DISTRIBUTED MEASUREMENTS AND EXPERIMENTS IN SATELLITE COMMUNICATIONS AND NETWORKING: THE SATNEX EXPERIMENTAL PLATFORM[■]

CNIT (IT) (de Cola T., Pecorella T., Ronga L.)
CNR-ISTI (IT) (Barsocchi P., Chessa S., Ferro E., Gotta A., Oligeri G., Potortì F., Secchi R.)
University of Aberdeen (UK) (Sathiaseelan A.)
University of Barcellona (ES) (Castro V. M, Peral R. J.)
University of Bologna (IT) (Caini C., Firrincieli R.)

Abstract

One of the aims of the SatNEx (Satellite Network of Excellence) European project is to provide the scientific community that operates on satellite communications with simulation/emulation tools and real measurement campaigns, in order to support various research activities that investigate problems in satellite communications at different levels of the OSI stack. In particular, one among the various joint activities (JAs) in SatNEx is devoted to "Research tools, trials and testbeds", and, within this, one group of activities concentrates on "Access, network, and transport layer trials" (JA 2410). This paper addresses the results obtained just within this joint activity, thus presenting both the real experiments carried out on a satellite platform, and describing the specific testbeds developed for this aim. The high number of authors of this paper is an indication of the strong collaborative work sustained, the large measurement campaigns, and the complexity in setting up some tests. In the SatNEx project we envisaged a quite complex reference scenario, where particular research aspects needed to be sustained by measurements and adequate tools of investigation. The research activity developed in SatNEx addresses the most relevant aspects of satellite communications, such as access schemes, advanced networking, end-to-end quality of service, transport performance, and deployment of new applications. In this view, specific integration issues among heterogeneous networks, encompassing satellite and wireless segments, were taken into account. The testbeds developed allowed the study of interplanetary environments, as well. In this specific case, attention was put in the communication and networking challenges arising in the design of a telecommunication infrastructure suitable to convey data derived from remote stations, located on planets, and addressed to Earth data gathering centres.

1. INTRODUCTION

The increasingly raising demand for broadband multimedia services to be offered seamlessly and ubiquitously to both mobile and fixed users has highlighted over the last years the role jointly played by satellite and wireless technologies, which are consolidated as excellence means to enable communications anywhere and anytime

---

[■] Work funded by the European Community in the framework of the FP6 SatNEx NoE project. The project developed in two phases: SatNEx I and II, contract N. 507052 and N. 027393, respectively. The second phase is still in progress. Please visit www.satnex.org.

[1]. From this point of view, the integration of telecommunication infrastructures built upon satellite and terrestrial wireless technologies has been assuming a primary importance [2] and pushing different companies to promote and offer quadruple-play services (high-speed Internet access, television, and telephone, over wireless technologies) to mobile, as well as fixed, terminals, thus extending the range of interested customers. In particular, this necessity has turned out in translating typical statically cabled areas into highly dynamic communities, able to make phone calls, browse Internet pages, and receive TV video streams, by exploiting Beyond Third Generation (B3G) technologies, reconfiguration capabilities of mobile ad-hoc networks (MANETs) [3] - such as vehicular area networks (VANs) [4]- and the large coverage offered by satellites. As a consequence, the communication paradigm has been rapidly evolving and, therefore, changing the frontiers of the Internet and the services currently transported, owing to improved transmission techniques and more effective protocol architectures. In particular, the edges of the Internet are being extended also to interplanetary networks, given the high interest expressed towards space exploration and the large number of space missions.

It is straightforward to see that this scenario offers interesting perspectives not only to companies that sell multimedia and broadband services, but also to the scientific community, due to the research challenges that may rise in terms of both communication and networking design; in this philosophy, the research activities performed by the SatNEx community perfectly fit. More precisely, SatNEx's goal not only is to provide the theoretical background necessary to properly investigate all the aforementioned communication and networking issues, but also to validate the efficacy of protocol candidates identified during the preliminary study phase. Such a task drives our attention to conceiving tools and testbeds suitable to experimenting some of the most interesting aspects of Satellite-Terrestrial Wireless communications.

In practice, while addressing validation tasks, different viable approaches can be adopted in terms of simulation, emulation, and trials on the field, which differentiate one from another in terms of realisation complexity and degree of real environment "reproducibility". Given the increasing complexity of current communication networks and the inherent need of experimenting novel protocol architectures, the adoption of *simulation* tools is a viable solution, because the advantages offered in terms of limited software complexity, flexibility, and ease of implementation are remarkable. Significant examples are offered by general purpose simulation frameworks, such as Ns-2, OPNET, and OMNET++, just o cite a few, along with a large number of other simulation tools, designed to assess the performance of data communication achieved in specific environments, such as wireless networks. Despite the low complexity, by contrast, simulation imposes the constraint of modelling a great portion of the network environment under investigation, and this implies the approximation of the real world.

Opposite to simulation is the *emulation* concept that relies upon using devices configured and properly interconnected in order to approximate the real environment behaviour as much as possible. From the point of view of satellite communications, an emulation tool helps to more deeply investigate the dynamics of data communications, since real traffic is transported through a satellite emulated scenario, traversing real devices, such as switches and routers. It is straightforward that the adoption of such an approach implies a complexity, in terms of hardware configuration and software implementation, quite close to trials on the field, as real devices are concerned. Obviously, emulation facilities are not limited to satellites, but reasonably also apply to wireless, provided that an increase in complexity due to the larger number of involved nodes (wireless environments are usually composed of a multitude of nodes) does not affect the whole system reliability.

As far as *real experimentation* is concerned, the advantages offered by a testbed are immediate, because no modelling task has to be performed, and real traffic traverses the satellite/wireless segments, thus allowing inferring the problems characterising the satellite communications. The major drawback is due to the economical costs of the satellite platforms, which also include the need of borrowing bandwidth resources from a satellite service provider. In second instance, the implementation complexity grows if compared with the other two approaches, since proposals of novel medium access schemes or on-board processing capacities cannot be tested without a great deal of effort, differently from the simulation/emulation tools that allow contemplating such features.

This paper basically concentrates on the emulation approaches and the real measurement campaigns developed up to now in the SatNEx project, as they are attractive for the design and configuration issues they raise. Indeed, within the SatNEx project many research activities have been and currently are carried out by means of simulations as well; in most cases, however, simulation software is proprietary and written just for specific purposes. Basically, the simulation field is so wide that it would be meaningless to focus our attention on all the tools actually employed within SatNEx. On the contrary, we prefer to emphasize the presentation of the validation methodologies, specifically finalised to the evaluation of protocols and architectures involved in satellite data communications. Eventually, this approach, hence relying on simulation as well as emulation and trials on the field, will allow us to give some insights into real systems performance.

Given the high complexity of satellite communications, the experimentation framework can be suitably subdivided into two different parts: the *reference scenario* and the *protocol architecture validation* (Fig. 1). The former is more closely related to the characterisation of the peculiarities of the environment under study, in order to point out the main factors that could affect the performance offered by the protocols used to perform data communications. The latter addresses the protocol and architecture performance aspects, and it is actually composed by two main layers. At the bottom, access, network, and transport issues are merged together in a unique framework, in order to evaluate how configurations of parameters and transmission schemes may impact on some protocols' performance, by operating at different layers. This implies cross-layer investigation and optimisation, as demanded for in the most advanced terrestrial wireless and satellite systems. At the top, the application is actually conceived to define services (e.g., audio, video conference) to be handled by the protocol entities that work in the underlying layers. In this case, the experimentation mainly focuses on the "quality" of video/ audio services and on the effectiveness of the signalling mechanisms, thereby applied to establish the multimedia conference session.

The experimentation activity we present has been carried out by means of suitable investigation and measurement tools, specifically devised to allow for accurate analysis of several aspects in the field of satellite communications, as pointed out later on. In fact, the methodology adopted to conduct tests was two-fold and reflected the peculiarities and features offered by the different tools: on the one hand, attention has been paid to measurement campaigns, while on the other hand the use of emulation tools and their integration with real platforms was taken as a reference. In the first case, we describe the experimental environment, the measurement campaign itself, and the gross results we obtained, without entering the specific results, which are described in a more specific set of papers; in the second case, we describe the integrated investigation tools, composed of emulation testbeds and real platforms, the facilities they offer, and only briefly we present some results we obtained, as an example of utilization.

The paper is organized as follows. Section 2 introduces the reference scenario and the conceptual segments over which experimentation campaigns have been conducted, namely wireless, satellite, and deep space. Section 3 is devoted to the experimental satellite platform involved in most of the experiments presented in the rest of the paper. Section 4 addresses the measurement experiments performed over wireless and satellite segments, while Section 5 presents the integrated investigation tools that were devised to analyse performance figures offered by protocols and architectures applied in complex satellite environments. Finally, Section 6 draws the conclusions of this work and indicates possible extensions for the future.

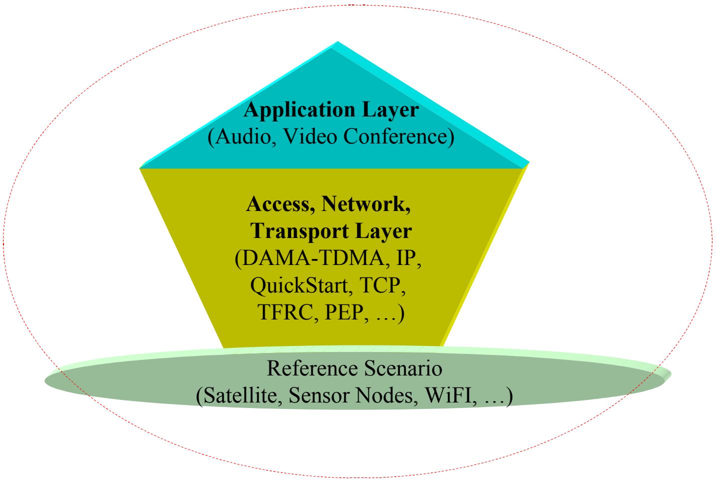


Figure 1. The Global Experimentation Framework, enclosed in the circle.

2. THE REFERENCE SCENARIO

The focus of the experimentation envisioned in the SatNEx project is on heterogeneous networks, with a particular attention to the role of satellites, seen as fundamental means to guarantee widespread coverage and broadband multimedia communications. Satellite links have to ensure interconnectivity among networked islands, which can be built upon cabled and wireless technologies. As mobile users require more and more services, the wireless technology has been widely adopted, for its intrinsic capability of connecting people at moderate costs. From this point of view, we consider wireless access, based on WiFi, 2G, 3G, and 4G [3], which offers moderate bandwidth for transmitting and receiving multimedia and data; it takes advantage of satellite connections. This general view completes by considering also the extension of common Internet infrastructures to involve interplanetary links, aimed at providing connectivity also towards deep space applications. On this side, links connecting local Earth data gathering centres and remote stations, located on

Mars or the Moon, are assumed. In particular, due to the high relevance given to the exploration of planets, the case of remote interplanetary sensing applications, performed by sensor nodes and remotely controlled by Earth stations, is taken as a reference in this work, as regards this particular networking segment. The global scenario investigated is shown in Fig. 2.

