Roseti, University of Rome "Tor Vergata," Faculty of Engineering, Dept. of Electronics Engineering, Via del Politecnico 1, 00133, Rome, E-mail: (luglio@uniroma2.it); G. Theodoridis and N. Pavlidou, Aristotle University of Thessaloniki, Greece.

I. INTRODUCTION

Wireless systems represent an economic, flexible, and efficient means of providing the "last mile" connectivity, and in many scenarios they are the only viable solution. Among all solutions, thanks to their intrinsic cost-effectiveness in supporting broadcast and multicast services and their easy setup, satellite systems are particularly well suited to fulfill communication requirements especially in terms of large coverage and long-range mobility. They can also be utilized to cover sparsely populated areas where huge bandwidth resources cannot be provided through terrestrial infrastructures or regions where deployment of terrestrial facilities remains impractical. Unfortunately, in the case of GEO satellites, the long platform to ground distance imposes limitations in terms of both free-space path loss and long propagation delay. In particular, the latter strongly affects the overall network performance especially when TCP/IP protocols are utilized [1, 2].

In addition, quasi-stationary high altitude platforms (HAPs), positioned at altitudes up to 22 km (stratosphere), can act either as stand-alone base stations or as radio relay towards a satellite [3–5]. In particular, HAPs can be used to overcome several inherent impairments of the satellite systems, since they present potential benefits such as low propagation delay, rapid deployment time, and relatively low costs for the platform maintenance.

Therefore, in the present paper an integrated HAP-satellite architecture is adopted (Fig. 1) to provide broadband capability in a large set of scenarios in which a rapid and cost-efficient deployment of means is required:

exceptional events (Olympic Games, political meetings, etc.);
emergency situations (earthquakes, terrorist attacks, chemical disasters, wars, etc.) [6];
temporary overload of the terrestrial networks (increase capacity in high traffic areas).

We refer to a scenario where data are exchanged between the core network and fixed or small mobile terminals through a gateway → GEO → HAP path. The HAP allows connectivity both among ground terminals and with remote sites through the satellite segment. Thus, strictly speaking, the generated traffic can be considered HAP-based traffic, although many original satellite users may upgrade to HAP service and achieve meaningful benefits.

In this scenario, optimum bandwidth management and allocation is critical due to scarce availability of radio resources, and thus the implementation of an adequate call admission control (CAC) scheme can greatly improve performance by:

guaranteeing the QoS requirements of both the active and the candidate users,
maximizing the utilization of the network resources.

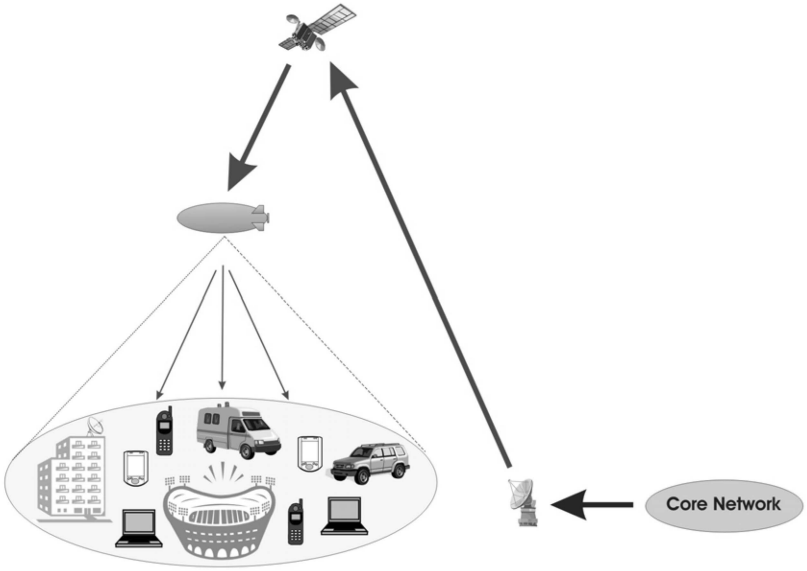


Fig. 1. Reference scenario.

Moreover, most of the current Internet applications run on top of transmission control protocol (TCP) that implements a reactive congestion control mechanism using an internal variable, called "congestion window," to limit the amount of data "in flight." The congestion window variation depends on both the round-trip delay perceived by the sender and the detection of TCP packet losses. Consequently, TCP control loop decides the actual data rate requirement, which is in average drastically lower than the nominal data rate in case of long propagation delays and frequent transmission errors. Therefore, a CAC scheme applied to TCP-based traffic over satellite taking into account only static parameters or physical measurements can lead to inefficient utilization of the precious radio resource.

In [7], radio blockage in IP network with encrypted network core is detected by using end-to-end measurement techniques [8], and such information is used to adjust CAC policies. QoS functions are implemented at the edges of the encrypted network to collect measurements aiming at identifying possible blockages and modifying CAC policies accordingly. A TCP-aware CAC scheme is proposed in [9]. It exploits the inverse proportionality between the average throughput of a TCP connection and packet loss probability. If a new connection request arrives when the system does not have enough resources, instead of rejecting the connection, a TCP-aware CAC scheme will limit its maximum transmission rate by adequately forced packet losses.

Improving on this, we propose an innovative CAC scheme based on a cross-layer interaction between data link and transport layer able to well serve QoS requirements of different classes of users, while, in the case of TCP connections via satellite, transport layer statistics are considered in order to avoid capacity waste.

In [10], the authors have already proposed a basic TCP-driven CAC algorithm coping with different coverage zones (leading to different packet error rate (PER) perceived at the transport layer). This paper introduces a more complex CAC algorithm with new features to take into account different terminal classes, and different QoS requirements.

To evaluate performance of the proposed CAC algorithm, we have developed a C++ tool that accepts as input TCP statistics obtained through the network simulator ns-2 [11]. Performance has been evaluated for different traffic load, percentage of TCP users, average PER, and blocking probability ratio (BPR) among three QoS classes in terms of both average throughput and overall blocking probability.

The paper is organized as follows. Section II presents the basic notions behind the CAC schemes and the TCP protocol; Section III describes the system reference architecture; Section IV includes a thorough analysis of the CAC-TCP interworking, while Section V deals with channel model used as reference in the simulations; Section VI shows simulation results and draws conclusions.

II. BASIC CONCEPTS

A. The Call Admission Control

CAC is namely an algorithm that runs at connection setup time in order to decide upon the admittance/rejection of the new connection, based on some predefined criteria:

1) *The availability of resources at the service network*—Primarily bandwidth, but also buffer space and processing capability can also rise as valuable resources.

2) *The connections QoS requirements*—Based on the network architecture, the average data rate and burst size and duration as well as acceptable error rate and leniency towards delay and jitter, are translated by the CAC algorithm into corresponding resource requirements.

The goal of CAC mechanism is twofold: achieving maximum utilization of the network resources and reducing the probability of violating the QoS guarantees provided to both the candidate and the already active users. As a result, the efficiency of a CAC algorithm is defined by the optimal choice of the acceptance criteria according to the special features of each telecommunication system [12] and by the development of methods for the estimation of both the network's status and the connections' requirements at any given instant [13].

Due to the importance of the admission process and the large volume of parameters that have to be taken into account, numerous CAC algorithms have evolved covering a wide range of applications and network architectures [14–16], exploiting the inherent characteristics of each scenario.

B. The Transmission Control Protocol

TCP provides to upper layers a connection-oriented, reliable, and byte stream service. At the transmitting side, TCP mainly receives a data flow from an application and subdivides it into properly-sized chunks, called "segments" or "packets." It then assigns a sequence number to each segment. At the receiving side, TCP reacts to the corrected and in-sequence reception of segments sending corresponding acknowledgements (ACKs) to the sender and delivering data to the application [17].

