This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2018.2865150, IEEE Transactions on Aerospace and Electronic Systems

TCP-based M2M Traffic via Random-Access Satellite links: Throughput Estimation

Manlio Bacco, Tomaso De Cola, Giovanni Giambene, Alberto Gotta

Provisioning of IoT/M2M services over satellite has been experiencing a continuous growth in the last years, which is expected to further increase in the near future so as to meet the demands of users and enterprises. The design of a suitable network architecture is, hence, of paramount importance to properly take into account the requirements imposed by the technology available nowadays and to properly consider the interaction of the so-defined physical layer with transport and application layers. In this light, this paper analyses the use-case of TCP-based M2M services operating over DVB-RCS2 satellite links, where a Contention Resolution Diversity Slotted Aloha access scheme is applied. The main goal of the paper is to provide a thorough understanding of the interactions of TCP and random access schemes, recognised as key elements to enable efficient M2M services over satellite. In this regard, this paper also develops a novel TCP throughput model, which has been validated through extensive simulation campaigns, proving the value of the proposed theoretical framework and its applicability to study the performance of M2M services in more general satellite scenarios.

***Index Terms*—satellite communications, DVB-RCS2, random access, M2M, TCP model**

I. Introduction

According to data traffic forecast reports (e.g., CISCO VNI report 2017), the volume of data transported by Internet in 2021 will exceed the threshold of 3.0 zettabytes per year, generated by roughly 30 billions of devices. Only a minor portion of the traffic will be generated by PCs, as commonly observed in the recent past; on the contrary, a large quota of Internet traffic will be generated by TVs, tablets, smartphones, and Machine-to-Machine (M2M) devices. In particular, it has been highlighted that M2M traffic will experience a growth rate in the order of about 60%, with the total number of M2M devices to represent the 50% of overall devices.

In response to this forecast, the scientific community started to thoroughly investigate the network design implications based on the experience of the many M2M services running over terrestrial infrastructures. The survey contained in [1] underlines the fragmentation of current M2M markets, leading to the design and implementation of vertical technological solutions, isolated in the so-called *silos*. Further to this, the paper in [1] shows that typical Internet of Things (IoT)/M2M services exhibit intermittent behavior (e.g., ON-OFF oscillations), low data-rates, and high traffic burstiness, thus motivating the use of Random Access (RA) schemes to make use of the available network capacity properly.

The use of satellite platforms for M2M services has become more and more widespread, with an unprecedented increase over the last five years (see for example [2], [3]) and expected further diffusion in the near future [4]. The reason for such a success is twofold: on the one hand, very highly dense networks, such as M2M ones, pose several challenges to terrestrial wireless networks [5]; on the other hand, the modernization of important industry sectors, such as maritime and aeronautical communications, calls for more advanced transmission techniques to support ship tracking services, and aeronautical telemetry transmission, just to mention a few [6].

A relevant example of satellite technology in the context of M2M is represented by the second generation of DVB-RCS (DVB-RCS2) (ETSI EN 301 545-2, DVB-RCS2: Lower Layers for Satellite Standard.), specifying the transmission schemes and the protocol architecture for the return link, where Contention Resolution Diversity Slotted ALOHA (CRDSA) [7] along with Slotted ALOHA (SA) are proposed as complementary methods to the traditional Demand Assignment Multiple Access (DAMA). CRDSA offers immediate access to satellite capacity without incurring in the reservation delays of DAMA that can penalise the transmission of M2M bursty traffic. It consists in a more sophisticated version of SA, where multiple replicas of the same Media Access Control (MAC) packet are sent and where accidental collisions can be solved using Successive Interference Cancellation (SIC).

Despite the increasing interest dedicated by the scientific community to M2M applications over RA satellite links, the studies conducted so far have been mostly focused around the implications of RA on physical layer design, whereas its interactions with upper layers (e.g., TCP/IP stack) have received less attention. Actually, to the best of authors' knowledge, a few studies analysed these aspects: it is here worth citing the preliminary analytical investigation in [8], considering a simplified TCP over an SA satellite link, while the work in [9] provides simulation results over CRDSA. In addition, the works in [10]–[12] consider the use of TCP-based M2M/IoT providing an empirical evaluation of the performance at the application layer. Anyway, those works contain preliminary evaluations on the impact of the collision resolution scheme on TCP congestion control in terms of delay as well as the reaction of TCP in response to collisions. A rigorous investigation is still missing from a theoretical standpoint.

To this end, this paper attempts to bridge this scientific gap, by analyzing the performance of TCP-based application protocols over CRDSA++, using 3 replicas and running in a

(corresponding author: Tomaso De Cola, Tomaso.deCola@dlr.de)

Manlio Bacco and Alberto Gotta are with the Institute of Information Science and Technologies, National Research Council (CNR), Pisa, Italy

Tomaso De Cola is with the German Aerospace Center (DLR), Institute of Communications and Navigation, Oberpfaffenhofen, Germany

Giovanni Giambene is with the Department of Information Engineering and Mathematical Sciences, University of Siena, Italy

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2018.2865150, IEEE Transactions on Aerospace and Electronic Systems

DVB-S2/RCS2 network under different traffic load conditions. A solid analytical model of TCP dynamics over RA is provided here, adapting and extending the TCP models in [13]–[15] that were not intended for RA-based scenarios. Furthermore, we propose a simple but effective way to estimate the input parameters required by our analytical model for the TCP throughput evaluation over RA via satellite, by only relying on the collision rate experienced in the RA channel. This analytical approach is then validated by means of extensive NS-3-based simulations. It is worth noticing that the availability of a TCP model can provide some more insights into the protocol performance than a simulation tool, owing to the much more reduced computation times and the capability of enabling deeper theoretical studies as to transport protocol optimisation for M2M services. Finally, the performance analysis herein proposed shows very good correspondence between the results coming from simulations and the proposed model, hence providing additional arguments in favour of our model.