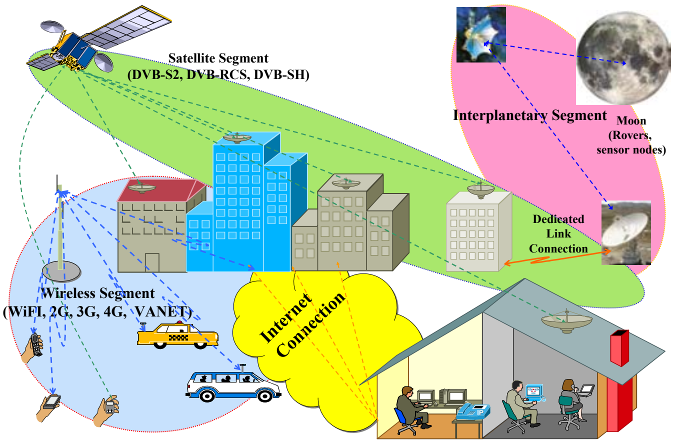


Figure 2. The reference scenario investigated.

The characterization of the reference environment has to be completed by also considering the aspects that the experimentation is finalized at looking into. As previously introduced, performance investigation is mainly concentrated on the protocol architectures involved, as well as the services that are expected to be effectively transported over the network under analysis. As far as protocols and architectures are involved, a particular note has to be reserved to some solutions, applied at network and transport layers, suited to dealing with multimedia and data efficiently. In this perspective, we perform the analysis of the performance offered by protocols (some versions of TCP), when handling either data or multimedia, by taking into account also the specifics of the transmission means, in terms of physical peculiarities and medium access schemes (*VOTOS experiment*). The analysis of protocols and architectures completes by also considering the application layer and putting emphasis on services. The experiment presented (*VoIP over satellite experiment*) exclusively addresses VoIP applications running on different platforms, by focusing on the signalling solutions adopted to establish the conference sessions, and, more important, on typical metrics employed while analyzing quality of service issues (e.g., delay, jitter, etc.). In a heterogeneous environment, where wireless and satellite links work together for carrying multimedia data, the physical characterization of the investigated environment is important. This is achieved by means of a measurement framework that lies transversal with respect to the aforementioned ones, and influences all of them, as protocol effectiveness strictly relies upon environment characteristics (*WICHMO experiment*). The previous experiments have been carried out by means of real measurement campaigns.

In addition to research activities performing measurement campaigns, the development of specific tools suitable to performance evaluation was addressed as well by defining two experiments (*EGGS* and *UCIT*), more focused on the aspects related to the implementation and the facilities they can offer.

The former was developed for studying some specific aspects of interplanetary links, and it is focused on the Delay Tolerant Network (DTN) architecture; the latter for the evaluation of advanced transport protocol and architectures over heterogeneous networks, which includes satellite links. EGGS, dealing with deep space communications, for which performing real measurements is impossible or too limiting, relies on an emulation of the links characteristics. UCIT, addressing satellite communications, can either rely on emulated channels or real satellite platforms.

3. THE REAL SATELLITE PLATFORM

In all the experiments involving a real satellite platform, the Skyplex DATA platform based on the Skyplex OBP (On-Board Processing) technology run by Eutelsat on the HotBird 6 satellite [5] was used. It implements DVB-RCS features, while not being fully DVB-RCS compliant. It carries four Ka band transponders equipped with a Skyplex Unit; the units are composed of channels configurable in Low Rate (at 2.112 Mbit/s) or High Rate (at 6.226 Mbit/s) mode. The channels can be configured in SCPC (Single Channel Per Carrier) or TDMA (Time Division Multiple Access) mode, capacity sharing only being available in the second case. In TDMA mode, the frame structure is composed by N time slots, occupied by bursts; each burst contains M MPEG cells, thus $N \times M$ cells per frame are hosted. In such a structure, N also represents the number of users per frame and defines the Skyplex TDMA Mode. In order to avoid the heavy constraint that only up to N users can share the channel (i.e., one burst per frame assigned to each terminal) the frame structure has been extended by grouping together L frames in a multi-frame. Assigning, as a minimum, one burst per multi-frame to each user, the number of users per channel becomes $N \times L$, L being the low rate, with a minimum granularity in our system of 44 Kbit/s (N=6, M=8, L=8) instead of 352 Kbit/s (N=6, M=8, L=1) for a single frame structure. In fact, the burst configuration present on the satellite platform referenced in this paper corresponds to N=6, M=8, L=8. Finally, three multi-frames constitute a super-frame defined for signalling purposes. In our configuration, the super frame length results to be 820 ms long, which is the period of retransmission and validity of the schedule for the assignment of the multi-frame time slots to the terminals (Burst Time Plan - BTP).
Bandwidth assignment can be either dynamic (Bandwidth on Demand, BoD), or static (Committed Information Rate, CIR) or mixed. In the first case, each terminal periodically requires the bandwidth on the basis of its own instantaneous need, and the time slots are assigned in a best-effort mode. The BoD assignment does not provide any priority among terminals or any quality of service. In case of static allocation, a fixed number of time slots is pre-configured and assigned to a particular terminal.

4. THE MEASUREMENT EXPERIMENTS: VOTOS, VoIP OVER SATELLITE, WICHMO

4.1. VOTOS: Description of the experiment

When satellite links interconnect terrestrial wireless networks, TCP dynamics are dominated by link errors, whose negative effects are amplified by the large delay-bandwidth product. The aim of the VOTOS (VariOus TCPs on Satellite) experiment is to analyse the performance of TCP congestion control over satellite DAMA (Demand Allocation Multiple Access) systems.

Part of VOTOS [6] consisted in evaluating how standard New Reno TCP fares when compared with several other TCP variants, such as SACK [7] and Westwood+ [8], a sender-only modification of TCP. The steady-state behaviour of the Linux versions of these TCP flavours were studied in [6] over the Skyplex testbed, in a scenario that combines a satellite channel introducing a long delay and a WLAN segment experiencing channel errors. All versions have been tested in pairs (one connection per type), over the same path, in order to verify the inter-protocol fairness

A second part of VOTOS deals with the *startup phase* of TCP, as opposed to steady-state. With the diffusion of DAMA satellite systems, congestion control algorithms (such as TCP, TFRC, etc.) are increasingly going to experiment poor performance at startup, because the Slow Start phase is slowed down by DAMA. In fact, in rate-based DAMA systems, the bandwidth is assigned on the basis of the stations' requests, which, in turn, depend on the current transmission rates. Requests from the stations undergo a 250 ms propagation delay on geostationary satellite networks, and the allocation needs an additional 250 ms to be broadcast to the stations. This means that assignments are always late with respect to incoming traffic by at least 500 ms, to which significant management overheads are usually added. Since the throughput in the Slow Start phase typically increases at each RTT (Round Trip Time), the allocated bandwidth is always less than the offered traffic, which accounts for the very long Slow Start phase we observed in DAMA systems [9].

A third part of VOTOS considers possible solutions for the *slow startup* when multimedia traffic is involved. With the advent of audiovisual streaming on the Internet, congestion control mechanisms, such as TCP Friendly Rate Control (TFRC) [10], will be mandatory for multimedia traffic. Since these algorithms emulate the behaviour of TCP during the start-up phase, performance degradation similar to that of TCP is observed on DAMA links for these protocols. This means that the startup slowdown on DAMA systems affects both short TCP connections, like web browsing, and streaming applications using TFRC (TCP-Friendly Rate Control) in the first few seconds of streaming, the most critical for fast switching between channels.

Other than using the Skyplex platform, we studied the interaction of Quick Start with TFRC and DAMA by using the ns-2 simulator. Since no DAMA support is available in ns-2, we developed a module for simulating a DAMA controlled satellite network [11]. The testbed was used to validate the DAMA module for the ns-2 simulation.

4.1.1. The results

TCP in steady state. First results are relative to fairness measured in steady-state conditions. We can make some observations on the measurements, shown in Table I. First, Westwood+ with SACK exhibited a more aggressive behaviour versus New Reno with SACK (trial #4) with respect to *goodput* in quasi error-free conditions. In trial #3, Westwood+ appeared to be more robust than New Reno with respect to FER (Frame Error Rate), with a higher goodput and less timeouts. In trials #1 and #2, which exhibited the most severe fading conditions, New Reno with SACK achieved the highest normalized goodput. New Reno SACK, the most diffused TCP flavour, was thus able to provide good performance in its Linux version. Westwood+ SACK appeared to provide slightly better performance.

TCP start-up. These results are relative to the start-up phase of a single TCP connection over a lossless satellite hop. Figure 3 shows the first 20s of acknowledged sequence numbers for the Linux implementations of the TCP stack, reporting 9 connection traces. The same happens with FreeBSD's TCP, though Linux is quicker because it sets the initial window to three packets (according to RFC 3390) while FreeBSD more conservatively uses an initial window of one packet only.

Table I. Mean and Median of: RTT, Congestion Window, and Normalized Goodput; Wi-Fi Frame Error Rate; Number of Time Outs (T.O.)

Trial Number	TCP variant	Wireless FER	Mean, Median			Number of timeouts
			RTT [s]	CWND [MSU=1500B]	Normalized Goodput	
1	New Reno	$2.6\cdot10^{-3}$	1.56, 1.23	59.6, 50.2	0.32, 0.38	16
	New Reno SACK	$3.3\cdot10^{-3}$	2.03, 1.37	119.0, 63.0	0.39, 0.42	11
2	Westwood+	$4.6\cdot10^{-3}$	1.83, 1.24	101.0, 54.0	0.31, 0.33	21
	Westwood+ SACK	$3.9\cdot10^{-3}$	1.72, 1.09	78.0, 45.0	0.32, 0.32	12
3	New Reno	$1.4\cdot10^{-3}$	1.90, 1.30	58.9, 55.0	0.27, 0.28	21
	Westwood+	$1.1\cdot10^{-3}$	1.65, 1.34	58.3, 56.0	0.30, 0.28	11
4	New Reno SACK	$3.1\cdot10^{-4}$	3.99, 3.99	196, 170	0.46, 0.45	1
	Westwood+ SACK	$4.3\cdot10^{-4}$	4.05, 4.10	213, 224	0.50, 0.52	2

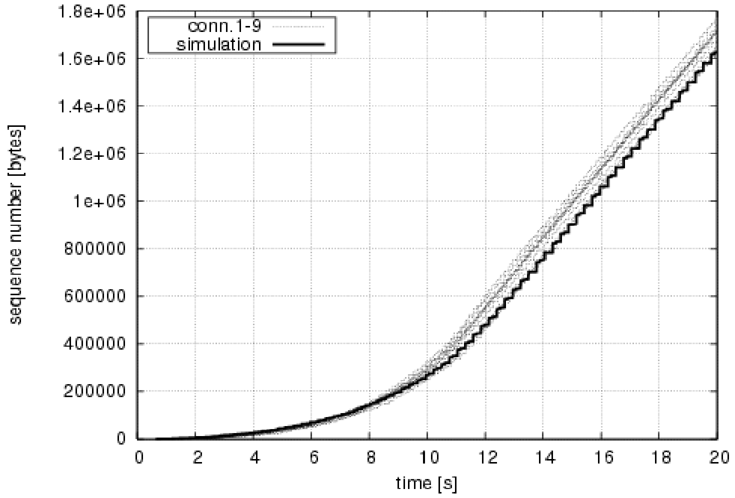


Figure 3. Startup behaviour for Linux TCP.