In addition, to control the amount of data "in flight" so as to exceed neither the available channel bandwidth nor the capacity of the receiver buffer, TCP implements a flow control mechanism called "sliding window," [18] which is basically used to determine how many (unacknowledged) segments the sender can have in transit towards the receiver and moves up in the segment space as soon as acknowledgements are received. Its size is constantly altered through a congestion control scheme that is based on an internal variable called "congestion window." In turn different algorithms, such as 1) slow start, 2) congestion avoidance, 3) retransmission timeout, and 4) fast retransmission and fast recovery [19], allow the congestion window to dynamically vary according to the network status [20]. At the same time, at the other end of the link, the receiver advertises a maximum window size, "advertised window," to the sender according to the free space in its internal buffer. As a result of this procedure, at every time instant, the value of the transmission window is given by the minimum between "congestion window" and "advertised window." As a consequence, TCP experiences severe efficiency degradation when utilized in satellite environment mainly due to the following three aspects [1, 21]: long latency, link availability and bit error rate (BER), and large delay-bandwidth product.

III. SYSTEM ARCHITECTURE AND RELATED USER CLASSIFICATION CRITERIA

The reference scenario concerns the forward channel of an integrated HAP-satellite architecture to provide wireless access to users (Fig. 1). The HAP actually operates as the network access point; the data streams originating in the core network and addressed to the users under the HAP footprint are transmitted through a satellite gateway towards the GEO Satellite that is in turn connected with the HAP, and finally delivered to the user terminals.

In spite of the entailed complexity, the integrated architecture is rather favorable in comparison with a stand-alone HAP or satellite network. In fact, the presence of HAPs:

1) relaxes satellite payload requirements;
2) allows the use of terrestrial-like terminals without the need for an intermediate module repeater (IMR) [22] (the proximity of the HAPs to the ground minimizes the free space attenuation);
3) smoothes the interoperability with other working terrestrial systems, as HAP access networks can utilize terrestrial standards;
4) enhances coverage in urban areas (high line-of-sight (LOS) probability);
5) strongly decreases perceived latency for communication under the same HAP footprint;
6) alleviates traffic management by handling local traffic.

On the other hand satellite introduces the advantages of

1) extending coverage,
2) interconnecting clusters of HAPs,
3) providing backbone connectivity to the core network or other cooperating systems and thus allowing the on-the-fly deployment of HAPs, without the need for any terrestrial infrastructure (in case of remote locations or emergency situations) [23, 24].

To evaluate performance of the presented TCP-driven CAC algorithm under a wide range of network conditions, the users of the integrated HAP-satellite system are classified according to the following three criteria.

1) *Bandwidth requirements*: Users are divided into three QoS-classes (traffic groups) with nominal data rate of 128, 256, and 512 kbit/s, respectively.

The bandwidth reservation made at each connection's setup time equals the user's nominal data rate.

2) *Transport layer protocol type*: Users are distinguished in TCP and non-TCP users. This distinction is made because only TCP connections experience actual requirements degradation due to the harsh wireless reception environment and the long round-trip delay.

3) *Mobility*: Users are classified into fixed, nomadic, and mobile which imply different propagation conditions and thus different PER. Fixed users are equipped with terminals in LOS towards the HAP. Nomadic users are equipped with portable terminals but working in stationary conditions. Mobile users, equipped with small terminals, are further classified in mobile-urban (mob_urb), mobile-suburban (mob_sub) and mobile-highway (mob_high) as a function of the specific kind of environment they are moving through.

The HAP is located at an altitude of 19 km offering coverage with a minimum elevation angle of 15 deg, which corresponds to a circular area of 70 km radius. For the sake of simplicity, the HAP footprint is assumed to be single-cell, since in our scenario resources are primarily limited by the satellite segment, and thus, to consider multi-cellular HAP architecture would be meaningless [5]. Furthermore, only 10 Mbit/s are assumed to be dedicated to the provision of services by the integrated HAP-satellite system, while the rest of the available bandwidth, both on the HAP and on satellite, is utilized to satisfy other traffic requirements (e.g., intra-HAP user traffic).

Finally, multi frequency time division multiple access (MF-TDMA) is assumed as multiple access technique for the return link (gateway-satellite-HAP-ground). According to the proposed scheme, to maximize the utilization of the network resources and to decrease the blockage probability, the time slots/channels are no longer assigned statically to each user for the whole duration of the connection but they are dynamically reallocated among all the active users on the basis of their instantaneous needs (estimated for the TCP users via real-time measurements of their actual TCP data rate).

IV. CAC-TCP INTERACTION DESIGN

TCP suffers from several impairments in the case of leaky links with a high delay. In particular, packet losses are misinterpreted as evidence of congestion, and thus, via the "congestion window" mechanism, the actual transmission rate is reduced. In addition, since the TCP "reaction time" is slowed down by the long perceived latency, the TCP data rate remains unjustifiably decreased. Therefore, since the satellite segment introduces about 560 ms in the overall round trip time (RTT), while the communication may be affected by transmission errors depending on propagation conditions [4], TCP behavior can entail a severe performance limitation in terms of actual average bit rate.

In this context, if radio resources are allocated taking into account TCP window evolution [25], in order to optimize the utilization of shared radio resources without compromising QoS requirements, the implementation of a CAC scheme that takes decisions on the basis of the constrains risen at the transport layer is recommended. In fact, if CAC decision is based only on static parameters, such as the "nominal rate" of each terminal's QoS-class, over-reservation of bandwidth for degraded TCP flows may imply that the candidate connections are blocked despite capacity availability.

To avoid this possible inefficiency, we propose a cross-layer interaction between transport layer (where TCP runs) and data link layer (where CAC runs). The basic idea is to monitor all the active connections through a TCP proxy in order to have continuous information about TCP dynamics and assign capacity to each link accordingly [25]. Then, samples of the current data rate are computed for each connection at regular intervals and passed as input to the CAC algorithm in order to monitor the maximum achievable rate of all the active terminals. In this way, CAC:

1) estimates the unused capacity more precisely;
2) reaches a solid acceptance/rejection decision of the new connections.

Moreover, in order to both minimize the signaling overhead and make the cross-layer interoperability viable, the TCP proxy is supposed to be located on the HAP, parallel to the CAC. Further details regarding the special features of the proposed TCP-CAC interaction are given in the following subsection.

A. The TCP-Driven CAC Algorithm

Should all the users have equal access technique to the wireless link, the low data rate flows would be more likely to be admitted, while the bandwidth consuming connections would be practically excluded from the network. Therefore, in order to achieve fair resource sharing among the three QoS classes, the call admission process makes use of a robust weighted priority scheme presented in [26]. In brief, the following steps are performed.

1) The network administrator defines the desirable blocking probability ratio (BPR) among the various traffic groups. Specifically, the variables $BPR_1 = BP_{256}/BP_{128}$ and $BPR_2 = BP_{512}/BP_{256}$ are introduced, where BP_i denotes the blocking probability of the QoS-class with nominal data rate equal to i.

2) The aggregate bandwidth of the network is divided into a number of segments equal to the number of the supported QoS classes, i.e., each QoS-class is assigned a certain percentage of the aggregate bandwidth.

3) A new connection is admitted only if the bandwidth assigned to the QoS-class of the connection is sufficient to fulfill its nominal data rate requirements, regardless of the overall availability of resources in the network.

4) The blocking probability is calculated separately for each traffic class and, at predefined regular intervals (expressed in number of new arrivals), the values for BPR1 and BPR2 are computed.

5) The bandwidth is reallocated among the QoS-classes on the basis of a comparison of the computed BPRs with their target values, so that the system can meet the desired behavior.