The most used acronyms in this work are reported in Table I, and the rest of this paper is structured as follows: Section II surveys the main findings on TCP modeling and interactions with RA schemes, paying special attention to the case of satellite networks. Section III introduces the reference system, while Section IV proposes some refinements that are necessary on RA links in order to correctly estimate TCP throughput. Section V offers a simple but effective model to link together burst losses at MAC layer due to collisions and segment losses at transport layer. The performance analysis is carried out in Section VI. The conclusions are drawn in Section VII.

II. Background and main contributions

A. State of the art of TCP modeling

A thorough understanding of the interactions between RA and TCP dynamics actually requires delving into the current literature devoted to TCP modeling, by paying attention on the underlying channel model and the applied theoretical frameworks. Since the study presented in this paper deals with the interactions of different protocol layers (*cross-layer* approach), we see the renewal process modeling as a very suitable candidate to analyse the TCP behavior. In other words, the main advantage offered by renewal processes is in easily capturing the not-trivial interactions of TCP congestion control algorithms with other architectural elements occurring at random time instants after which the system returns to a state equivalent to the starting state. As a matter of fact, the first successful attempt to model the TCP performance using a renewal process is contained in the seminal work by Padye *et al.* [13] and later in [16]. Its main outcomes, then refined in [17], are the so-called Square Root Formula (SRF) and the Approximated Model (AM). The model in [13] treats segment losses as independent events in different rounds and as correlated events within the same round: each segment delivered after the first segment loss is supposed to be lost as well, independently of its actual outcome. The throughput models in [13] have been derived for TCP Reno recovery algorithm, which can manage a single loss during the Fast Recovery / Fast Retransmit (FR)[1]. However, recent TCP implementations, like NewReno or Cubic, have more complex error recovery mechanisms that allow managing multiple losses during the FR phase. Other works have extended SRF and AM in order to account for the dynamics of FR phase with Slow-But-Steady (SBS) flavor [14], [15]. The most prominent innovation in [14] and [15], with respect to [13], is that both bursty and independent losses are modeled.

B. TCP in satellite scenarios

When specifically targeting the use of TCP in satellite links, optimized flavors are available (for instance, see [18], [19]) that however are rarely used in real systems because of the ossification of the network stacks implemented in operating systems. The work in [8] is the very first rigorous analytical approach to study the TCP dynamics in the presence of an RA channel access, for which upper and lower throughput bounds are provided for a simplified TCP model. Another work considering the use of TCP via satellite channels can be found in [20], comparing dedicated access and random access performance in the presence of elastic traffic; the authors conclude that hybrid approaches may be of interest, similarly to what described in [21], which also suggests a possible architecture for taking advantage of both access methods in the presence of M2M traffic, mixed with different traffic sources, via geostationary satellite uplinks. When considering the coexistence of elastic and inelastic traffic in the same channel, the analytical model in [22] has inspired other congestion control algorithms, as for instance the so-called TCP-Friendly Rate Control (TFRC)[2], designed to provide acceptable fairness in the channel use. An optimization of TFRC for long delay links, able to distinguish between losses due to buffer overflows from those due to channel errors or collisions has been proposed in [23]. The use of a TFRC-like congestion control algorithm, running at the application layer, is still of interest nowadays in M2M/IoT scenarios because of its trade-off between complexity and achievable performance, as analysed in [24]. The performance of more common TCP flavors is analyzed in [25], considering a satellite RA medium. In particular, the authors assume that the MAC layer operates at a target load G^* and that the Network Control Centre (NCC) periodically broadcasts an *activity factor* (i.e., a throttling parameter) to dynamically adjust a flow control policy applied at Return Channel Satellite Terminals (RCSTs), to maximize the achievable throughput. Analogously, [26] analyzes TCP over CRDSA when packet-level Forward Error Correction (FEC) techniques are employed. Furthermore, a single TCP connection is modeled through a fluidic model in [26] that anyway provides a less precise approach than the packet-level one we adopt.

C. Main contributions of this work

This work extends the preliminary contribution in [27], where the problem was first discussed. Here, a more realistic

[1]RFC 6582: The NewReno Modification to TCP Fast Recovery Algorithm.
[2]RFC 5348, TCP Friendly Rate Control (TFRC): Protocol Specification.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2018.2865150, IEEE Transactions on Aerospace and Electronic Systems

set of MAC parameters is taken into account thanks to the use of the S-NS3 simulator [28], allowing for a precise analysis of the behavior of TCP over a CRDSA-based satellite channel. Thanks to the aforementioned simulation environment, we have shown that our work provides a more accurate analysis than that in [20], [21]. A work closely related to this one can be found in [11] that however focuses on a different performance metric (i.e., the completion time), neglecting the impact of data fragmentation, and only taking low-to-medium traffic conditions into account. Instead, in this work, we envisage a wide range of nominal offered loads, thus providing a complete overview of TCP behaviour for different numbers of M2M/IoT terminals. Furthermore, this work also suggests guidelines for TCP use in the presence of IoT/M2M traffic over CRDSA++, removing the need for any other flow control scheme at MAC layer.