We can distinguish two phases: an approximately exponential increase in sequence number followed by a linear increase phase. Indeed, since during the Slow Start phase the congestion window is increased by twice the amount of acknowledged segments, the ACK reception rate increases exponentially until the TCP throughput reaches the available channel capacity; after that point, ACK rate remains constant. However, we noticed that the time to double the congestion window is considerably longer than the connection round trip time usually observed over terrestrial networks. This phenomenon should be attributed to the DAMA scheme adopted to share the satellite bandwidth.

To describe the interaction between the DAMA bandwidth assignment and the TCP startup dynamics, a simple approximate analytical model of TCP startup has been developed in [9]. This model points out that the Slow Start evolution in the presence of DAMA is equivalent to the one that would occur without DAMA, but with a longer RTT. In the Skyplex scenario, we assessed that the throughput of the system during Slow Start would be the same if the whole channel bandwidth were permanently assigned to the users, but the propagation delay were about 520 ms rather than the real delay of about 250 ms. This means that short TCP connections which never reach steady state, such as most HTTP requests, are considerably slower on a DAMA allocation system than they are on a fixed-assignment one.

Ns-2 DAMA module. In order to better understand the interactions between DAMA and TCP, we developed a DAMA simulation module for ns-2. The Skyplex platform was used to validate the model by comparing simulated traces with real traffic ones. In particular, we set up a unicast connection between two hosts, running Linux 2.6, one located at CNR-ISTI in Pisa and the other one at the CNIT Research Unit in Pisa, as well. In order to carry out measurements of one-way delay, we synchronized the two hosts by using the Network Time Protocol (NTP), which provided an accuracy of 10 ms for the delay estimation.

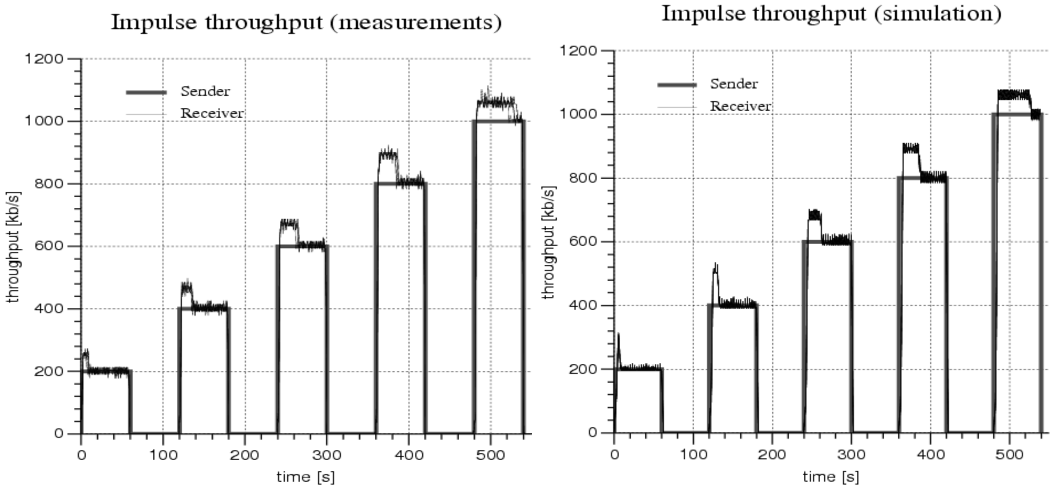


Fig. 4. Measured and simulated throughput with offered impulse traffic.

A total of eight simulation runs, each with a different offset of the bandwidth request with respect to the beginning of the TDMA frame (thus covering all possible cases), were performed. Figure 4 shows the response of the Skyplex network to impulses of UDP traffic in terms of throughput. Graphs are depicted for both ns-2 simulation results and real trials. As detailed in [11], the response to an offered traffic impulse is a throughput transient with length depending on the DAMA parameters and amplitude proportional to the pulse height. Comparison of measured data with simulated ones yields excellent agreement.

Improving the startup behaviour on DAMA. As discussed above, both experiments and simulation confirmed that the interaction of DAMA with the TCP startup phase causes significant performance degradation. One possible approach to reduce it [12] is using the QS (Quick-Start) mechanism [13]. QS is an IETF (Internet Engineering Task Force) experimental protocol designed to provide lightweight signalling of the level of congestion (specifically available capacity) between routers and a pair of communicating end hosts.

Figure 5 illustrates the dynamics of end-to-end delay during the first 40s of a 1024 Kbit/s CBR TFRC flow. The vertical bars indicate the span of delay over all possible offsets of TCP starting time with respect to DVB frame, while the solid lines are median delays. The QS mechanism is successful in reducing the end-to-end delay and in speeding up TFRC during the slow-start phase [14]. Indeed, in this case the median delay was reduced by nearly 40% and the median delay required in order to complete the startup phase reduced from 35s to 25s. This allows not only to reduce the size of de-jitter buffers, but also to reduce the transient phase, thus leading to considerable increase in performance. Quick-Start is consequently a promising mechanism to be coupled with future TFRC-enabled networks.

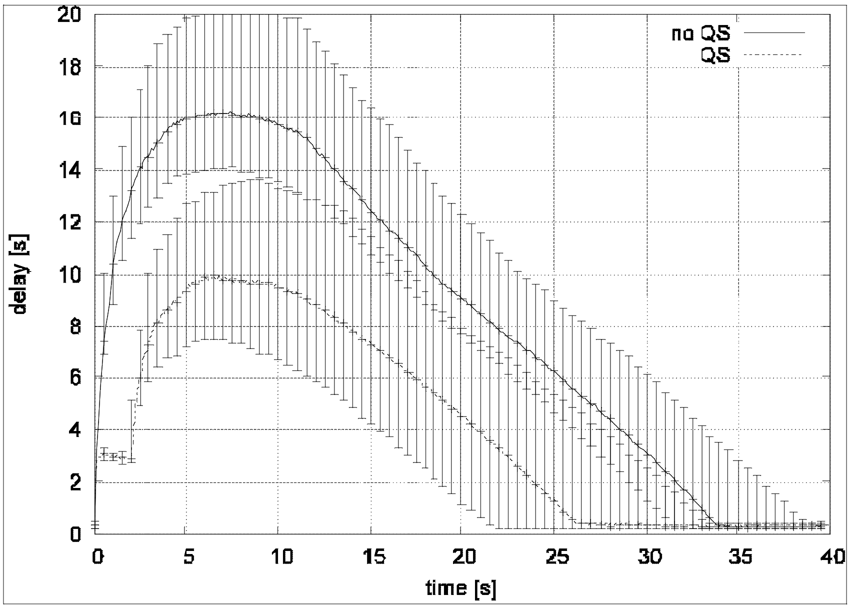


Figure 5. End-to-end delay of a single TFRC connection over simulated Skyplex link.

4.2. VoIP over Satellite: Description of the experiment

The VoIP experiment is oriented to the exploration of all the relevant issues of the usage of IP-oriented voice services over geostationary satellites. The measured indexes show a relationship between quality metrics and the resulting user perception, when different configurations, voice codecs, and services are adopted.

The following VoIP services have been evaluated over the satellite link:

- single VoIP call among remote users,
- Multiple VoIP calls among remote users,
- VoIP Conferencing with mixed population (local and remote).

The experiments have been conducted with various satellite network load, in order to inspect the impact of the combination of large delays and various packet loss rates.

The basic testbed adopted in the experiments is shown in Fig. 6.

A local Software VoIP PABX was installed in each satellite site. This choice was due to the requirement of coexistence between site local and remote voice services. The experiments collected statistical data of the user perception on the call quality and other objective performance indexes, such as packet arrival times and jitter, packet loss, bandwidth usage and signalling (connection and tear-down) events.

The software used can be divided into 3 main categories: 1) VoIP PABX; 2) Soft Phones; 3) Traffic analysis.

Among the VoIP PABX software we tested three typical solutions, as follows:

1) Linux Debian + Asterisk [15, 16]
2) Trixbox [17]

3) AsteriskNOW beta5 [18]

The three server systems are very similar, as they are all built around the Asterisk VoIP PABX server system, but some important differences must be noted. The first solution leads to the highest degree of configuration on what is running on the server, but also to the highest probability of mis-configuration and software conflicts. Trixbox is a well-known integrated solution for Asterisk, with auto-installation features. However, its components are not fully integrated; hence, for instance, accounting management and billing operations can be cumbersome at first. The user interface, moreover, is not uniform among Trixbox parts.

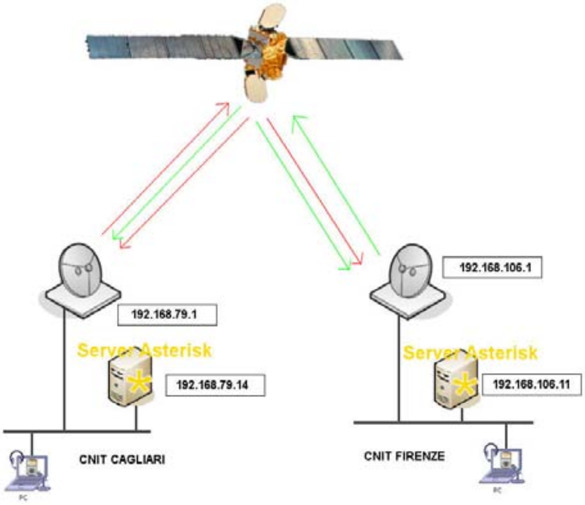


Figure 6. VoIP: basic experimental testbed.

AsteriskNOW beta5 is the latest product from Digium (makers of Asterisk) and, though still in beta stage, it seems to be the most powerful, easy and clean web-based interface for Asterisk. AsteriskNow comes with self-installing features and a customized Linux OS (rPath Linux, [19]). Due to the built-in feature set available in AsteriskNOW (e.g., conference calls, web-based management, and hardware devices integration), all the software tests have been made on this platform.

The SoftPhones used are of two kinds, according to the protocol adopted. A number of SoftPhones have been evaluated, but the results have not shown any real audio quality difference among them. The main differences are the UIs, the codec availability and the protocols supported (SIP / IAX). For the tests two SoftPhones have been selected, namely, Xlite [20] and Idefisk [21], shown in Fig. 7. Finally the system's performance (delay, jitter, and bandwidth) was analyzed by using WireShark (ex-Ethereal) [22]. It should be noted that WireShark can trap SIP (Session Initiation Protocol) and RTP (Real-Time Protocol) packets, and can provide statistics based on SIP and RTP fields, but cannot perform similar operations with IAX, due to its binary in-channel signalling structure.

Figure 7. VoIP: SoftPhones used in the tests.

4.2.1. The results

A set of experiments has been carried out with both G.711u and GSM codecs. Results show that the G.711u average bandwidth occupation is 69.02 Kbit/s with a standard deviation of 18.34 Kbit/s. The distribution of the bandwidth occupancy is shown in Fig. 8. This is particularly interesting for DBA (Dynamic Bandwidth Assignment) systems, since the non-constant bandwidth can trigger different DBA requests. On the contrary the GSM codec bandwidth usage is almost constant due to the missing silence suppression feature.