Moreover, such a resource management scheme allows the evaluation of the efficiency of the proposed algorithm against various distributions of the users' data rate, by simply manipulating BPR_1 and BPR_2 parameters.

However, the novelty introduced with the proposed TCP-driven CAC scheme lies in the notion that, since TCP's data rate is severely degraded under conditions of error prone links with long latency (gateway $\rightarrow$ GEO $\rightarrow$ HAP $\rightarrow$ user terminal), feedback from the transport layer would allow the CAC to refrain from over-provisioning bandwidth to TCP users. Being more specific, the TCP proxy performs sampling of all the active TCP connections to estimate the actual TCP transmission rate of each flow. This information, which is continuously updated, is utilized by the CAC algorithm at the arrival of a new user in order to calculate the current occupancy of the channel and decide whether there are available resources for the candidate user to be admitted.

Nevertheless, since the decision about admission/rejection relies upon the instantaneous data rate of the connection, CAC could lead to a highly unstable system, because abrupt alterations in the TCP congestion window size would cause the misconception of unavailability of resources. Specifically, there are two main reasons for overestimating the available bandwidth:

1) Short lasting deterioration of the communication path, which, in combination with the long RTT, causes the TCP congestion window to shrink;

2) All TCP connections go through a slow-start phase until they reach their working point.

Thus, if a new request for admittance coincides with the already active flows experiencing such conditions, then the candidate user will be unsoundly accepted. Yet, when the cause of data rate degradation is removed (improvement of propagation environment, end of transition phase), the increase of the offered load, in the presence of the extra traffic due to the new connection, will result in congestion. On the contrary, the reverse phenomenon can occur due to temporary improvement of the channel; in particular, temporary high data rates would potentially lead to the unjustified exclusion of any candidate user at that period.

Therefore, the CAC-TCP interaction should be further enhanced to guarantee the QoS requirements of the users. Thus, three main precaution mechanisms are added to the admission algorithm to avoid such inconvenience.

1) *Averaging the TCP-samples*: A long-term study of the TCP transmission window evolution of each connection is required so that the statistical processing of the TCP output will provide more solid and reliable information to the CAC algorithm. To this aim, at each sampling instant, the average of the new sample with all the previous samples of the connection congestion window is computed. Consequently, the input to the TCP-driven CAC algorithm is a sequence of average data rate samples that asymptotically converge to the average data rate of the connection. For the rest of the paper, we refer to these average values of the TCP samples, as aver_TCP_smpl in contrast to inst_TCP_smpl (instantaneous, unprocessed TCP feedback).

2) *Introduction of "Safety Margin"*: Although the prediction is much more solid by averaging the sampling sequence, still the chance that the network capacity is exceeded cannot be excluded. Therefore, our TCP-driven CAC scheme adopts a more pessimistic approach, considering each connection's data rate equal to the aver_TCP_smpl plus a safety margin. This margin is set as a percentage of the difference between the nominal data rate of the connection and the current aver_TCP_smpl:

$$\text{safety\_margin} = p \cdot (\text{nominal\_rate-aver\_TCP\_smpl}), \quad 0 \leq p \leq 1. \qquad (1)$$

Assuming that the data rate of each connection is higher than the feedback from the transport layer, some resources are reserved for the case that the traffic load increases beyond the currently measured TCP congestion window size. Thus, choosing the value for the parameter p is a tradeoff between avoiding congestion ($p = 1$) and maximizing the network throughput ($p = 0$). For $p = 1$, the CAC process does not take into account the TCP feedback, i.e., all the connections are treated on the basis of their nominal data rate, while for $p = 0$ only the current (averaged) TCP sample is considered. As it is presented in Section VI, for a wide range of configurations, $p = 0.3$ is considered the best tradeoff.

3) *Monitoring users in transition phase*: As long as a connection is still under transition phase, no safe predictions can be made regarding its average data rate (working point). In this context, during their slow start phase the connections are regarded to have their nominal (maximum) data rate.

Furthermore, needless to say that in case of congestion occurrence, i.e., the aggregate traffic load exceeds the channel capacity, no new connection is accepted. Finally, light bottleneck phenomena can be handled properly utilizing adequate buffering and scheduling methods.

B. Simulation Model

The TCP-driven CAC scheme is simulated through the offline combination of two different simulation tools that run sequentially.

At first, in order to acquire the TCP feedback, the Network Simulator ns-2 (release 2.27) [11] is executed by configuring sender and receiver nodes, running TCP NewReno as transport protocol and FTP as application protocol. The ns-2 simulations provide the TCP statistics of a connection characterized 1) by the channel model of the mobility-group that the user belongs to, and 2) by the nominal data rate of the user QoS-class. The output file contains samples of the actual data rate required by the user, taken at regular time intervals (10 s). Thus, the ns-2 output is a set of fifteen TCP-files (5 mobility-groups $\times$ 3 QoS-classes), one for each possible type of user.

Additionally, a C++ simulation tool capable of emulating the general network scenario and manipulating the TCP measurements has been developed. In particular, all the functionalities of the TCP-driven CAC scheme, as described in the previous subsection, are implemented. Moreover, adequate routines calculate the necessary metrics in order to evaluate the proposed algorithm effectiveness.

1) *Blocking probability* (BP): It is computed as the percentage of blocked calls in comparison with the total number of arrived calls.

2) *Average throughput* (AT): It depicts the utilization of the network resources. AT is calculated as the average occupancy of the overall capacity during the whole simulation.

3) *Probability density function* (pdf) of the norm_load that is defined as the aggregate traffic load—aggr_load (kbit/s) normalized to the system capacity—capacity (kbit/s). Overestimation of the available bandwidth can lead to congestion, which in turn causes degradation of the users' QoS, in terms of sustainable data rate, packet loss, and delay. Therefore, it is necessary that a scheme capable of quantifying the ability of CAC guarantees resource availability to all the accepted connections. Nevertheless, beyond the prediction accuracy of the CAC, performance of each flow is also highly dependent on various system configuration parameters, such as buffer size and scheduling scheme. In this framework, instead of measuring packet dropping, delay, etc., an alternative scheme is implemented to evaluate the algorithm efficiency in meeting users QoS requirements. In particular, the bottleneck phenomenon is not taken into account, i.e., all the arriving traffic load is supposed to be forwarded even if it exceeds system capacity, thus congestion does not affect the evolution of the flows. Instead, the distribution of norm_load is calculated. In more detail, at each TCP sampling instant, aggr_load equals the sum of inst_TCP_smpl of all the currently active connections and

$$\text{norm\_load} = \text{aggr\_load}/\text{capacity} \tag{2}$$

where norm_load > 1 corresponds to congestion. In turn, the range of norm_load values is divided into small chips with a step of 0.05 ($[0,0.05],[0.05,0.1],[0.1,0.15]\ldots$) and the number of norm_load samples belonging to each chip is computed. Finally, the probability that a norm_load sample lies in a specific chip (norm_load pdf) is calculated. As a result, norm_load pdf provides a solid metric of both the extent of congestion and the frequency of its occurrence.

Fig. 2 presents a flowchart describing thoroughly the functionalities of the TCP-driven CAC scheme, as analyzed above. All the parameters included in Fig. 2 are defined in Table I. The details of the mechanism guaranteeing the balance among the QoS-classes have been omitted, since they are out of the scope of the present paper. Inquiring readers can refer to [26].

V. CHANNEL MODEL IMPLEMENTATION

We reproduced a communication scenario where various types of terminals can simultaneously share the wireless channel while experiencing different channel propagation conditions. Thus, to evaluate performance of the proposed CAC scheme, packet error distributions (derived at TCP level) suitable for HAP-terminal communications are provided as inputs to ns-2.