To summarize, the main contributions of this work are:

- a finer and more precise model of TCP steady-state throughput, by bringing together the main findings from [13]–[15];
- a simple but effective model establishing a relation between the loss rate at MAC layer and the Segment Loss Rate (SLR) at transport layer (cross-layer study);
- the analysis of how the TCP congestion control algorithm behaves in a channel dominated by RA-induced collisions, where TCP auto-regulates the sending rate to counteract an SLR increase;
- finally, considerations are provided on the system stability guaranteed by the TCP congestion control algorithm, even at very high offered loads, i.e., for a very large population insisting on the satellite RA channel.

III. System Architecture and Setup Parameters

A. Reference scenario

This paper focuses on the case of M2M data distribution from remote sites to local consumers by means of an RA satellite link because of the bursty nature of M2M traffic [29]. In more detail, we focus on event-driven scenarios with indirect access (i.e., by means of a local gateway) when a large number of RCSTs, greater than the number of transmission opportunities per time-unit (see Section III-B), contend for the channel. This setup reproduces real satellite-based M2M services such as mobile messaging, tracking and monitoring of assets for transportation and maritime environments (e.g., *Orbcomm* or *Globalstar* services), or general-purpose M2M/IoT applications as envisioned for S-MIM applicability [3], [30]. More particularly, we consider the case of multiple M2M sources generating data, which are in turn *published*, according to a Publish / Subscribe (PUB/SUB) paradigm; eventually, a specialised gateway (*broker* node) collects data to be forwarded to *subscribers* via satellite link. Our reference scenario is depicted in Figure 1, where sensing nodes are organised in local Wireless Sensor Networks (WSNs), managed by a local broker, also referred to as Cluster Head (CH).

As far as the protocol architecture is concerned, typical installations adopt Message Queue Telemetry Transport (MQTT)

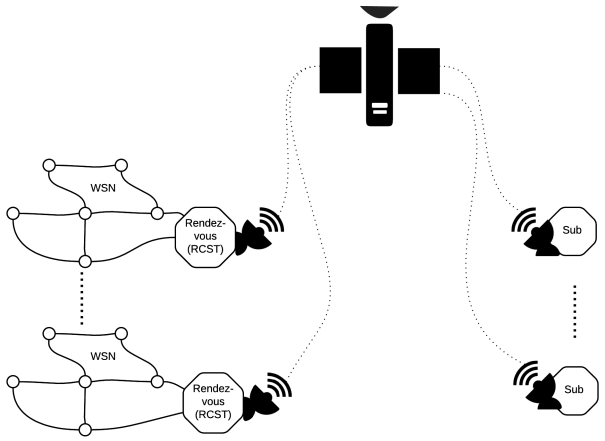


Fig. 1: Scenario under investigation: multiple RCSTs acting as rendezvous nodes (or brokers) and delivering data to remote subscribers via RA satellite link

on sensing nodes and local subscribers, an IoT/M2M application protocol developed by IBM in 1999 to enable messaging applications via satellite and recently standardised by OASIS. An alternative option is the Constrained Application Protocol (CoAP), specified in IETF RFC 7252 recommendation, which is indicated by ETSI (see for example ETSI TS 102 690) as a reference protocol for M2M/IoT scenarios. Nevertheless, we consider the use of an MQTT-like protocol architecture as more promising for the scenario under investigation in this paper, since MQTT runs on top of TCP and operates according to a PUB/SUB paradigm envisioning a broker node acting similarly to a Split-TCP PEP (Performance Enhancement Proxy), a commonly-implemented entity in satellite networks. On the contrary, CoAP runs on top of UDP and relies on a request/response interaction model based on REST (Representational State Transfer) syntax; some considerations about its use in satellite-based M2M scenarios can be found in [12].

In our envisaged scenario based on MQTT, the stream of M2M messages is exchanged between service publishers and subscribers by means of TCP connections. In more detail, it is assumed that each broker initiates one persistent TCP connection to transfer the data generated by the corresponding WSNs to the subscribers. The TCP connection is actually established only at the very first data transmission to avoid performing multiple consecutive three-way handshake (3WHS) operations for each new transmission instance between a broker node and the corresponding subscribers. In turn, TCP connection segments are transported by the underlying satellite technology (detailed in the next section), linking brokers to subscribers.

On the other hand, as far as the local WSN among the IoT/M2M nodes and the broker node is concerned, we assume that no packet losses are introduced, whereby the network challenges are confined to the satellite link, which actually represents the core of the study carried out in this paper.

Finally, the performance metric under consideration is the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2018.2865150, IEEE Transactions on Aerospace and Electronic Systems

TABLE I: List of frequently used acronyms

Abbreviation	Full text	Abbreviation	Full text
ACK	Acknowledgment	RCST	Return Channel Satellite Terminal
ARQ	Automatic Repeat reQuest	RTT	Round Trip Time
BLR	Burst Loss Rate	SIC	Successive Interference Cancellation
CA	Congestion Avoidance	SLR	Segment Loss Rate
CAFR	Congestion Avoidance - Fast Recovery / Fast Retransmit	SS	Slow Start
CRDSA	Contention Resolution Diversity Slotted Aloha	SST	Slow Start Threshold
CWND	Congestion Window	TD	Triple-Duplicate ACK
FR	Fast Retransmit / Fast Recovery	TDP	TD Period
MSS	Maximum Segment Size	TO	TimeOut
NCC	Network Control Center	WF	Waveform
RA	Random Access	3WHS	Three-Way HandShake

throughput at transport layer[3] that corresponds to the throughput experienced by the application itself, apart from the overhead introduced at the application layer.