As concerns the propagation delays and jitter, the latter has been estimated by (1), in order to overcome the lack of a tight synchronization between sender and receiver:

$$J_i = J_{i-1} + \left(\left|D(i-1,i)\right| - J_{i-1}\right)/16 \qquad (1)$$

where J_i is the jitter measure available at the time of arrival of the i-th RTP packet (same arguments hold for J_{i-1}), and D(i,j) is:

$$D(i,j) = (R_j - R_i) - (S_j - S_i) = (R_j - S_j) - (R_i - S_i) \qquad (2)$$

where R_i is the time of reception of the i-th packet, and S_i its RTP timestamp. This method produces an exponentially averaged de-trend on arrival times, resulting in a good approximation of effective jitter. The measured jitter introduced by the network has an average of 33.1 ms and a standard deviation of 4.59 ms; this is perfectly compatible with the VoIP QoS requirements.

Sets of experiments have been also carried out in order to compare IAX (see [23] and following) and SIP [24] data channel performance differences, shown in Fig. 9. No significant difference has been noted between the two protocols, even in the presence of background traffic to fill the available bandwidth. It should be observed that the results do not include traffic multiplexing and/or SIP signalling channel synchronization. In the presence of interfering UDP traffic, the sum of queuing and propagation delays produce a detrimental effect on call quality, as shown in Table II, which summarizes the MOS rating of the sample of users with different codecs and UDP interfering traffic. As shown in the table, with a shared satellite capacity of 1.4 Mbit/s, the voice call cannot be initiated (marked as X in the table) at signalling level with an interfering traffic saturating the link. With 1.2 Mbit/s, the SIP protocol connects the terminals but the voice signals are impaired (marked as LAGGED in the table). With 1.1 Mbit/s of UDP traffic and less, the call quality improves with moderate differences among the selected codecs.

Table II. MOS rating with different codecs and UDP interfering traffic

	1.4 Mbit/s	1.2 Mbit/s	1.1 Mbit/s	0.75 Mbit/s	0.35 Mbit/s
GMS	X	LAGGED	2	2/3	3
G.711a	X	LAGGED	2/3*	3	4
iLBC	X	LAGGED	2/3	3	3/4

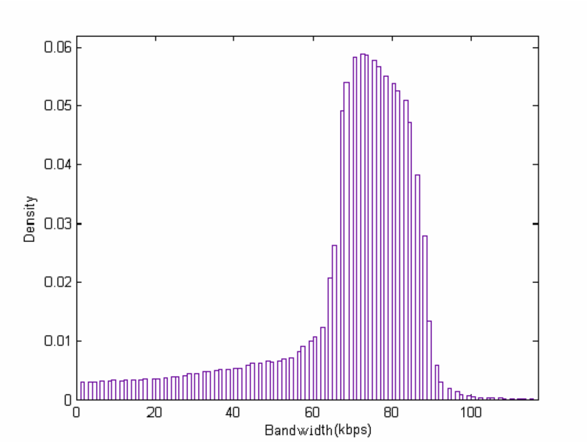
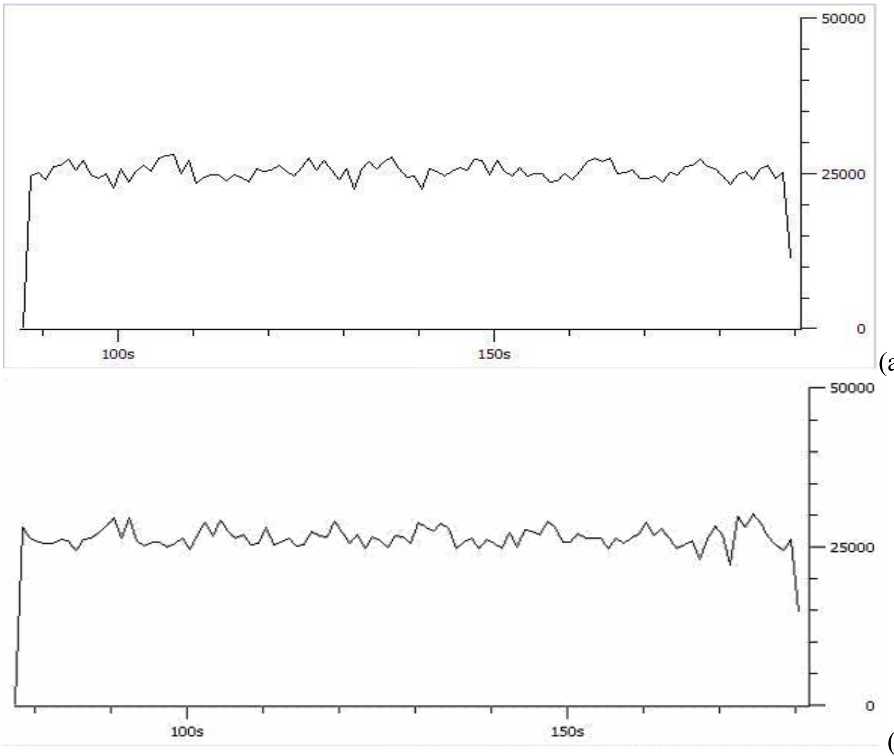


Figure 8. VoIP: bandwidth occupancy, G.711u codec.



(a)

(b

Figure 9. VoIP: SIP (9.a) and IAX (9.b) bandwidth occupancy (Y axis, expressed in bit/s) versus time (X axis). GSM codec

4.3. WICHMO: Description of the experiment

The WICHMO (WIreless CHannel MOdelling) experiment considers a hybrid network with a Wi-Fi link at the network edge and a satellite link somewhere in the network core. Examples of scenarios where this configuration is realistic are ships or airplanes where Internet access on board is provided through a Wi-Fi access point and a satellite link with a geostationary satellite [25]. The serialisation of terrestrial and satellite wireless links is problematic from the point of view of a number of applications, be they based on video streaming, interactive audio or TCP. The reason is the combination of high latency, caused by the geostationary satellite link, and frequent, correlated packet losses caused by the local wireless terrestrial link [26].

In WICHMO we concentrate on frame error models targeted to investigate the performance of TCP-based applications on such hybrid networks. Since TCP interprets packet loss as a sign of congestion, thus slowing its pace to avoid worsening the network conditions, frame errors due to corruption cause decreased throughput, and this effect is worsened by high path latency. Various techniques exist that tackle this problem [27], but no definitive answers yet.

While errors are rare on the satellite segment, an error model is required for the terrestrial segment. A complete packet error model for the Wi-Fi link requires three levels of operation: a *frame error* model at the raw channel level, an implementation of the Wi-Fi *ARQ* (automatic retransmission) mechanism, and an implementation of the vendor-dependent *speed switching* mechanism. The first level gives a statistical representation of the frame error process on the raw channel due to either bad preamble acquisition or bad CRC due to corrupted bits in the MAC protocol data unit. The second level implements ARQ as defined in the IEEE 802.11 standard, whereas a sender retransmits a frame up to a configurable number of times – typically 7 – or until it receives an acknowledgement. The third level implements an adaptive coding and modulation scheme (ACM), by which the sender may switch to a different transmission speed when it perceives a change of the channel conditions.

WICHMO tackles the first two levels by choosing a statistical frame error model that is adequate for TCP on a hybrid channel with fixed-speed Wi-Fi (no ACM). To this end, we run a simulation using real traces of frame errors on a Wi-Fi channel; we then rerun simulation using synthetic traces of frame errors produced by commonly used models, and compare the results. WICHMO additionally considers the third level (i.e. ACM) on a Wi-Fi-only connection: integration of the latter results with the former ones is a matter of future research.

4.3.1. *The hardware/software used*

We obtained traces of the packet losses both at the CNR-ISTI Institute in Pisa (IT) and at the Engineering School at the Universitat Autónoma de Barcelona (ES).

Pisa experiments were concerned with the frame error process with ARQ but no ACM. In Pisa, we connected a Wi-Fi ad hoc network to the Skyplex satellite system to test the feasibility of the simulated experiment, which we did successfully. We then considered the terrestrial and satellite segments separately in order to obtain parameters for the simulation.

As far as the satellite link is concerned, we considered a 2 Mbit/s bandwidth with a latency of 300 ms, a transmit buffer sized as the bandwidth-delay product, which is 150 Kbytes, and a Bernoullian channel with a Frame Error Rate (FER) of 10^{-5}, which in our case is essentially the same as errorless [28].

As far as the Wi-Fi link is concerned, we used two laptops equipped with a Debian operating system and PCMCIA Wi-Fi interfaces configured in ad hoc mode. Measurements were carried out in an office where thin concrete walls delimit the rooms. The transmitter sent a flow of 1000-byte unicast frames spaced by 5 ms, at a fixed rate of 11 Mbit/s, with fragmentation and retransmission disabled. The receiver checked the sequence number inside the frames and kept a trace of the lost ones. Since the channel occupancy of a frame is about 1.2 ms, this kind of measure traces the wireless indoor channel conditions quite accurately.

Barcelona experiments were concerned with ACM performance, so both retransmissions (ARQ) and adaptation to channel (ACM) were enabled. Measurements were carried out at a floor with corridors and classrooms. We measured the packet losses for different channel states, each state being defined as a range of the SNR (Signal to Noise Ratio). The dynamics of the slowly changing signal strength has been modelled in the literature as a memoryless Markov chain [29]. Within each state, we measured the packet loss rate by using an Intel Pro Wireless 2200BG card. We measured the channel in several representative scenarios in order to obtain a general indoor channel model.

4.3.2. *The results: comparison of traces with commonly used packet error models*

Results relative to frame errors and ARQ without ACM were obtained in Pisa by simulating the hybrid network on ns-2 and comparing results obtained with real frame error traces measured on the Wi-Fi link with results obtained using a frame error model, in order to choose the most useful Wi-Fi frame error model for TCP over hybrid networks.

Results relative to ACM, which were obtained in Barcelona, show the influence of ACM over Wi-Fi performance: integration of these results in the hybrid scenario will be the subject of future research.

TCP performance on synthetic versus measured packet loss. The most common frame error model used for Wi-Fi channels is the Bernoullian process (sometimes called a *Poisson* process), which is defined by a single parameter. A slightly more complex model requiring two parameters defines the channel as being in one of two states, namely a *good* state, where all frames are successfully received, and a *bad* state, where all frames are lost because of corruption. We refer to the simple good-bad model as a *bistable* model.

Both models have geometrically decaying correlations. However, the frame error traces we measured exhibit different statistics, which is the main ground for this investigation. Figure 10 shows the mean lengths of bursts and gaps versus FER as observed in our measurements, both of which may be significant in the context of the Pisa experiments. To simulate the behaviour of TCP in hybrid networks made up of a satellite link and a Wi-Fi link, without ACM, we used the ns-2 simulator with Ttem, a purposely-written error model that reads a trace of the channel state, applies the IEEE 802.11 ARQ algorithm, and discards packets that cannot be received after a given number of retries, 7 in our case. By using Ttem, we could measure what the performance of Wi-Fi is on the channel for which we have real frame error measurements and compare it with channels with synthetic frame errors (Table III).