As a first realistic assumption, the satellite-HAP and the gateway-satellite links are considered to be quasi error-free, since both the satellite-HAP link and the gateway-satellite link are in LOS. Therefore, the error rate of the end-to-end communication path corresponds to that of the HAP-terminal link. In general, it is widely accepted that some of the channel models developed for satellite environment [27] also apply to HAP communication. Specifically, the following two approaches have been followed, depending on the mobility class of the users and on the corresponding characteristics of the terminals.

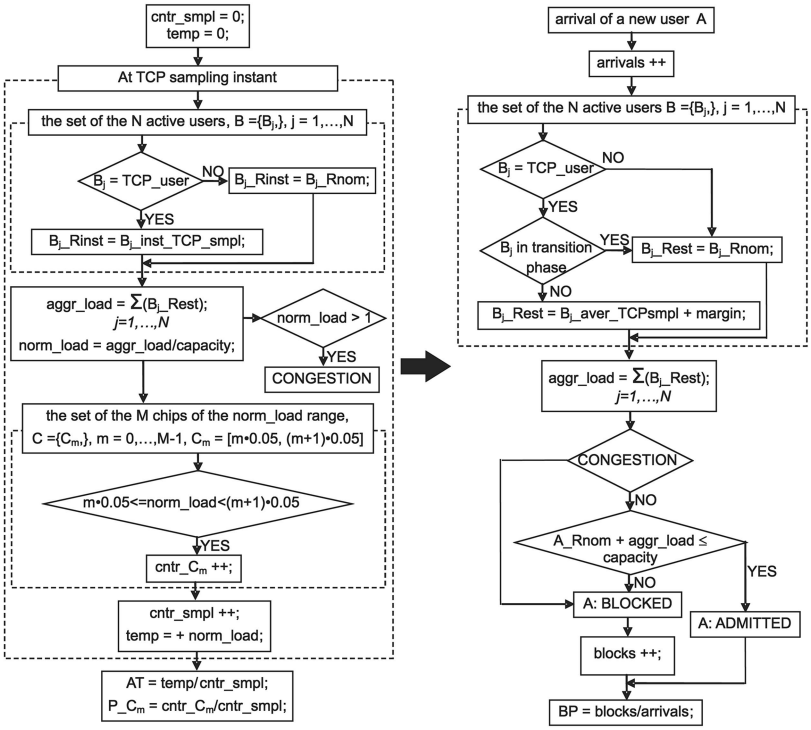


Fig. 2. Flow chart of the TCP-driven CAC along with the routines for computing BP, AT, and norm_load's pdf.

TABLE I
Flow Chart Parameters

Set of parameters common for the whole system	
arrivals	Aggregate number of arrivals to the network, either they have been admitted or rejected
blocks	Aggregate number of blocked calls
cntr_smpl	Counter holding the number of TCP samples taken per flow
C	The set of the M chips that the range of norm_load has been divided into.
	$C = \{C_m\},\ m = 0,\ldots,M-1,\ C_m = [m \cdot 0.05, (m+1) \cdot 0.05]$
cntr_C_m	Counter holding the number of norm_load samples lying in the C_m interval
Set of parameters per user U	
U_Rnom (kbit/s)	Nominal data rate required by the user U (it is equal to the nominal data rate of the user's QoS class)
U_Rinst (kbit/s)	Instantaneous (real—measured by the TCP proxy) datarate of the user U. For TCP users Rinst = inst_TCP_smpl, while for non-TCP users Rinst = Rnom
U_Rest (kbit/s)	Estimate of the user's current datarate requirements, taking into account TCP feedback as well as all the precaution mechanisms.

1) In the case of fixed terminals (including portable), we assume that LOS conditions are always ensured. This scenario is well modeled by means of a Rice distribution taking into account the effects of multipath [27].

2) As far as mobile terminals are concerned, according to the ITU-R recommendation [28], three different scenarios are identified (urban, suburban and highway) and a two-state channel model [29] is used to characterize the alternating LOS and shadowing condition. Finally, taking as reference the probabilities of the durations of each state reported in [28] for an elevation angle of 21 deg and considering probabilities of state transitions based on an approximation of the land mobile satellite (LMS) channel, a suitable two-state error model has been utilized for our simulations.

In the ns-2 simulations, statistical packet loss distributions have been adopted considering that a loss corresponds to a TCP segment received corrupted and thus to an error perceived at the transport layer. For fixed and portable terminals uniform loss distributions are used, with mean values equal to 10^{-4} and 10^{-3},

TABLE II
User Terminal Classes in Terms of Environment and Average PER

Terminal Characteristics	Average PER
Fixed	10^{-4}
Portable	10^{-3}
Mobile (urban environment)	$4 \cdot 10^{-2}$
Mobile (suburban environment)	$2 \cdot 10^{-2}$
Mobile (highway environment)	10^{-2}

respectively, while for mobile terminals a two-state Markov model has been adopted. The "bad" state has an average duration consistent with values provided in [28] and corresponds to all TCP packets dropped (PER = 1). On the other hand, during the "good" state, a uniform loss distribution with a low average PER is considered (PER ≈ 0); as a result, the overall average PER can be approximated with the average duration of the "bad" conditions.

Moreover, although not strictly necessary as simulation parameters, the average PER for each mobility group is calculated to acquire a metric of the channel quality. These values, along with the average PER of the fixed and the nomadic users, are summarized in Table II and are used in Section VI to present the performance of the proposed algorithm for various environments. Finally, to guarantee the randomness of the channel error distributions among different runs of the ns-2, the initializing variable "seed" of the ns-2 random number generator is changed before every new run. In this way, two different users always present different dynamics even if they belong to the same QoS-class and mobility-group.

VI. SIMULATION CAMPAIGN

A. Simulation Parameters Setup

The C++ simulator has been developed on the basis of an event-driven model, where an event can be either the arrival or the termination of a connection. Both new admittance requests and terminations of active connections are considered to follow Poisson distribution [26]. Thus, the time between two successive arrivals of users (inter-arrival time, denoted as τ) as well as the duration of each admitted connection (denoted as d) are exponentially distributed, with mean values $1/\lambda$ (sec^{-1}) and $1/\mu$ (sec^{-1}) correspondingly.

$$\begin{aligned} \text{pdf}(\tau) &= \lambda \cdot e^{-\lambda \cdot \tau}, \qquad \text{E}[\tau] = 1/\lambda \\ \text{pdf}(d) &= \mu \cdot e^{-\mu \cdot d}, \qquad \text{E}[d] = 1/\mu. \end{aligned} \tag{3}$$

The parameters E[d] and E[τ] along with the aggregate number of users in the network, denoted as S, determine the average traffic load of the network. (Note that S is the total number of users subscribed to the network for service, not only the active users at each instant.) In detail, in order to introduce a measure of traffic load in the network, we define L as the average traffic load that could be forwarded if the capacity of the integrated HAP-satellite system were infinite and no calls were blocked. Having assumed uniform distribution of the users among the three different QoS-classes, L (kbit/s) is given by the equation:

$$\begin{aligned} L = \left(\frac{1}{3} \cdot S\right) \cdot \frac{\text{E}[d]}{\text{E}[\tau]} \cdot 128 + \left(\frac{1}{3} \cdot S\right) \cdot \frac{\text{E}[d]}{\text{E}[\tau]} \cdot 256 \\ + \left(\frac{1}{3} \cdot S\right) \cdot \frac{\text{E}[d]}{\text{E}[\tau]} \cdot 512 = \frac{896}{3} \cdot S \cdot \frac{\text{E}[d]}{\text{E}[\tau]} \end{aligned} \tag{4}$$

where E[d] is equal to 500 s, which for 512 kbit/s nominal rate corresponds to data files with 32 Mbyte average size, while a total of 52 users are considered to be subscribed to the network. Thus, by manipulating the parameter E[τ], we determine L. The performance of the identified metrics (Section IVB) has been evaluated from four different points of view:

1) traffic load (L),
2) average PER,
3) BPR among the three QoS-classes (BPR1 and BPR2),
4) percentage of TCP users (TCPperc).