B. Satellite system

We assume a geostationary bent-pipe satellite system based on the DVB-S2/RCS2 technology, composed of a finite number N of RCSTs, exploiting the CRDSA++ access scheme [31] for the return link. The channel access is implemented in a time slotted manner and organised in consecutive *superframes*, each further subdivided in a number of *frames*; according to DVB-RCS2 terminology, a random access frame is referred to as *RA block*. In our configuration, an RA block spans the whole superframe and is composed of T_S time slots or *transmission opportunities*. Each RCST can exploit any transmission opportunity to send data blocks over the return channel, also referred to as *bursts*, which can be in a finite number s per frame per RCST. Without loss of generality, we assume $s = 1$ (i.e., each RCST can use a single time slot per RA block) throughout the rest of this paper and a transmission opportunity can be used with probability equal to 1, as soon as data are available for transmission. Finally, each burst is transmitted by the physical layer according to a predefined Waveform (WF) ID, specifying the modulation, the channel coding rate, and the overall gross burst size. Typically, WFs are chosen on the basis of specific application profiles (as discussed later in this section) or transmission channel conditions (e.g., according to SINR measurements).

Interworking between DVB-RCS2 MAC sublayer and the higher protocol stack is guaranteed by the Return Link Encapsulation (RLE) protocol operating at layer 2 and in charge of encapsulating IP layer datagrams into physical layer bursts. To this end, classical datalink layer protocol functions such as *fragmentation*, *frame packing*, and protocol header implementation are performed. On the one hand, a datagram is fragmented into multiple RLE packets in case it is larger than the maximum MAC payload length. On the other hand, multiple RLE packets can be packet together in the same frame if the size is smaller than the MAC payload length. The optimal case is when the datagram size exactly fits the maximum MAC payload length.

In order to better formalize encapsulation and frame packing processes, let $r \geq 1$ denote the length of a physical layer burst measured in RLE data units (corresponding to IP datagrams) that are packed together, while $f = \lceil r \rceil \geq 1$ represents the number of RLE data units lost in a single MAC-layer collision event on the RA channel. In the case of a collision, $f = \lceil r \rceil$ RLE data units and correspondingly f IP datagrams are erased, thus requiring retransmissions. Moreover, we assume no IP fragmentation in the following, so that one IP datagram carries a whole TCP segment, meaning that the loss of f IP datagrams will result in the loss of f TCP segments, then triggering the necessary TCP recovery functions. Conversely, the case of IP datagram fragmentation into multiple RLE data units will cause the loss and retransmission of just one TCP segment.

As far as the DVB-RCS2 configuration is concerned, the standard defines bursts with two possible lengths (536 and 1616 symbols, respectively) that are transmitted using a rich set of waveforms (WFs). The redundancy added by channel coding as well as the modulation order can be tuned according to channel conditions: DVB-RCS2 provides several predefined configurations, or WF IDs, in order to achieve target Bit Error Rate (BER) figures. As a result, the time slots of an RA block can accommodate different amounts of data.

In the present work, WF 14 and WF 3 configurations are selected (see Table II for more details). The RA block of WF 14 is composed of time slots each carrying 188 bytes; instead, the RA block of WF 3 is composed of time slots each carrying 38 bytes. Note that WF 3 is one of the most robust and used waveforms, relying on Quadrature Phase Shift-Keying (QPSK) $1/3$, while WF 14 uses QPSK $1/2$. The choice of these two WFs stems from the fact that a typical M2M message length is in this range (38 - 188 bytes) [29].

As to the operative channel conditions, we assume that the system operates in clear-sky conditions (e.g., under ideal fading conditions) to have a precise understanding of the interaction between RA strategies and TCP congestion control algorithms. Moreover, RCSTs are power balanced[4], which

[3]Throughout the rest of this paper, we will refer to TCP throughput, although non-TCP flows could also share the satellite link. Nevertheless, the coexistence of TCP-based and non-TCP-based traffic flows in the satellite return link is beyond the scope of this work and left to future studies.

[4]In the reference scenario, RCSTs are uniformly distributed within the same beam. We assume that an uplink power control scheme is implemented, so that the power of the signal received at the satellite gateways from each RCST is approximately the same.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2018.2865150, IEEE Transactions on Aerospace and Electronic Systems

TABLE II: Details on the DVB-RCS2 waveforms in use. The number of time slots T_S depends on additional factors.

Waveform ID	Burst length [symbols]	Payload length ([B], [symbols])	Modulation scheme	Code rate	no. time slots T_S
3	536	38, 456	QPSK	1/3	194
14	1616	188, 1504	QPSK	1/2	64

corresponds to the most challenging configuration for RCSTs using RA schemes, because power unbalancing would induce the *capture effect*, thus improving the performance of SIC.

Under this configuration, we will provide an analytical model in Sections IV and V to predict TCP throughput in supporting IoT/M2M traffic over a satellite RA channel affected by segment losses due to collisions. In particular, this model is developed taking as reference the case of an endless queue of IoT/M2M data pending for transmission, so that the offered traffic is sustained, which is a reasonable assumption for large deployments of sensors nodes, as in our study.