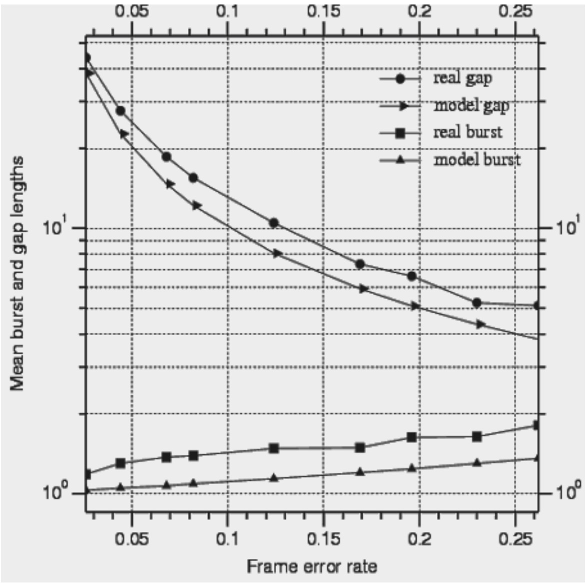


Figure 10. Mean burst and gap length distributions for the measured data and for a Bernoullian channel with the same mean frame error rate.

We run 10 instances of a single TCP connection for 2000 s on the simulated hybrid network [28]. To test the performance of an error model we plot the single-connection TCP goodput versus mean FER on the raw channel (that is, frame errors before ARQ); we do so by using the model, and compare the resulted graph with the one obtained using measured data. In other words, we compare the goodput obtained when frame errors are generated according to Bernoulli and bistable models with the goodput obtained when frame errors are those measured in the real indoor scenario. Taking the steady-state TCP single-connection goodput as a performance measure, we see how well different statistical models fit the measured frame error traces. We consider four different frame error models in addition to the real traces, as shown in Table III.

Table III. Traces used in the simulation experiment

(a)	Bernoulli process having the same frame error rate as the real traces
(b)	Bernoulli process having the same mean burst length as the real traces
(c)	Bernoulli process having the same mean gap length as the real traces
(d)	bistable process having the same frame error rate, mean burst length and mean gap length as the real traces
(e)	the real traces

When using a Bernoulli process as the frame error model, one needs to tune a single parameter to fit the model to observed data; that is, one should choose a significant statistical parameter of the observed data and use its value to generate the Bernoulli process. The choice of such a parameter is not obvious. In our case, the simplest choice of a parameter is the mean FER. However, there are some good reasons why choosing a different parameter could be wiser. One candidate as an alternative to the FER is the mean burst length, an important parameter with respect to ARQ performance: the longer the error bursts, the higher the probability that ARQ cannot recover a lost frame. Another candidate is the mean gap length, which is related to TCP performance. As observed in [30], TCP performance is higher for higher burstiness of the segment error process, because the congestion window has a higher probability of becoming big if gaps are long. In fact, the measured error process has both longer bursts and longer gaps than a Bernoulli process with the same FER.

The main result of the Pisa experiment is the plot in Fig. 11, where the goodput of TCP on the hybrid network is plotted versus the frame error rate of the *real traces* of the Wi-Fi segment. The error bars indicate the minimum, mean and maximum goodput values. Looking at Fig. 11 we can see that the simplest frame error process does not adequately model TCP's goodput on the hybrid network; moreover, the behaviour of the curves is different, as the curve that refers to the real case (case (e) in Table III), is convex, while (a) is not. By using the mean burst length as the parameter of interest gives the same behaviour as the real traces, as it can be seen by comparing cases (b) and (e), but the values are very different. As discussed above, this observation is a hint that the behaviour of ARQ is dominant in the chosen performance measure, which is the TCP goodput. On the other hand, the gap length appears to be much less important, as using it as the parameter for tuning the Bernoulli process does not yield satisfactory results: the trace for the (c) case is the one farthest from (e). One reason could be that, as shown in Fig. 10, a Bernoulli process with the same mean gap length as the real traces has a lower FER than the real traces. Case (d) is the most similar to (e): the bistable (thus using two parameters instead of one) model best fits the chosen performance measure and is our recommended choice.

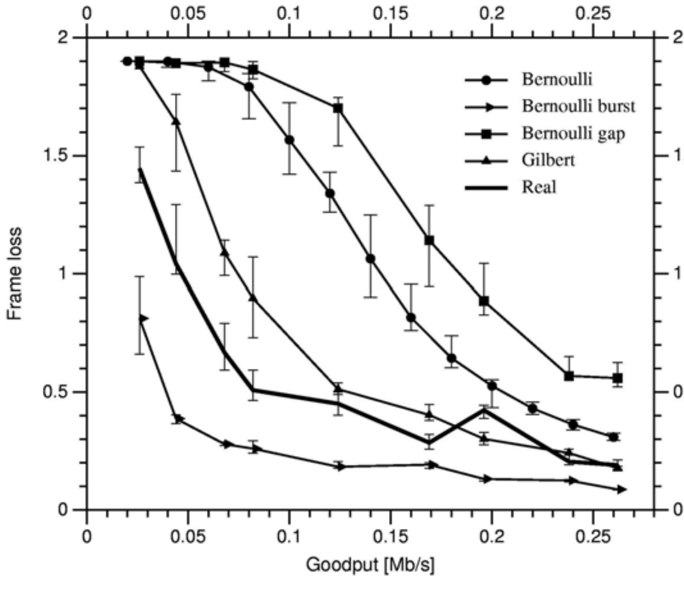


Figure 11. TCP goodput versus frame error rate of real traces for different error models.

Adaptive Coding and Modulation effect on the hybrid channel. In the Barcelona experiment, we analysed the third level of a packet error model, i.e. the effect of the performance of the adaptive modulation and coding (ACM) scheme of the WLAN card, this time independently of the satellite link. All members of the IEEE 802.11 WLAN family (802.11a/b/g) provide multi-rate capabilities. To achieve a high performance under varying conditions, these devices dynamically adapt their transmission rate according to the channel variations. While this rate adaptation algorithm is a critical component in the overall system performance and it also is a critical component of the accuracy of a packet error model, only very few algorithms have been published and the implementation issues associated with these mechanisms are hardly publicly available.

Figure 12 shows the measured nested channel states of the received power and the packet loss within each state, respectively. Note that during the snapshot shown the channel only experienced the states under shadow conditions, 2 and 3. Figure 13 shows the comparison of mean error burst length probability without applying rate adaptation and applying two different algorithms: ARF [31] and LD-ARF [29]. It is interesting to point out the non-intuitive results that are shown. When rate adaptation is applied, the rate is increased after having received a pre-defined number of successful transmissions, whereas it is decreased by a pre-defined number of unsuccessful transmissions or measured performance threshold. This means that adaptation to bad channel conditions results into lower bit rate and therefore long error bursts. On the other hand, adaptation to good channel conditions results in high bit rate implying short successful burst duration. Moreover, the better the performance of the adaptation algorithm the shorter the successful bursts. Clearly, shorter bursts with rate adaptation imply higher throughput than longer bursts without rate adaptation. The effect on throughput can be easily quantified but requires different cases and scenarios to be defined and cannot be shown here for space limitations.

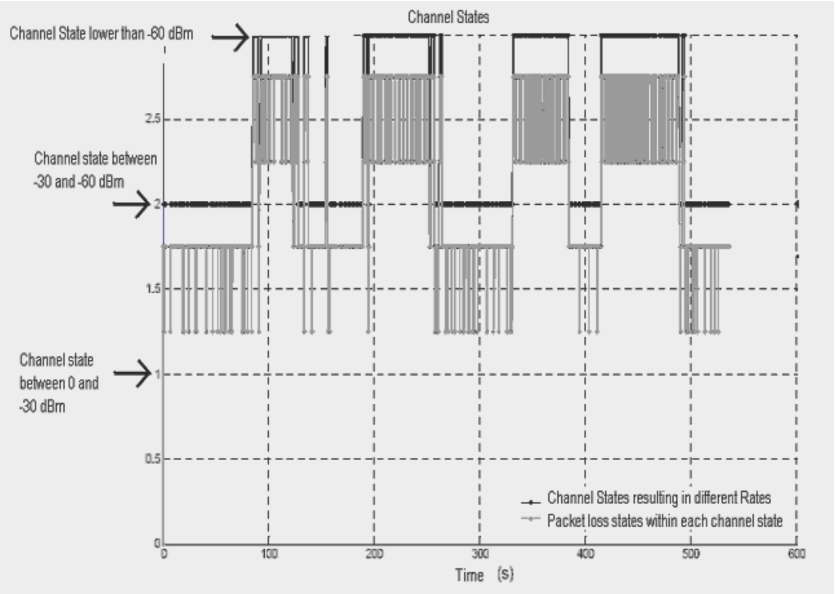


Figure 12. SNR states and packet loss within each state. Channel III, II and I correspond to SNR lower than -60 dBm, between -60 and -30 and above -30 dBm respectively.

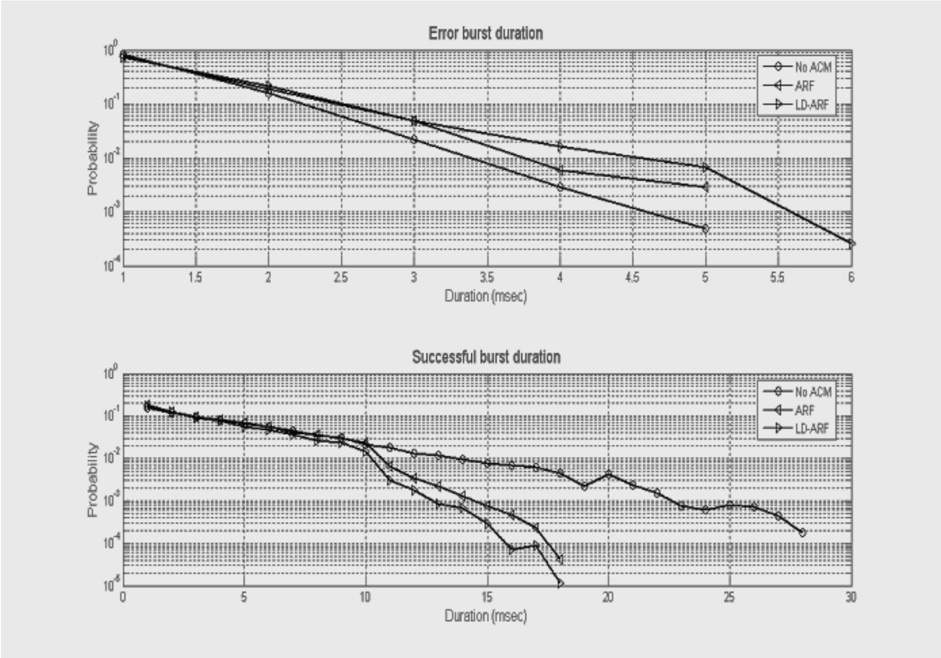


Figure 13. Mean error and successful burst length probability when ARF and LD-ARF are considered compared to the case where no rate adaptation is applied.