In the following subsections, performance of the TCP-driven CAC is evaluated for different values of p. The improvements introduced as well as the optimum choice of p are addressed.

B. Traffic Load

Figs. 3 and 4 present BP and AT of the system for a wide range of traffic loads, while Figs. 5 and 6 present the pdf of norm_load for L = 8400 kbit/s and L = 11000 kbit/s respectively. Uniform distribution of the users among the mobility groups and equal BP of the three QoS-classes ($\text{BPR}_1 = \text{BPR}_2 = 1$) are considered; all the users are assumed to use TCP as transport protocol (TCP_perc = 100). According to Figs. 3 and 4, the exploitation of the TCP feedback can lead to a great improvement in the network performance for the whole range of traffic conditions. Moreover, as it is expected, the higher the p, the lower the gain in BP and in AT. On the other hand, except for $p = 0$, the revenue from the network resources is maximized, and such a scenario fails to provide the agreed QoS to the users, as the probability and the extent of congestion occurrence rises with the traffic load (Figs. 5 and 6). Nevertheless, setting $p = 0.3$ guarantees the reliability of the system, as the probability that the aggregate load exceeds the network capacity is practically nullified even in cases of overload. In particular, for L = 11000 kbit/s (110 percent of the

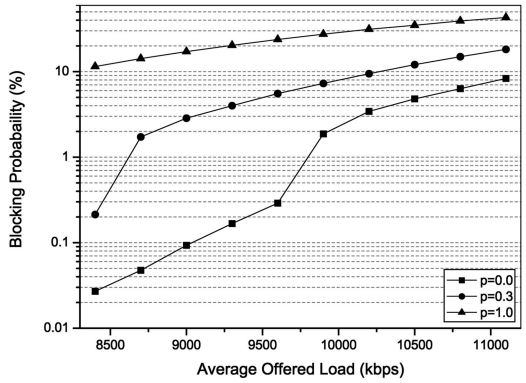


Fig. 3. Behavior of the system in terms of blocking probability for different traffic conditions (L), uniform distribution of the users among the mobility-groups, equal BP ($BPR_1 = BPR_2 = 1$), TCP_perc = 100.

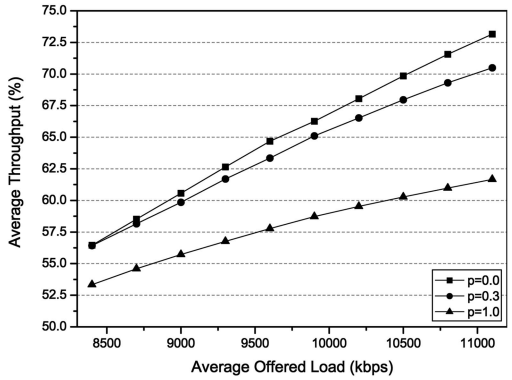


Fig. 4. Behavior of the system in terms of average throughput for different traffic conditions (L), uniform distribution of the users among the mobility-groups, equal BP ($BPR_1 = BPR_2 = 1$), TCP_perc = 100.

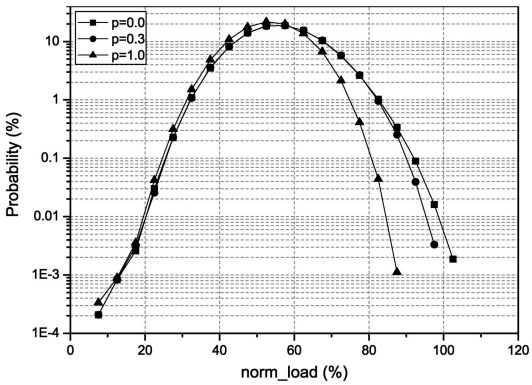


Fig. 5. PDF of norm_load for $L = 8400$ kbit/s, uniform distribution of the users among the mobility-groups, equal BP ($BPR_1 = BPR_2 = 1$), TCP_perc = 100.

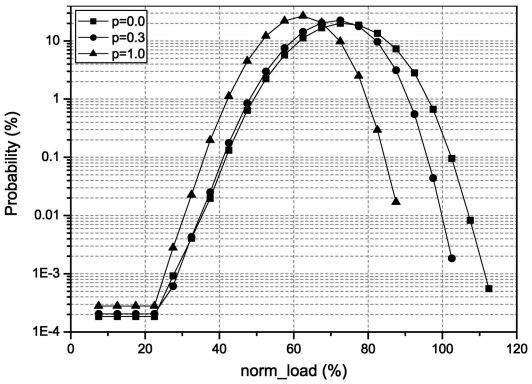


Fig. 6. PDF of norm_load for $L = 11000$ kbit/s, uniform distribution of the users among the mobility-groups, equal BP ($BPR_1 = BPR_2 = 1$), TCP_perc = 100.

system's capacity), $P[100\% = \text{norm\_load} < 105\%] \approx 2 * 10^{-3}$.

C. Average PER

As described in Section IV, different mobility groups are characterized by different channel models and consequently by different PER. According to Section V, to acquire a metric of channel performance, we calculate the average PER for each one of the five mobility groups as a meaningful parameter of the channel quality. Thus, since higher PER implies greater decrease of the TCP average window size, the ratio among the users of different mobility status plays an important role in the system performance. As a matter of fact, since

$$\begin{aligned} \text{E[PER\{fixed\}]} &< \text{E[PER\{nomadic\}]} \\ &< \text{E[PER\{mob\_high\}]} \\ &< \text{E[PER\{mob\_sub\}]} \\ &< \text{E[PER\{mob\_urb\}]} \end{aligned} \tag{5}$$

the percentage of mobile users (especially of the suburban and urban ones) is higher and the contribution of the TCP feedback to the system efficiency is more essential. Therefore, to control user distribution among the five mobility groups, a new parameter, mob_ratio, is introduced as

$$\begin{aligned} \text{mob\_ratio} &= \frac{\text{perc\{mob\_urb\}}}{\text{perc\{mob\_sub\}}} = \frac{\text{perc\{mob\_sub\}}}{\text{perc\{mob\_high\}}} \\ &= \frac{\text{perc\{mob\_high\}}}{\text{perc\{nomadic\}}} = \frac{\text{perc\{nomadic\}}}{\text{perc\{fixed\}}} \end{aligned} \tag{6}$$

where perc denotes the percentage of each mobility group in S. Moreover, mob_ratio is closely related to the average PER (av_PER) of the overall network:

$$\text{av\_PER} = \sum_{k \in K} \{\text{perc}(k) \cdot \text{E[PER}(k)]\} \tag{7}$$

$$\sum_{k \in K} \{\text{perc}(k)\} = 100 \tag{8}$$

with $k = 5$ as the number of mobility groups. Based on (6), (7), and (8) as well as on the average PER of each mobility-group (Table II), Table III shows the av_PER for a variety of mob_ratio values.

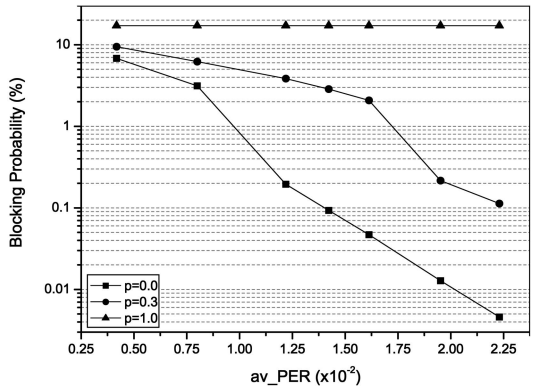


Fig. 7. BP decrease achieved by the CAC-TCP interaction for $0.5 \leq$ mob_ratio ≤ 1.5, TCP_perc = 100 and $BPR_1 = BPR_2 = 1$, traffic load equal to $L = 9000$ kbit/s.