C. TCP segment size and fragmentation at MAC layer

Two alternative Maximum Segment Sizes (MSSs) are considered: 23 and 173 bytes, meant to exactly fit the payload size provided by WF 3 and by WF 14, respectively. IoT/M2M terminals generate bursts of data at low data-rates, thus those MSSs are enough to transmit small amounts of data. Since we are dealing with small IoT/M2M payloads, RObust Header Compression (ROHC) is used to reduce the impact of TCP/IP overhead, eventually resulting in 7 bytes for data segments and to 6 bytes for Acknowledgment (ACK) segments.

Recalling that an IP datagram carries a whole TCP segment, three scenarios are possible, as described in Table III, depending on the time slot payload size:

1) the MSS is equal to the MAC payload length ($r = 1$): one TCP segment exactly fits one MAC burst, thus a collision event causes a single segment loss, i.e., $f = 1$;
2) the MSS is larger than the MAC payload length ($r < 1$): one TCP segment fits into several MAC bursts. A collision event causes a single segment loss, because losing a fraction of the segment is equivalent to losing the whole segment, i.e., $f = 1$;
3) the MSS is smaller than the MAC payload length ($r > 1$): several segments fit into a single MAC burst, thus a collision event causes $f = \lceil r \rceil > 1$ segment losses.

The third case is discussed separately in Section IV because it violates the hypothesis of uncorrelated losses under which the TCP model has been here developed. Moreover, we will show in Section VI that this configuration provides poor performance: any configurations with $f > 1$ should be avoided because it is not efficient.

D. Interactions between TCP and the underlying RA scheme

The traffic arrival process for each RCST is driven by a sustained and continuous load at application layer, because each broker node continuously collects IoT/M2M data from the WSN and sends them to remote subscribers. The

TABLE III: Fragmentation at MAC layer

MSS + headers [bytes]	WF ID	$r = \frac{RLE\,payl.(WF)}{MSS+head.}$	$f = \lceil r \rceil$
23 + 15	3	1	1
23 + 15	14	5.7	6
173 + 15	3	0.175	1
173 + 15	14	1	1

DVB-RCS2 standard specifies a layer-2 load control mechanism (see Section VI-C) aimed at keeping the collision rate below a given threshold. However, when using TCP on top of DVB-RCS2, the operating point G is determined by TCP congestion control algorithm; because of this, the load control algorithm is not used in this work. Every time a collision event causes data losses, f TCP segments are lost, leading to the retransmission of those segments and to a reduction of the TCP sending rate, according to TCP recovery algorithms.

In order to clarify how TCP congestion control works, we introduce the following concepts. A TCP *cycle* is defined as the time interval between two consecutive loss events (loss of TCP segments) and is equivalent to an *epoch* of the corresponding renewal process. The performance of TCP is modeled in terms of *rounds*: an epoch contains m rounds. A round begins with the transmission of W segments, called Congestion Window (CWND), and ends upon the receipt of the ACKs confirming the correct reception of these segments. The duration of a round is assumed to be independent of the CWND size and dependent on the Round-Trip Time (RTT). The CWND grows at each round, until a loss event occurs; then, TCP enters the recovery phase. Apart from the recovery algorithm that may vary according to the TCP flavor, TCP leaves the recovery phase with or without the expiration of the TimeOut (TO), thus entering a new epoch. The first rounds of the new epoch correspond to either a Slow Start (SS) or a Congestion Avoidance (CA) phase; the former if TO has expired, the latter otherwise. In [13], a Triple-Duplicate ACK (TD) is used as loss indication, and a TD Period (TDP) (i.e., the period between two TD loss indications) coincides with an epoch of the renewal process. TCP NewReno deals with segment losses relying on: (*i*) the use of the FR mechanism; (*ii*) the Time Out (TO) event, triggered if FR is not successful or if entering FR is not possible. The CWND is, then, set to a lower value and the Slow Start Threshold (SST) is set accordingly. In both cases, the sending rate is lowered trying to counteract further loss events. In this work, the satellite return/forward links are supposed operating in clear-sky conditions[5] in order to focus only on segment loss events caused by collisions (ACKs are assumed to be always received correctly). No Automatic

[5]DVB-RCS2 adopts adaptive modulation and coding to fulfill a target BER requirement, thus making feasible a quasi-error-free link assumption.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2018.2865150, IEEE Transactions on Aerospace and Electronic Systems

Repeat reQuest (ARQ) algorithm is in use at MAC layer. Each TCP segment sent to the lower layers (i.e., network and MAC layers) is queued into a large finite buffer so that buffer losses can be neglected w.r.t. segment losses due to collisions.

IV. Analytical Model of TCP NewReno

As preliminarily observed in [27], the TCP throughput estimation models available nowadays (e.g., [13], [14]) do not accurately match the empirical results of our TCP simulation campaigns, as also further discussed in Section VI, thus motivating additional study. In particular, the works in [14], [15] proposed two similar models for TCP NewReno, which present some little flaws that do not allow to accurately mimic the protocol dynamics in the case of multiple independent losses within the same congestion window (e.g., losses during the FR phase leading to the problem of *partial acknowledgments*). This aspect is particularly relevant in the understanding of the performance of TCP over RA satellite links, where multiple independent losses are very likely to occur at moderate/heavy traffic loads. To this end, we developed the model presented in what follows, namely *TCP-SAT-RA*, aimed at unifying the theoretical frameworks contained in [14], [15] with the needed extensions to properly characterise the case of multiple losses.