5. THE INTEGRATED TESTBEDS: UCIT, EGGS

5.1. UCIT: a tool for performance evaluation

The challenges exhibited by satellite communications, in terms of large bandwidth-delay product and link errors, have been thoroughly explored over the last decade. In addition the extension of this environment as to include also wired and wireless links, has fostered the scientific community towards the definition of a large number of transport protocols and architecture solutions tailored to this larger scenario. Here, we limit ourselves to cite TCP Hybla [32] and TCP-Westwood [8] as transport protocols, even if this could be further extended. An alternative approach, widely adopted in satellite environments, is represented by “accelerators” or PEPs (Performance Enhancing Proxies) [33], which basically rely on the introduction of intermediate agents at transport layer. In a future perspective, the Delay/Disruptive Tolerant Networking (DTN) architecture [34] may represent an interesting solution not only for deep space communications, but also for the most challenging satellite networks.
Given this complex context, it is paramount to design and set up effective tools to comparatively evaluate the proposed solutions in a controlled testing environment. For this purpose, CNIT and the University of Bologna (UoB) decided to integrate their assets and expertise to realize a common testbed platform for performance evaluation at the Transport layer, called UoB-CNIT Integrated Testbed, hereafter referred to as *UCIT*. Here, we focus on the description of this testbed, referring, whenever necessary, to the configuration recently used to carry out a series of joint experiments in cooperation with the Network Research Lab at UCLA Computer Science Department.

5.1.1. Testbed Overview

The testbed logical layout, sketched in Fig. 14, aims at reproducing a heterogeneous network, where satellite connections have to compete for network resources with wired connections. Of course, it is also possible to study a pure satellite environment, by simply disabling background terrestrial components. Satellite connections are composed of wired legs and a final satellite link, while TCP background traffic is present only in the entirely wired paths. All the TCP variants available in the Linux kernel are supported, as well as the FreeBSD implementation of Westwood (directly provided by UCLA). Satellite and wired connections share the Router 1-Router 2 link, whose bandwidth can be limited in order to study congestion effects. The satellite link can be realized either by emulation or by means of any real satellite system, as done in the joint UoB-CNIT-UCLA measurement campaign, where the Skyplex platform was exploited. Finally, the testbed allows the user to assess the performance of alternative solutions, based on satellite accelerators or on the DTN architecture.

5.1.2. Testbed Components

To describe in more details the testbed components, let us refer to the much more complex physical layout of the testbed, in the configuration used during the recent joint experiments' campaign (Fig. 15). The following components can be distinguished:

- a testbed controller, located at UoB;
- the UoB TATPA testbed (Testbed on Advanced Transport Protocols and Architecture) [35], temporarily relocated in Genoa,
- a TCP satellite receiver (one Linux PC) in Naples and the CNIT Skyplex GEO satellite platform [5], which links the testbed core with the satellite receiver. We will examine the main components separately.

5.1.3. The Testbed Controller

The controller, connected to the TATPA core via a Virtual Private Network, allows the user a ubiquitous remote control of the testbed via a standard web browser. A dedicated PC hosts the web server and the control software engine, developed in PHP and based on a MySQL database. The aim of the TATPA web interfaces and management control system is to facilitate the shared use of the testbed. In particular, they allowed us to hide the software and hardware complexity related to the configuration, synchronization and utilization of all testbed elements. Moreover, the web controller provides an increased level of security, as users cannot access testbed components directly, and greatly speeds up both test execution and result collection.

As an example, the configuration page of Router 1 is given in Fig. 16. Analogous interfaces are available for all the testbed entities. They can be opened by simply double clicking on entity icons on the main configuration page.

5.1.4. The TATPA Core

The TATPA core consists of several PCs running the Linux operating system (kernel 2.6.20), and a PC running FreeBSD, which is directly maintained by UCLA. Multi-TCP [36] on Linux senders implements the full version of TCP Hybla (including packet spacing and Hoe's initial ssthresh estimation) and introduces powerful log functions, essential for an in-depth analysis of results. In particular, they allow to retrieve meaningful indications about the TCP dynamics, by capturing the values of the internal TCP variables (e.g., congestion window *cwnd*, slow start threshold *ssthresh* and retransmission timeout *RTO*) It is worth noting that the internal TCP variables are extracted directly from the kernel core, since common traffic analysis tools (e.g., tcpdump) cannot derive values of TCP variables.

Router 1 can follow either a DT (Drop Tail) policy or a RED (Random Early Detection) policy (see Fig. 16). The PEPsal package [37] is installed on the Router 2 to evaluate the performance of satellite accelerators. Finally, to study DTN architectures, DTN agents have been mounted on both the satellite sender and receiver, as well as on the Router 2. DTN performance can be evaluated by means of DTNperf application [38].

5.1.5. Data Analysis

Owing to the features offered by the aforementioned three components of the testbed, a user is allowed to specify and run tests aimed at assessing the performance of TCP/UDP traffic flows and, alternatively, of DTN architecture applied over heterogeneous satellite networks. The tests consist in transferring data for a time duration specified by the user at the beginning of the trials. Transfer of data is managed by the testbed core transparently to the user and performed by means of either Iperf or DTNperf tools, in dependence on the specific protocol architecture under investigation. After collecting results through the integrated testbed, they have to be processed and carefully examined, in order to draw general conclusions not only about the effectiveness of different solutions, but also about the mechanisms that lead to different performance. To this end, the capacity of analyzing also the internal TCP variables is instrumental.

To show the features offered by the integrated testbed, let us consider the example given in Fig. 17, which refers to the data transfer performed in an interval of 200s by means of TCP Hybla. Only one satellite connection is considered active (no background wired traffic). The graph depicted in Fig. 17 can be obtained automatically by starting from the log files made available by the web interface. It is immediate to see common TCP performance indicators: the time series of the segments sent are reported, together with the cwnd and the ssthresh, all collected through the log-functions of the MultiTCP package. We can observe the very fast re-opening of the cwnd after loss events that characterize this TCP variant.

The UoB-CNIT integrated testbed is an important result of the integration activities carried out in SatNEx. The integration of assets of different partners was instrumental to obtain a powerful tool for performance evaluation of transport protocols and architecture. It was used in two joint UoB-CNIT-UCLA measurement campaigns. An excerpt of the results obtained in the first campaign is reported in [39].

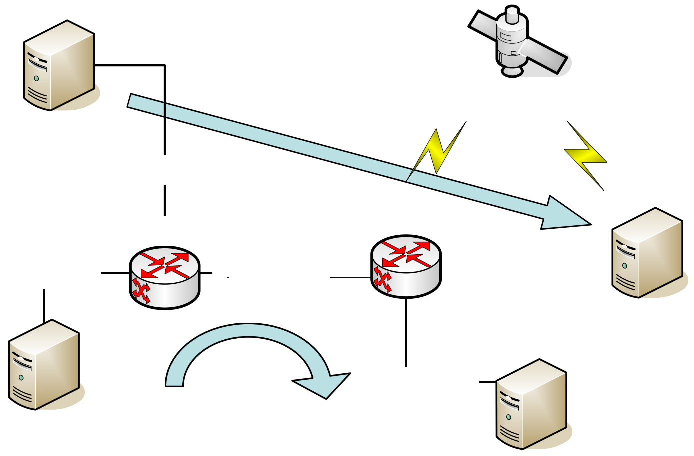


Figure 14. Testbed logical layout.

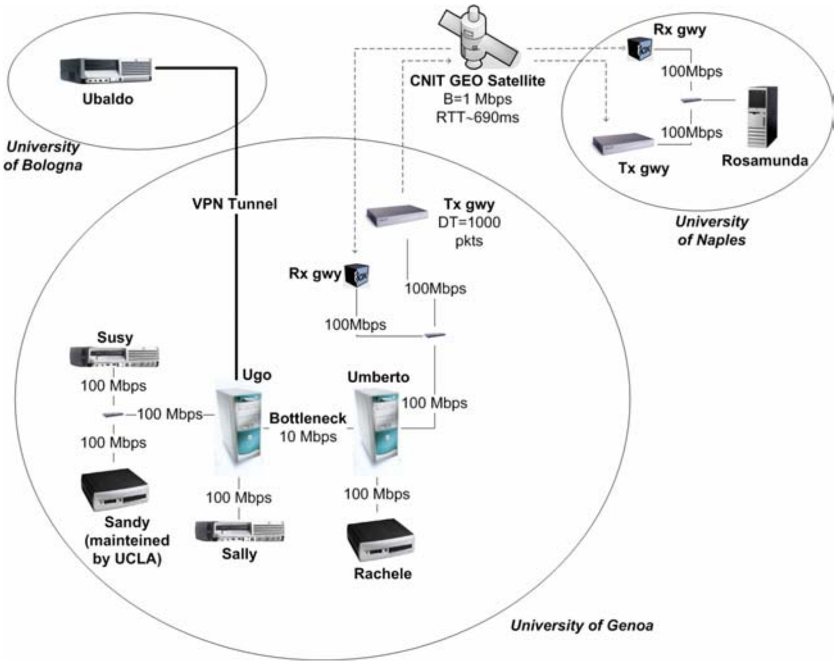
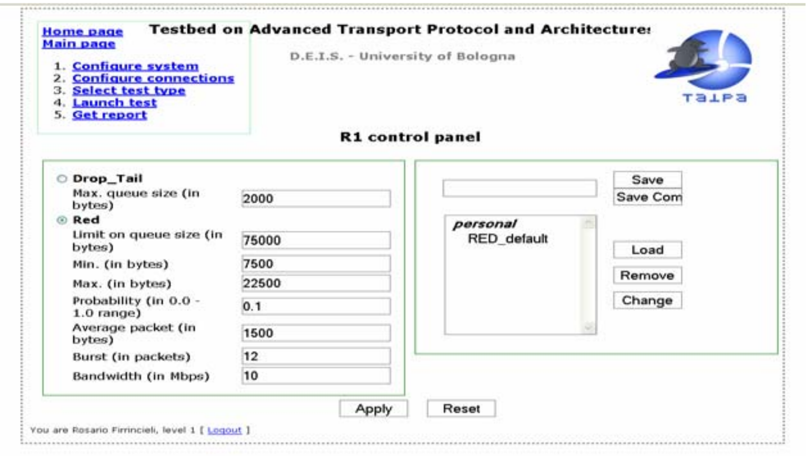


Figure 15. Testbed physical layout.



○ Drop_Tail
Max. queue size (in bytes): 2000
● Red
Limit on queue size (in bytes): 75000
Min. (in bytes): 7500
Max. (in bytes): 22500
Probability (in 0.0 - 1.0 range): 0.1
Average packet (in bytes): 1500
Burst (in packets): 12
Bandwidth (in Mbps): 10

Figure 16.TATPA web interface: configuration of Router 1 settings.