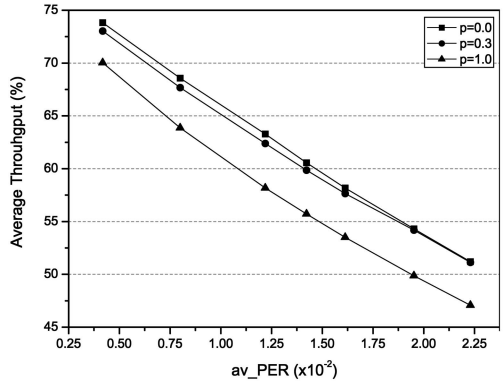


Fig. 8. AT increase achieved by the CAC-TCP interaction for $0.5 \leq$ mob_ratio ≤ 1.5, TCP_perc = 100 and $BPR_1 = BPR_2 = 1$, traffic load equal to $L = 9000$ kbit/s.

TABLE III
Average PER as Function of the mob_ratio Parameter

mob_ratio	av_PER
0.5	0.00418065
0.7	0.0079925
0.9	0.0121912
1	0.01422
1.1	0.0161314
1.3	0.0195159
1.5	0.0223014

Figs. 7 and 8 present the BP decrease and the AT raise that are achieved by the CAC-TCP interaction, for $0.5 \leq$ mob_ratio ≤ 1.5. TCP_perc = 100 and $BPR_1 = BPR_2 = 1$, while the traffic load is considered equal to $L = 9000$ kbit/s. On the basis of these figures, the average PER (higher mob_ratio) is higher, i.e., the reception conditions are harsher, making it, more important to take into account the TCP feedback in the CAC procedure. On the contrary, low average rate networks set a more challenging environment for the ability of the algorithm to adapt to the variations in data rate requirements (Figs. 9 and 10). Specifically, in contrast to high average PER scenarios, the system working point is closer to its

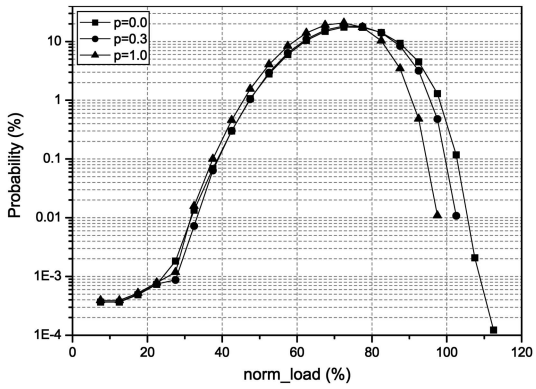


Fig. 9. PDF of norm_load for mob_ratio = 0.5, $L = 9000$ kbit/s, equal BP (BPR1 = BPR2 = 1), TCP_perc = 100.

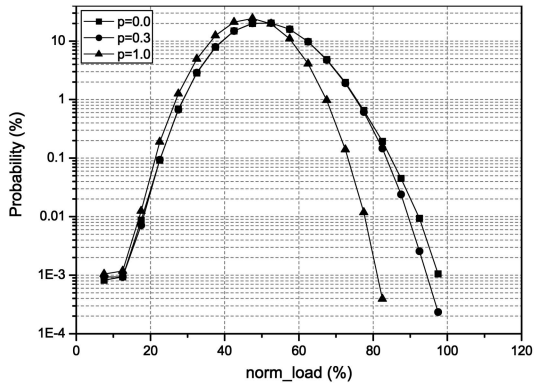


Fig. 10. PDF of norm_load for mob_ratio = 1.5, $L = 9000$ kbit/s, equal BP ($BPR_1 = BPR_2 = 1$), TCP_perc = 100.

capacity (Fig. 8) due to the fact that the average data rate of the flows approximates their nominal data rate; thus even a short-scale, unforeseen growth of bandwidth demand can lead to congestion. For p equal to 0.3 the desirable tradeoff regardless of the overall communication conditions is achieved.

D. Blocking Probability Ratio Among the QoS-Classes

Under conditions of a specific PER, the higher the nominal data rate of a connection, the greater the degradation (measured in kbit/s) that the connection speed experiences. Thus, assigning larger proportions of the network capacity to QoS-classes with higher data rate requirements, the redundancy of assigned bandwidth increases. Consequently, the implementation of our proposed scheme maximizes the revenue for networks serving high data rate users.

These conclusions are evident in Figs. 11 and 12 where system performance is evaluated for $0.5 \leq BPR_1 = BPR_2 \leq 1.5$. TCP_perc = 100, mob_ratio = 1 and $L = 9000$ kbit/s. Moreover, according to Fig. 12, high data rate connections leave unused larger proportions of the system capacity (lower AT), which can compensate for any abrupt changes in their transmission rate. Therefore, under such circumstances, less stringent anti-congestion

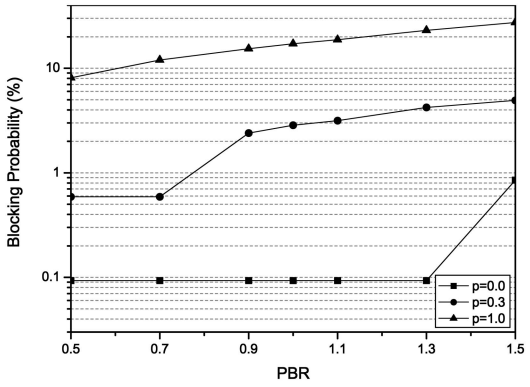


Fig. 11. System performance in terms of BP for $0.5 \leq BPR_1 = BPR_2 \leq 1.5$, TCP_perc = 100, mob_ratio = 1, and L = 9000 kbit/s.

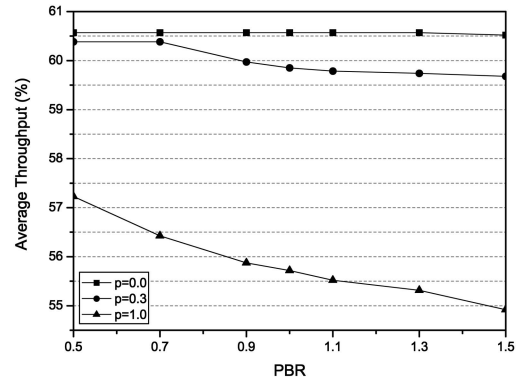


Fig. 12. System performance in terms of AT for $0.5 \leq BPR_1 = BPR_2 \leq 1.5$, TCP_perc = 100, mob_ratio = 1, and L = 9000 kbit/s.

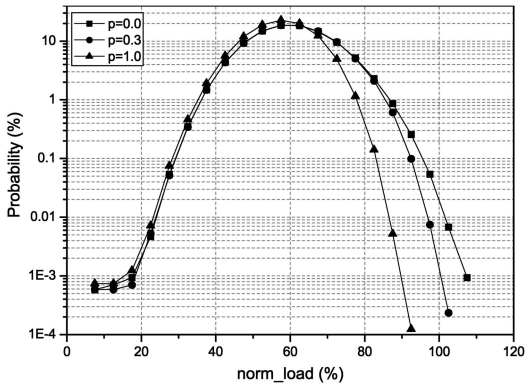


Fig. 13. PDF of norm_load for $BPR_1 = BPR_2 = 0.5$, L = 9000 kbit/s, mob_ratio = 1, TCP_perc = 100.