The presentation of the novel TCP NewReno model is given according to the following conceptual path:

1) the TCP-SAT-RA model is developed in Section IV-A assuming that all loss events are identified only by means of Triple-Duplicate ACKs (TDs), and then in Section IV-B handling both TDs and TOs (the *full model*);
2) the relation between RA-induced losses (i.e., because of collisions) and TCP dynamics is further studied in Section V, with the definition of the *BLR model*.

Finally, it is worth noting that no assumptions are made on the specific RA protocol in use, making the final findings quite general. Furthermore, given the IoT/M2M traffic profile under consideration, different TCP flavors may show basically the same performance level, because of the limited CWND size and the reduced available bandwidth, the latter being a typical case of satellite RA links.

A. Throughput model if loss indications are TDs only

In this section, we derive the model for estimating the TCP throughput in the absence of TOs.

The throughput T of a TCP flow can be estimated by analyzing a TCP cycle, an epoch of the renewal process. Let S_i^{CA} and S_i^{FR} be the number of segments successfully transmitted during the CA and FR phases of the i-th cycle, respectively, and S_i^{CAFR} the number of segments sent during a whole cycle, here denoted as Congestion Avoidance - Fast Recovery / Fast Retransmit (CAFR) [14]:

$$S_i^{CAFR} = S_i^{CA} + S_i^{FR}. \tag{1}$$

Let D_i^{CA} and D_i^{FR} denote the time duration of CA and FR periods and D_i^{CAFR} the duration of the i-th CAFR cycle:

$$D_i^{CAFR} = D_i^{CA} + D_i^{FR}. \tag{2}$$

During the CA phase, the receiver sends one ACK every b segments it receives (*delayed ACK* feature[6]), causing the CWND to increase linearly with a slope of $1/b$ segments per round, until the first segment loss occurs. It is worth recalling that a loss event occurs when at least one segment is lost in a CWND, thus triggering the FR phase (or a TO, as in Section IV-B). The first segment loss in a CWND marks the beginning of a *loss event*. Let us denote by α_i the position of the first lost segment in a drop window W_i of the i-th cycle. A *drop window* is defined as the CWND where the loss event has occurred. Let us also denote by X_i the round where this loss occurs. The total number of segments sent in the i-th cycle is $S_i^{CAFR} = \alpha_i + \gamma_i$, being γ_i the number of segments sent between the first loss α_i and the last one in the drop window W_i. The expected value of (1) results as:

$$S^{CAFR} = \mathbb{E}[\alpha] + \mathbb{E}[\gamma]. \tag{3}$$

The expected number of segments sent in a cycle, having k rounds, up to α_i, is given in [15] as:

$$\mathbb{E}[\alpha] = \sum_{k=1}^{\infty} k(1-p)^{k-1}p = \frac{1}{p}, \tag{4}$$

where p represents the average rate of loss events.

The condition for entering the FR phase is to successfully deliver at least three segments in the drop window W_i. Hence, let us assume $\delta_i \geq 1$ losses with rate q over the remaining $(W_i - 3)$ segments, where q corresponds to the average SLR. The number of losses δ_i follows a binomial distribution $B[u; v, \chi] = \binom{v}{u}\chi^u(1-\chi)^{v-u}$ over a drop window; in particular, let us consider the probability distribution of δ_i losses over $(W_i - 3)$ segments, conditioned on $\delta_i \geq 1$, as:

$$\begin{aligned} &Prob\{\delta_i = j \,|\, \delta_i \geq 1, W_i > 3\} = B[j-1; W_i - 4, q] \\ &= \binom{W_i - 4}{j-1}(1-q)^{W_i-3-j}q^{j-1}, j \in [1, W_i - 3]. \end{aligned} \tag{5}$$

Analogously to (4), the average number of segments sent between two consecutive losses is $1/q$. Therefore, if δ_i losses are assumed, the average number of segments among δ_i losses in the same CWND is $(\delta_i - 1)/q$. The expected value of γ can be calculated as follows:

$$\begin{aligned} \mathbb{E}[\gamma] &= \mathbb{E}\left[\sum_{j=1}^{W_i-3} \frac{(j-1)}{q}\left(q\, B[j-1; W_i - 4, q]\right)\right] \\ &= q\,(\mathbb{E}[W] - 4), \end{aligned} \tag{6}$$

where $\mathbb{E}[W]$ is the expected value of the CWND size, under the assumption of steady state, and $q\,B[j-1; W_i - 4, q]$ is the joint probability of the first loss α_i and of the other $(j-1)$ losses in the drop window. Note that $\mathbb{E}[\gamma]$ is defined only in the presence of multiple losses. If just a single segment loss occurs, then $\mathbb{E}[\gamma] = 0$; otherwise, if two or more segment losses occur, then $\mathbb{E}[\gamma] > 0$. When $q \to 1$, $\mathbb{E}[\gamma] \to (\mathbb{E}[W] - 4)$, i.e., the number of segments sent in a drop window after the first three segments allowing for TDs and the first loss α; conversely, when $q \to 0$, $\mathbb{E}[\gamma] \to 0$.