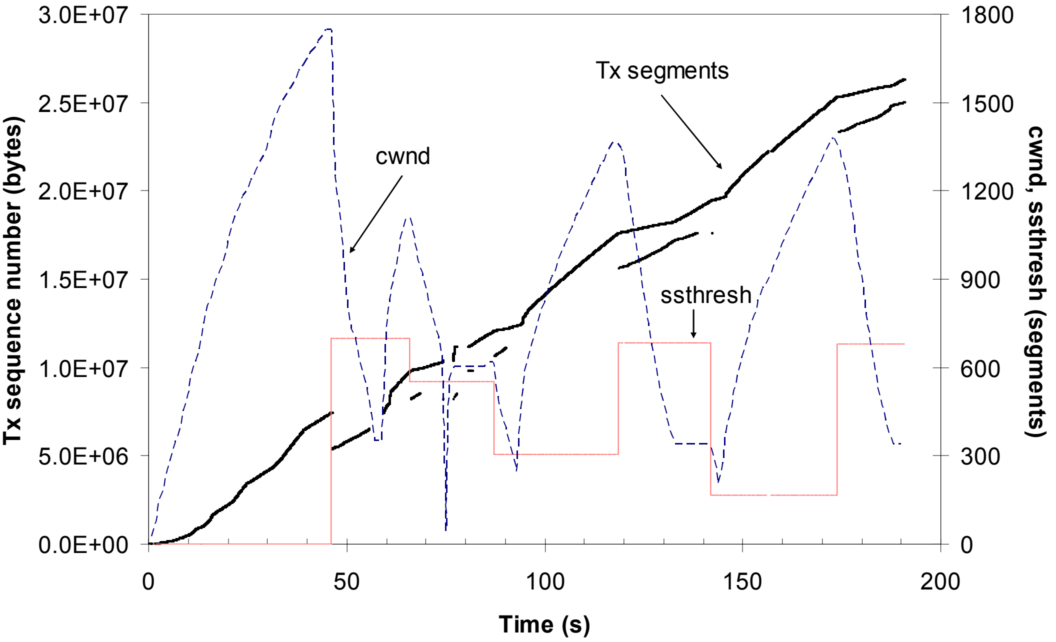


Figure 17. Example of analysis: time series of segments sent, cwnd, ssthresh. Hybla, CNIT Skyplex channel, no background traffic.

5.2. ***EGGS***

5.2.1. Description of the experiment

Remote measurement operations, habitat and environmental control tasks in interplanetary scenarios give raise to challenging issues, in particular in the design of communication protocols. Many peculiarities characterise interplanetary networks: large propagation delay, multi-path fading, solar wind and other hostile radiofrequency conditions that impact on the channel performance. Additionally, because of celestial motion, the line of sight is not always available. The results are: channel disconnections (expected or not) and high bit error rates (ranging from 10^{-3} to 10^{-2}) [40]. For these reasons TCP/IP is not the best candidate. Alternative solutions are Delay Tolerant Network (DTN) and CCSDS protocols [41], or customized flavours of TCP. Given such a wide choice of possibilities, implementing each single protocol within simulators is not tractable; especially if we consider that some of these proposals are still under development, hence permanently evolving. Testing over real testbeds belongs to the future. An experimental tool, which makes it possible to use protocols already implemented, without the costs and complexity of real testbeds is thus strongly needed. For these reasons we consider a viable solution the use of an emulation platform. In particular, the EGGS experiment (Satellite Networking in Challenging Environments) is aimed at evaluating the performance of adequate protocol architectures, suited to transport data over interplanetary networks, by addressing most of the design challenges that may arise in these harsh environment. To this end, an integrated testbed, composed of remote emulation platforms, is proposed, in order to develop a thorough analysis of all the aspects that require to be considered from the implementation point of view. In more detail, attention is mainly paid to the networking issues, by pointing out the advantages offered by the design of such a prototype.

The overall scenario is composed of:

- An Earth station responsible for collecting data originated by a remote sensor network;

- A sensor network [42] located on a remote planet surface;
- A satellite orbiting around the remote planet, serving as relay point between the planet and the Earth.

The presence of different satellite links together with the sensor network makes the use of a single platform impractical. The integration of different emulation tools is strongly required in order, on the one hand, to distribute the whole network complexity among several components and, on the other hand, to control the granularity aimed at characterising the peculiarities of each network portion. It is possible to individuate two kinds of links:

- A proximity link, established from/to the remote planet and the satellite orbiting around;
- A long-haul link, providing the data communication over the deep space path.

5.2.2. The hardware/software used

The integrated testbed is characterized by a long-haul satellite link emulator (called ACE) [43], a sensor testbed (SENS), and a proximity satellite link emulator (DUMMYNET) [44]. Different tools are used for the long-haul and proximity links because the requirements inferred by the environments are not the same.

The long-haul link. Concerning ACE, its basic aim is to emulate a network environment composed of a set of Earth stations that communicate through a satellite link. Its extension towards interplanetary communications is straightforward. It can emulate the data communication among an Earth station, responsible for gathering data arriving from the remote sensor network, and a satellite platform orbiting around the remote planet. Under this view, it is possible to think of these terminal agents as PCs connected to each other by means of an emulated satellite link, working as follows. The whole system is composed of three devices, hosting a Linux O. S., whose role is to perform the basic functionalities of data forwarding and to fully characterise the transmission channel. For this purpose, the emulation task is performed at the data link layer, in order to take into account frame formats, channel coding schemes and proper physical layer characteristics, such as propagation delay, channel bandwidth availability and statistical behaviour of the satellite link (e.g., in terms of bit error ratios and channel modelling).

The proximity link. As far as the proximity link emulation is concerned, it takes the Dummynet Free BSD kernel extension as reference. It allows tuning the propagation delay, the bandwidth availability and the link reliability (e.g., random, uniform loss of packets) through the application of proper schemes and policies implemented at the IP layer.

The sensor network. The sensor testbed consists of a set of Micaz motes from Crossbow Inc. [45]. They are equipped with an 8 MHz microcontroller, an IEEE 802.15.4 – compliant radio [46] and transducers to sample light, temperature, acceleration, magnetism and audio. The embedded software implements a distributed database management system [47] where relational algebra operators, including selection, projections and joins can be carried out on the nodes and interconnected via data stream channels. Users draw the layout of a query graphically and interactively via a GUI application. The query layout concisely represents all data sampling, data processing and data transfer activities to fulfil on each of the network nodes. The GUI interacts with the remote sink via a TCP connection for both sending commands and receiving sensor data. The experimentation core is based on running queries with a variable number of sampling nodes (during the initial tests, it ranges from one to three) and applying different sampling rates (from 100 ms to 2000 ms). The measurements, which typically involve light variations, are properly processed by the sensor network and encoded to allow their transmission to the sink node, which, in turn, relays them to the GUI.

The three components (ACE, DUMMYNET and SENS) described before are located at different premises (Genoa, Toulouse and Pisa). In order to link the three sites, a VPN topology is established with the advantage to make the use of a private addressing plan possible. The drawback of this distributed approach is the natural delay introduced by the Internet: it has to be taken into account. A Perl script repeatedly measuring the RTT helps to adjust the delay as needed.

5.2.3. The results

Two scenarios are considered: (a) full-TCP and (b) DTN/TCP. In the former, the sensor network GUI is located on the Earth and a TCP connection is established with the sensor network sink. The TCP connection therefore goes over the long-haul link with the expected limitations of TCP.

In the DTN/TCP scenario, the sensor network is connected to the orbiter via a TCP connection (proximity link). A DTN/TCP proxy in the orbiter translates sensor data sent from the sink into DTN bundles and forwards them via UDP to the Earth. The DTN/TCP proxy is written in Perl. The DTN stack is the reference implementation available from the DTN Research Group [48]. The sampling rate of the sensor network for the DTN scenario is set to 10 measures per second while it is twice that value for the full-TCP scenario. A measure is approximately 25 bytes. The current implementation of the DTN stack has difficulties to sustain high rates and issues a threading runtime error. Finally, all network links are constrained to a throughput of 64 Kbit/s. The propagation delay for the long haul link ranges from 125 ms to 200 s. The proximity link has a propagation delay of 40 s.

Measures are collected in the Earth station (i.e., in Genoa), being t*cpdump* used to record traffic traces. Currently, bit errors are not taken into account in the experiment, although they can be generated by the two link emulators if needed.

The full-TCP scenario. As far as the full TCP scenario is concerned, particular attention has been paid to the impact of large latencies on the transmission protocol. To better understand this aspect and hence to assess the operation of the proposed emulation platform, different propagation delays have been considered for the long haul link (250 ms, 1280 ms, 10 s, 50 s and 200 s). We first tested the behaviour of TCP, when the largest propagation delay (200s) is set. This trial showed, as expected, the unfeasibility of using TCP protocols over “very long” networks, due to the TCP timers and related algorithms tuned to the more common terrestrial path delays. To tackle these problems, TCP parameters have been tuned accordingly to the environment peculiarities. Trials with delays equal to 250 ms, 1280 ms, 10 s and 50 s have been successfully completed, while the case of 200 s still experienced some hazards, because of the limits of the TCP back-off algorithm triggered during timeout expirations.

Figures 18 and 19 show the behaviour of instantaneous packet transmissions and congestion window over the elapsed time and the TCP segment number, respectively. From these graphs, it is confirmed that the long haul link propagation delay impacts the dynamics of data transfer changes. In particular Fig. 1 demonstrates that for a 250 ms delay, the transmission of TCP segments follows a linear law, while, on the opposite, when a delay of 50 s is set, the increase in the number of segments sent each

round trip time is very low. As a consequence, steps are observed.
Finally, Figs 17 and 18 outline that, in general, even if large latencies are set, the proposed emulation platform is able to match the channel peculiarities. Additional tests have still to be conducted in order to assess the limits of the emulation platform, especially when large delays are used in conjunction with high data rates.

The DTN/TCP scenario. The DTN/TCP experiment deploys the bundle protocol over the long haul link. UDP was selected as subnetwork, in order to work around issues related to the impairment of conversational protocols over very long networks.

Figure 20 shows the received UDP datagrams with respect to time. The results related to the 200 s experiment display an interruption in the reception of the UDP flow. While this has still to be confirmed, buffering problems in the long haul emulator are suspected.

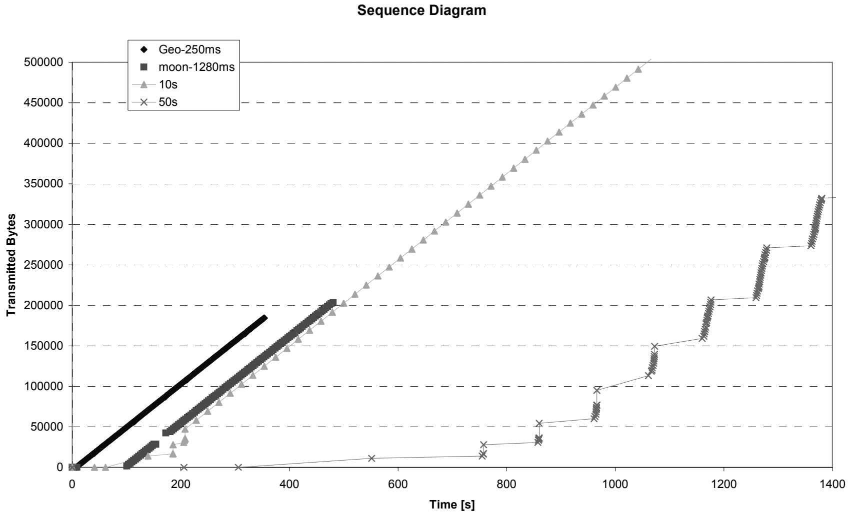


Figure 18. Diagram sequence for the four investigated scenarios.