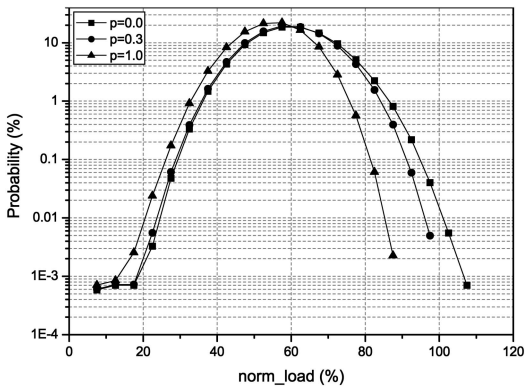


Fig. 14. PDF of norm_load for $BPR_1 = BPR_2 = 1.5$, L = 9000 kbit/s, mob_ratio = 1, TCP_perc = 100.

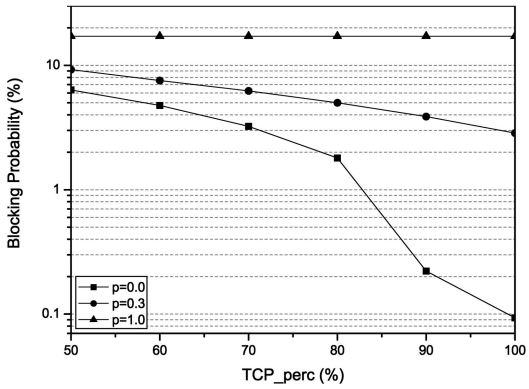


Fig. 15. BP versus percentage of TCP connections for different TCP_perc values, $BPR_1 = BPR_2 = 1$, mob_ratio = 1, L = 9000 kbit/s.

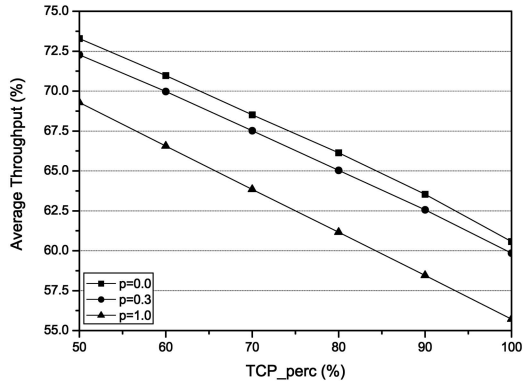


Fig. 16. AT versus percentage of TCP connections for different TCP_perc values, $BPR_1 = BPR_2 = 1$, mob_ratio = 1, L = 9000 kbit/s.

mechanisms (lower safety margin) are required in order to preserve the users QoS. Fig. 13 and Fig. 14 provide the pdf of the norm_load for BPR1 = BPR2 equal to 0.5 and 0.1 respectively.

E. Percentage of TCP Users

Since the CAC-TCP interaction is applied only under the hypothesis of TCP-driven communication, the virtue of such a scheme is rather dependent on the percentage of the network occupancy by TCP users. In this respect, Figs. 15 and 16 present the BP and AT of the system, for a variety of TCP_perc values. $BPR_1 = BPR_2 = 1$, mob_ratio = 1 and L = 9000 kbit/s.

Similar to the analysis made for av_PER and BPR, the less the TCP users there are, the higher the occupancy of the channel, as the non-TCP users retain their nominal data rate throughout the duration of their connection. Finally, Fig. 17 and Fig. 18 show the

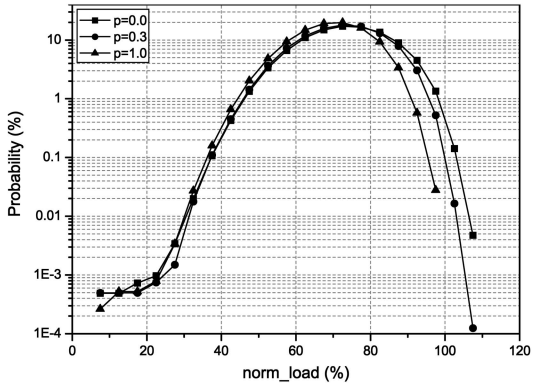


Fig. 17. PDF of norm_load for TCP_perc = 50, L = 9000 kbit/s, mob_ratio = 1, $BPR_1 = BPR_2 = 1$.

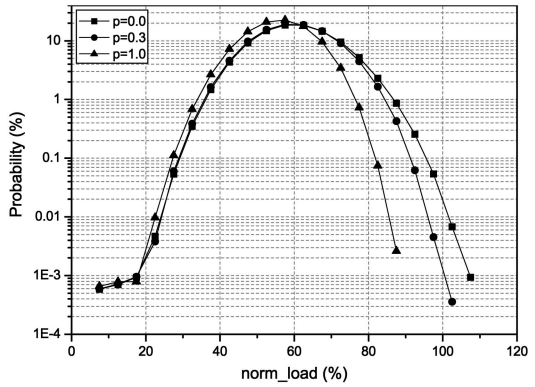


Fig. 18. PDF of norm_load for TCP_perc = 100, L = 9000 kbit/s, mob_ratio = 1, BPR1 = BPR2 = 1.

pdf of the normalized aggregated load in two cases: TCP traffic is the 50 percent of the overall traffic (Fig. 17), and all the traffic is TCP-based (Fig. 18).

VII. CONCLUSIONS

The combined use of HAP and satellite represents an innovative and challenging architecture to guarantee telecommunication broadband services even if terrestrial infrastructures are unavailable.

The implementation of a CAC algorithm can greatly help in ensuring QoS for multimedia services. The use of applications running TCP as transport protocol present degraded performance due to large bandwidth-delay product and to the presence of transmission errors. In this paper, we proposed a new CAC algorithm that aims to optimize the resource utilization by using inputs from transport layer.

Through simulations, we demonstrated a considerable improvement of the performance (in terms of AT and BP), with respect to a basic CAC algorithm that takes into account only the QoS requirements of the connections, ignoring the possibility that the users may fail to utilize all the bandwidth assigned to them.

REFERENCES

[1] Partridge, C., and Shepard, T.
TCP performance over satellite links.
IEEE Network, **11**, 5 (1997), 44–49.

[2] Ruhai, W., and Horan, S.
Impact of Van Jacobson header compression on TCP/IP throughput performance over lossy space channels.
IEEE Transactions on Aerospace and Electronic Systems, **41**, 2 (Apr. 2005), 681–692.

[3] Cianca, E., Prasad, R., De Sanctis, M., De Luise, A., Antonini, M., Teotino, D., and Ruggieri, M.
Integrated satellite-HAP system.
IEEE Communications Magazine, **43**, 12 (Dec. 2005), 33–39.

[4] Tozer, T. C., and Grace, D.
High-altitude platforms for wireless communications.
Electronics & Communication Engineering Journal, **13**, 3 (June 2001), 127–137.

[5] Karapantazis, S., and Pavlidou, N. F.
Broadband communications via high-altitude platforms: A survey.
IEEE Communications Surveys & Tutorials, **7**, 1 (2005), 2–31.

[6] Sujit, P. B., and Ghose, D.
Search using multiple UAVs with flight time constraints.
IEEE Transactions on Aerospace and Electronic Systems, **40**, 2 (Apr. 2004), 491–509.

[7] Elmastry, G. F., Russell, B., and McCann, C. J.
Enhancing TCP and CAC performance through detecting radio blockage at the plain text side.
IEEE Military Communications Conference 2005 (MILCOM 2005), Oct. 2005, 1–5.

[8] Brown, K., and Singh, S.
ATCP: TCP for mobile cellular networks.
ACM Computer Communications Review, **27**, 5 (1997), 19–43.

[9] Wang, X., Eun, D. Y., and Wang, W.
A TCP-aware call admission control scheme for packet-switched wireless networks.
Presented at the 25th IEEE Performance, Computing, and Communications Conference, Apr. 2006.