[6] As RFC 1122 suggests, the amount of traffic from the receiver to the sender should be reduced by sending a single ACK every b segments.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2018.2865150, IEEE Transactions on Aerospace and Electronic Systems

In order to derive the average number of rounds in a cycle, namely $\mathbb{E}[X]$, and the average CWND size, let us consider the evolution of CWND as a function of the number of rounds. According to [15] and [17], we have:

$$W_i = \frac{W_{i-1}}{2} + \frac{X_i}{b} - 1, \tag{7}$$

thus we can write the following relation, at regime, among average values:

$$\mathbb{E}[X] = b\left(\frac{\mathbb{E}[W]}{2} + 1\right). \tag{8}$$

The expected number of segments sent during the CA phase is determined in [17] as:

$$S^{CA} = \frac{3}{4}\,\mathbb{E}[X]\,\mathbb{E}[W] + \mathbb{E}[\beta], \tag{9}$$

where β_i is the number of segments sent in the last round of the *i*-th cycle. $\mathbb{E}[\beta] \approx \mathbb{E}[W]/2$ because we assume β to be uniformly distributed in $[1, W_i - 1]$.

The number of segments $S_{i,k}$ sent at a generic round k of the *i*-th FR cycle, considering the Partial Window Deflation[7] mechanism, can be derived from [32] as:

$$S_{i,k} = \max\Big(0,\, W_i/2 - \delta_i + k - 1\Big),\ 1 \le k \le \delta_i. \tag{10}$$

The start of an FR phase is due to the reception of a TD. When a partial ACK is received, CWND is decreased by the amount of data acknowledged and increased of one segment size. Therefore $S_{i,k+1} = S_{i,k} + 1$. $S_{i,k}$ has a maximum value for $k = \delta_i$, then $\max(S_{i,k}) = S_{i,\delta_i} = W_i/2 - 1$ [32]. Thus, the expected total number of segments sent in an FR cycle is obtained from (10) as follows:

$$\begin{aligned} S^{FR} &= \mathbb{E}\Big[\sum_{k=1}^{\delta_i} S_{i,k}\Big] \\ &= \begin{cases} \frac{1}{2}(\mathbb{E}[\delta]\,\mathbb{E}[W] - \mathbb{E}[\delta] - \mathbb{E}[\delta]^2), & \text{if } \mathbb{E}[\delta] < \mathbb{E}[W] \\ 0 & \text{if } \mathbb{E}[\delta] = \mathbb{E}[W]. \end{cases} \end{aligned} \tag{11}$$

In the case of a single loss, i.e., $\delta_i = 1$, $S^{FR} = \mathbb{E}[W]/2 - 1$. After the first loss, the remaining $(\delta_i - 1)$ segments are lost over $(W_i - 4)$ segments. Hence, we can write the following formula to characterize $\mathbb{E}[\delta]$, the expected value of the number of losses δ_i:

$$\mathbb{E}[\delta] = \mathbb{E}\left[\sum_{j=1}^{W_i-3} j\, B[j; W_i - 3, q]\right] \approx 1 + (\mathbb{E}[W] - 4)\, q. \tag{12}$$

Assuming that collisions on an RA channel are independent events and $f = 1$, then the segment losses are independent events, too. Instead, if $f > 1$, each collision in the channel causes a burst of segment losses, because a single MAC time slot carries up to f TCP segments, according to the number of segments ready for transmission at the start of the frame. The number of segments ready for transmission depends on TCP state: given that a time slot can fit up to f segments and at least a segment is enqueued, the burst of segment losses has a length in the interval $[1, f]$. Thus, if $f > 1$, TCP segment losses cannot be considered as independent ones because a single collision event can be responsible for multiple losses. Further details on this case are provided in Section VI-A2.

Using (9) and (11), we can express the expected number of segments sent during a CAFR cycle, i.e., the expected value of (1), as follows:

$$S^{CAFR} = S^{CA} + S^{FR}. \tag{13}$$

Then, the expected value of CWND, $\mathbb{E}[W]$, can be obtained by equating (3) and (13). In particular, we obtain a second order equation in $\mathbb{E}[W]$ (terms of order greater than two are neglected), whose only positive solution can be expressed as:

$$\mathbb{E}[W] \approx \Phi + \sqrt{\Phi^2 - \frac{60pq - 8p - 8}{p(8q + 3b)}}, \tag{14}$$

where $\Phi = \dfrac{22q - 3b - 4}{8q + 3b}$.

Obviously, Equation (14) must be defined for those values of p and q such that the argument of the square root is non-negative and $\mathbb{E}[W] \ge 1$, the latter condition coming from the fact that the congestion window is shrunk to 1 in case of a TO event. After few mathematical manipulations (omitted here for lack of space), it can be shown that these conditions are certainly met if $q \le 0.2, \forall p \in [0, 1]$, which fits the investigation domain of this paper, i.e., low-to-medium values of p and q[8]. Further considerations about the applicability of Equation (14) are drawn in Section V. It can also be highlighted that Equation (14) explicitly takes b into account, differently from Equation (14) in [14], where $b = 1$ is assumed. We will show in Section VI that, because of the finer estimations of S^{FR} and $\mathbb{E}[\delta]$, the expected value of CWND in (14) is closer to simulation results than in the other theoretical models under consideration [13]–[15].

Each RTT is supposed as a r. v., whose value does not depend on CWND, as in [13]–[15]. Figure 2 shows the evolution of CWND during CA and FR phases and the duration of the previous phases. A CAFR cycle ends with the end of the FR phase because of one or more segment losses. The duration of the last part of the CA phase is $D^\beta \cdot RTT$, where $\mathbb{E}[D^\beta]$ is the expected time, in number of rounds, after the first loss and before entering FR. $\mathbb{E}[D^\beta]$ can be approximated as $\mathbb{E}[D^\beta] \approx 1/2$. In fact, when entering FR, the last round of the CA phase ends prematurely with the reception of TDs. The two segments lost in Figure 2 are recovered during the FR phase that immediately follows, lasting $D^{FR} = 2$ RTTs. The CWND size is $(W_i/2 + 3)$ at the start of the FR phase and $W_i/2$ at the end of the FR phase. We recall that:

$$D^{CA} = \mathbb{E}[X] + \mathbb{E}[D^\beta], \tag{15}$$

where D^{CA} is the average duration, in number of rounds, of the CA phase, including the time spent waiting for entering

[7] RFC 6582: The NewReno modification to TCP's fast recovery algorithm.