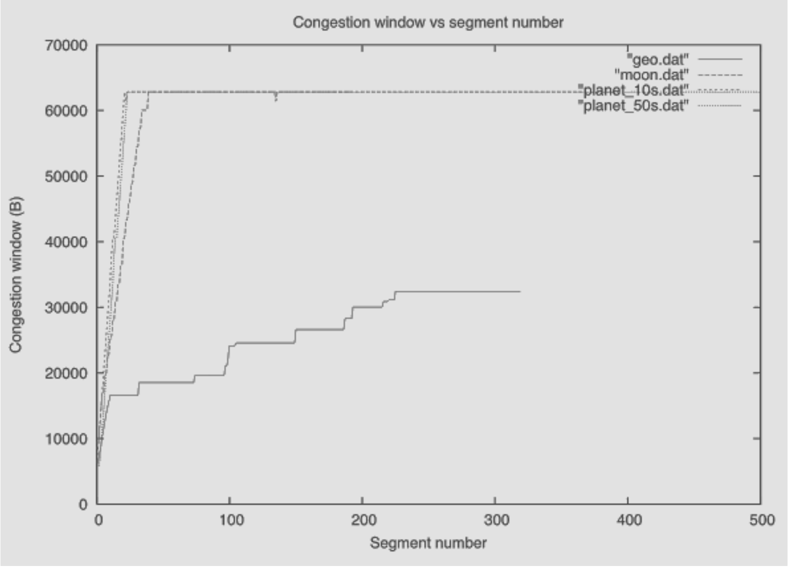


Figure 19. Congestion window diagram over TCP segments.

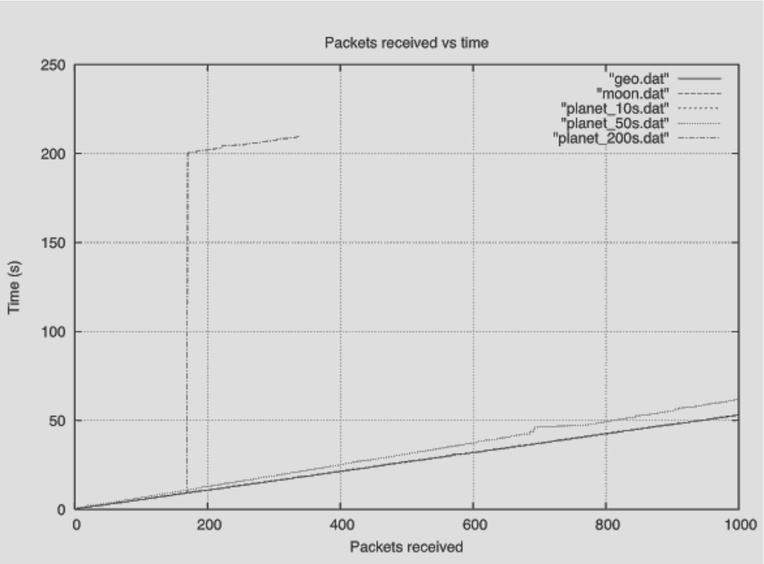


Figure 20. Received datagrams vs. time for the DTN/TCP trial.

6. CONCLUSIONS

The paper focused on the effort, made within the European-funded project SatNEx, in designing and developing investigation tools suitable for analysing the performance of data communication achieved in satellite environments. In this perspective, attention was devoted to experiments consisting of both measurement campaigns (i.e., VOTOS, VoIP over Satellite, and WICHMO) and realisation of integrated testbeds, allowing to study the performance offered by higher layer protocols in satellite environments (i.e., UCIT) and by Delay Tolerant Network architectures in interplanetary scenarios (i.e., EGGS).

On the one hand, measurement campaigns provided insights into dynamics of some of the most relevant networking and communication issues in satellite-based scenarios, such as channel modelling, analysis of multimedia applications and performance study of transport layer protocols, by taking the effect of DAMA-TDMA access schemes under consideration.

On the other hand, the desire for a more thorough investigation of the implementation issues arising in the development of real satellite systems led to the design of two testbeds able to study the behaviours of protocols and architectures, enabling communications in satellite and interplanetary scenarios. Owing to their powerful capabilities of analysis, these investigation tools resulted to be important means, available to the whole satellite community, for studying the dynamics of real satellite systems.

REFERENCES

1. Skianis C., Jamalipour A., Fairhurst G., Montpetit M-J, Donadio R. Guest editorial—Convergence of internet and broadcasting systems. *IEEE Network 2007*; **21(2)**: 4-5.
2. Evans B., Werner M., Lutz E., Bousquet M, Corazza G. E., Maral G., Rumeau R., Ferro E. Integration of satellite and terrestrial systems in future multimedia communications. IEEE Wireless Communications 2005; **12(5)**: 72-80.
3. Jamalipour A. Message from the editor-in-chief: New generation heterogeneous mobile networks. IEEE Wireless Communications 2007; **14(2)**: 2-3.
4. Wong K. D., Tepe K., Chen W., Gerla M. Guest editorial — Inter-vehicular communications. IEEE Wireless Communications 2006; **13(5)**: 6-7.
5. Skyplex, http://www.eutelsat.com/fr/satellites/pdf/dealers/annex_d_skyplex.pdf.
6. Annese A, Celandroni N, Davoli F, Ferro E, Gotta A. TCP performance measured over wireless

integrated networks with high delay-bandwidth products. *Proceedings of ASMS* **2006**.

7. Mathis M, Mahdavi J, Floyd S, Romanow A. TCP Selective Acknowledgement Options. *RFC 2018* **1996**.
8. Grieco L A, Mascolo S. Performance evaluation and comparison of Westwood+, New Reno and Vegas TCP congestion control. *ACM Computer Communication Review* **2004**; 34(2):25-38.
9. Gotta A, Secchi R, Potortì F. An analysis of TCP startup over an experimental DVB-RCS platform. *Proceedings of IWSSC* **2006**.
10. Handley M, Floyd S, Padhye J, Widmer J. TCP Friendly Rate Control (TFRC): Protocol Specification. *RFC 3448* **2003**.
11. Gotta A, Secchi R, Potortì F. Simulating dynamic bandwidth allocation on satellite links. *Proceedings of Valuetools, WNS2 workshop* **2006**.
12. Sathiaseelan A, Fairhurst G. Using Quickstart to Improve the Performance of TFRC-SP over Satellite Networks. *Proceedings of IWSSC* **2006**.
13. Floyd S, Allman M, Jain A, Sarolahti P. Quick-Start for TCP and IP. *RFC 4782* 2007.
14. Bonito AB, Gotta A, Secchi R. Advanced satellite infrastructures in future global Grid computing: network solutions to compensate delivery delay. *Proceedings of INGRID* **2007**.
15. Debian Linux, http://www.debian.org.
16. Asterisk VoIP PABX, http://www.asterisk.org.
17. trixbox - telephony application platform, http://www.trixbox.org.
18. AsteriskNOW, http://www.asterisknow.org.
19. rPath Linux, http://www.rpath.com.
20. XLite - SIP SoftPhone, http://www.counterpath.com.
21. Idefisk - IAX SoftPhone, http://www.asteriskguru.com/idefisk/free/.
22. WireShark, http://www.wireshark.org.
23. IAX2: Inter-Asterisk eXchange Version 2 - draft-guy-iax-03, http://www.ietf.org/internet-drafts/draft-guy-iax-03.txt.
24. SIP IETF RFCs/WGs, http://www.sipknowledge.com/SIP_RFC.htm.
25. Davoli F, Ferro E, Mouftah H. Wireless access to the global Internet: mobile radio networks and satellite systems. *Guest editorial, Int. Journal of Communication Systems* **2003**; **16(1)**
26. Barsocchi P. Packet Loss in Terrestrial Wireless and Hybrid Networks. *PhD Thesis, University of Pisa (IT)* **2007**.
27. Allman, Dawkinks S, Glover D, Griner J, Henderson T, Heidemann J, Ostermann S, Scott K, Semke J, Touch J, Tran D. Ongoing TCP research related to satellites*. IETF, RFC 2760* **2000**.
28. Barsocchi P, Oligeri G, Potortì F. Packet loss in TCP hybrid wireless networks. *Proceedings of the Advanced Satellite Mobile Systems Conference (ASMS)* **2006**.
29. Pang Q, Leung VCM. Liew SC. A Rate Adaptation Algorithm for IEEE 802.11 WLANs Based on MAC-Layer Loss Differentiation. *Proceedings of the International Conference on Broadband Networks* **2005**.
30. Altman E, Avrachenkov K, Barakat C. A stochastic model of TCP/IP with stationary random losses. *Computer Communication Review* **2000**; **30(4)**: 231-242.
31. Kamerman A, Monteban L. WaveLAN-II: a high-performance wireless LAN for the unlicensed band. *Bell Labs Technical Journal* **1997**; **2(3)**: 118-133.
32. Caini C, Firrincieli R. TCP Hybla: a TCP Enhancement for Heterogeneous Networks. *Int. J. Satell. Commun. Network* 2004; **22**: 547-566.
33. Border J, Kojo M, Griner J, Montenegro G, Shelby Z. Performance Enhancing Proxies Intended to Mitigate Link-Related Degradations. *IETF RFC 3135* **2001**.
34. Cerf V , Hooke A, Torgerson L, Durst R, Scott K, Fall K, Weiss H. Delay-Tolerant Networking Architecture. *IETF RFC 4838* **2007**.
35. TATPA, https://tatpa.deis.unibo.it.
36. Caini C, Firrincieli R, Lacamera D. A Linux Based Multi TCP Implementation for Experimental Evaluation of TCP Enhancements. *Proceedings of SPECTS **2005***.
37. Caini C., Firrincieli R., Lacamera D. PEPsal: a Performance Enhancing Proxy for TCP satellite connections. *IEEE Aerospace and Electronic Systems Magazine 2007*; **22(8)**: 7-16.
38. Caini C., Cornice P., Firrincieli R., Lacamera D., Tamagnini S. A DTN Approach to Satellite Communications. *Proceedings of IEEE IWSSC **2007.***
39. Caini C., Firrincieli R., Lacamera D., De Cola T., Marchese M., Marcondes C., Sanadidi M. Y., Gerla M. TCP Live Experiments on a Real GEO Satellite Testbed. *Proceedings of IEEE ISCC **2007.***

40. Baronti P., Pillai P., Chook V., Chessa S., Gotta A., Hu F.Y. Wireless Sensor Networks: a Survey on the State of the Art and the 802.15.4 and ZigBee Standards . *Computer Communications* **2007**; **30**: 1655-1695.
41. Akyildiz I. F., Akan O. B., Chen C., Fang J., Su W. The state of the art in interplanetary internet. *IEEE Communications Magazine* **2004**; **42(7)**: 108-118.
42. Hooke A. J. Towards an Interplantetary Internet: A Proposed Strategy for Standardization. *Proceedings of Space Operations Conference and World Space congress* **2002**.
43. Marchese M., Perrando M.. A packet-switching satellite emulator: A proposal about architecture and implementation. *Proceedings of IEEE International Conference on Communications (ICC)* **2002**.
44. Dummynet, http://info.iet.unipi.it/~luigi/ip_dummynet/.
45. Crossbow Technology Inc., http://www.xbow.com.
46. Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (LR-WPANs), IEEE Std. 802.15.4-2003 **2003**.
47. Amato G., Baronti P., Chessa S. MaD-WiSe: Programming and Accessing Data in a Wireless Sensor Networks. *Proceedings of IEEE Eurocon* **2005**.
48. Delay Tolerant Networking Research Group (DTNRG) http://www.dtnrg.org.