[10] Roseti, C., Theodoridis, G., Luglio, M., and Pavlidou, N. F.
TCP driven CAC scheme for HAPS and satellite integrated scenario.
In *Proceedings 1st International Workshop on High Altitude Platform Systems* (WHAPS), Athens, Greece, Sept. 2005.

[11] http://www.isi.edu/nsnam/ns/ns-documentation.html.

[12] Perros, H. G., and Elsayed, K. M.
Call admission control schemes: A review.
IEEE Communications Magazine, **34**, 11 (Nov. 1996), 82–91.

[13] Shiomoto, K., Yamanaka, N., and Takahashi, T.
Overview of measurement-based connection admission control methods in ATM networks.
IEEE Communications Surveys, (1999).

[14] Gibbens, R. J., Kelly, F. P., and Key, P. B.
A decision-theoretic approach to call admission control in ATM networks.
IEEE Journal on Selected Areas in Communications, **13**, 6 (Aug. 1995).

[15] Dimitriou, N,. Tafazolli, R., and Sfikas, G.
Quality of service for multimedia CDMA.
IEEE Communications Magazine, **38**, 7 (July 2000), 88–94.

[16] Niyato, D., and Hossain, E.
Call admission control for QoS provisioning in 4G wireless networks: Issues and approaches.
Network IEEE, **19**, 5 (Sept.–Oct. 2005), v5–11.

[17] Stevens, W.
TCP/IP Illustrated, vol. 1.
Reading, MA: Addison Wesley, 1994.

[18] Luglio, M., Roseti, C., and Gerla, M.
The impact of efficient flow control and OS features on TCP performance over satellite links.
ASSI Satellite Communication Letter (Sat-Comm Letter), (9th ed.), special issue on multimedia satellite communication, **III**, 1 (2004), 1–9.

[19] Stevens, W.
TCP slow start, congestion avoidance, fast retransmit, and fast recovery algorithms.
Internet RFC 2001, (1997).

[20] Jacobson, V.
Congestion avoidance and control.
ACM Computer Communications Review, **18**, 4 (Aug. 1988), 314–329.

[21] Lakshman, T., and Madhow, U.
The performance of TCP/IP for networks with high bandwidth-delay products and random loss.
IEEE/ACM Transactions on Networking, **5**, 3 (1997), 336–350.

[22] Evans, B. G., Mazzella, M., Corazza, G. E., Polydoros, A., Mertzanis, I., Philippopoulos, P., and De Win, W.
Service scenarios and system architecture for satellite UMTS IP based network (SATN).
In *Proceedings of AIAA 20th International Communications Satellite Systems Conference*, Montreal, Canada, May 12–15, 2002.

[23] Miura, R., and Oodo, M.
Wireless communications system using stratospheric platforms—R&D program on telecom and broadcasting system using high altitude platform stations.
Journal of the Communications Research Laboratory, **48**, 4 (2001), 33–48.

[24] Grace, D., Thornton, J., Konefal, T., Spillard, C., and Tozer, T. C.
Broadband communications from high altitude platforms—The helinet solution.
In *Proceedings of Wireless Personal Mobile Conference* (WPMC 2001), vol. 1, Aalborg, Denmark, Sept. 9–12, 2001, 75–80.

[25] Chini, P., Giambene, G., Bartolini, D., Luglio, M., and Roseti, C.
Dynamic resource allocation based on a TCP-MAC cross-layer approach for interactive satellite networks.
International Journal of Satellite Communications & Networking (special issue for cross-layer protocols for satellite communication networks), **24**, 4 (Sept. 2006), 367–385.

[26] Epstein, B. M., and Schwartz, M.
Predictive QoS-based admission control for multiclass traffic in cellular wireless networks.
IEEE Journal on Selected Areas in Communications, **18**, 3 (Mar. 2000), 523–534.

[27] Lutz, E., Cygan, D., Dippold, M., Dolainsky, F., and Papke, W.
The land mobile satellite communication channel—Recording, statistics, and channel model.
IEEE Transactions on Vehicular Technology, **40**, 2 (May 1991).

[28] Recommendation ITU-R P.681-6.

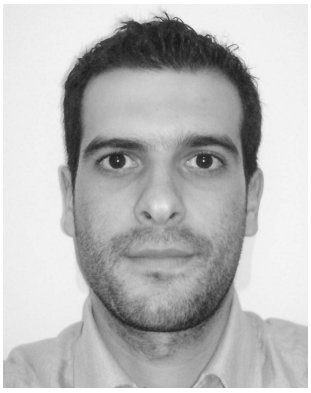
[29] Cuevas-Ruiz, J. L., and Delgado-Penin, J. A.
Channel model based on semi-Markovian processes. An approach for HAPS systems.
14th International Conference on Electronics, Communications and Computers, Feb. 2004, 52–56.

Luglio Michele received the Laurea degree in electronic engineering at the University of Rome "Tor Vergata." He received the Ph.D. degree in telecommunications in 1994.

From August to December 1992 he worked, as visiting staff engineering at Microwave Technology and Systems Division of Comsat Laboratories, Clarksburg, MD.

From 1995 to 2004 he was a research and teaching assistant at the University of Rome "Tor Vergata." At present he is associate professor of telecommunication at the same university. He works on designing satellite systems for multimedia services both mobile and fixed, in the frame of projects funded by EC, ESA and ASI. He taught Signal Theory and collaborated in teaching Digital Signal Processing and Elements of Telecommunications. In 2001 and 2002 he was a visiting professor at the Computer Science department of the University of California Los Angeles (UCLA) and taught a satellite networks class. Now he teaches satellite telecommunications and signals and transmission.

Dr. Luglio received the Young Scientist Award from ISSSE '95.



Georgios Theodoridis received his diploma in electrical and computer engineering from Aristotle University of Thessaloniki, Greece, in 2004. He is currently working towards his Ph.D. degree in the same department.

His research interests are in the field of call admission control and radio resource management in wireless terrestrial and satellite/HAP networks. He is involved in Greek and European projects in these fields.

He is a member of the Technical Chamber of Greece.

Cesare Roseti graduated cum laude in 2003 in telecommunication engineering at the University of Rome "Tor Vergata." He received the Ph.D. degree in space systems and technologies in 2007.

In 2003 and 2004, he was a visiting student in the Computer Science Department of the University of California, Los Angeles (UCLA). From August to December 2005 he worked at the TEC-SWS division of the European Space Agency, Noordwijk, The Netherlands. He is currently a research fellow at the University of Rome "Tor Vergata," teaching in the Laboratory of Signal Processing and collaborating in both satellite telecommunications and signal processing classes. His research interests include satellite communications and protocol design, cross-layer interactions, security, HAPS/UAV communications, protocol implementation and performance analysis in wired/wireless networks. He is involved in national and international projects in these fields.

Fotini-Niovi Pavlidou received a Ph.D. degree in electrical engineering from Aristotle University of Thessaloniki, Greece, in 1988 and a diploma in mechanical-electrical engineering in 1979 from the same institution.

She is currently a full professor at the Department of Electrical and Computer Engineering at Aristotle University, teaching in the undergraduate and post-graduate program in the areas of mobile communications and telecommunications networks. Her research interests are in the field of mobile and personal communications, satellite and HAP communications, multiple access systems, routing and traffic flow in networks, and QoS studies for multimedia applications over the Internet. She is involved in many national and international projects in these areas.

Dr. Pavlidou chaired the European COST262 Action on Spread Spectrum Techniques. She has served as a member of the TPC of many IEEE/IEE conferences. She is a permanent reviewer for many international journals. She has published about 80 papers in refereed journals and conferences. She is currently chairing the joint IEEE VTS & AESS Chapter in Greece.