[8] Actually, the range for p and q in Equation (14) is broader, also allowing for values of $q \in [0.25, 1]$, though with a reduced set of values for p. Nevertheless, larger values of q are of minor interest in this paper, since they would correspond to cases with heavy packet losses likely resulting from a mis-dimensioning of the satellite RA.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2018.2865150, IEEE Transactions on Aerospace and Electronic Systems

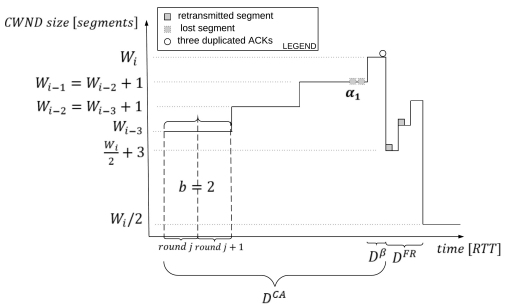


Fig. 2: Evolution of CWND during CA and FR phases.

in FR. Moreover, D^{FR} is the average duration, in number of rounds, of the FR phase:

$$D^{FR} = \mathbb{E}[\delta]. \tag{16}$$

In fact, NewReno recovers one lost segment per round in the FR phase. From (15) and (16), the expected duration of a cycle $D^{CAFR} = \mathbb{E}[D_i^{CAFR}]$ [s], results as:

$$D^{CAFR} = \mathbb{E}[RTT](D^{CA} + D^{FR}). \tag{17}$$

Combining (3) and (17), we can finally write the throughput expression [segments/s], neglecting TOs, as:

$$T_{noTO} = \frac{(\mathbb{E}[W] - 4)\, q + \frac{1}{p}}{\mathbb{E}[RTT]\left\{b(\frac{\mathbb{E}[W]}{2} + 1) + \frac{1}{2} + [1 + (\mathbb{E}[W] - 4)\, q]\right\}}, \tag{18}$$

where $\mathbb{E}[W]$ is expressed in (14).

B. Full Model

This section extends the model in Section IV-A to include also TOs as loss indications. When a TO occurs, an SS phase is needed before entering a new CA or CAFR phase (new cycle). TCP NewReno may experience a TO either during CA or during FR, given that a loss event (L) occurred with probability p. The former transition occurs with probability $p_{TOCA} = P\{TO|CA, L\}$, when TCP does not receive enough duplicate ACKs to trigger FR, while the latter transition occurs with probability $p_{TOFR} = P\{TO \,|\, FR,\, L\}$, when retransmitted segments are lost during the FR phase. TCP can experience a TO in the CA phase when more than $(W - 3)$ segments are lost in a drop window. The conditional probability p_{TOCA} is given by:

$$p_{TOCA} = \mathbb{E}\left[\sum_{j=W_i-2}^{W_i} \binom{W_i - 1}{j - 1}(1 - q)^{W_i - j} q^{j-1}\right]. \tag{19}$$

A TO occurs during FR if any of the retransmitted segments is lost. This condition can be approximated by assuming that if a loss event occurs in FR, then also the retransmitted segment is lost, thus triggering a TO [14]. For δ losses in the drop window, NewReno takes δ RTTs to recover the lost segments, during which it sends S^{FR} segments. The probability that the i-th segment is lost, given that the previous $(i - 1)$ segments are successfully delivered, is $(1-p)^{i-1}p$. Therefore, according to [14], it follows that:

$$p_{TOFR} = \mathbb{E}\left[\sum_{j=1}^{W_i-3} B[j; W_i - 3, q]\left[1 - (1 - p)^{j\frac{W_i}{2}}\right]\right]. \tag{20}$$

During a TO, TCP does not send any segments. According to [13], the duration of a TO period is given by:

$$D^{TO} = RTO\frac{1 + \sum_{j=0}^{5} 2^j p^{j+1}}{1 - p}, \tag{21}$$

where the initial value of the RTO parameter of TCP is 2 or 3 [s], in recent implementations (see RFC 6298, *Appendix A*).

In the SS phase, the initial CWND size is one and it grows until the SST, $W/2$, is reached. Therefore, in the last round of SS, TCP transmits $\mathbb{E}[W]/2$ segments, on average, then enters the CA phase. When using *delayed ACKs*, the CWND increases with rate $x = (1 + \frac{1}{b})$ on an RTT basis [33]:

$$\begin{aligned} S^{SS} &= \mathbb{E}\left[x^0 + x^1 + ... + x^{\log_x \frac{W_i}{4}}\right] = \mathbb{E}\left[b\left(x\,\frac{W_i}{4} - 1\right)\right] \\ &= b\left(x\frac{\mathbb{E}[W]}{4} - 1\right), \end{aligned} \tag{22}$$

where S^{SS} represents the expected value of the number of segments sent during the SS phase as in [14], considering that the segments of the last round of the SS phase as being part of the CA phase. D^{SS} represents the average duration of the SS phase, as in [14], but here explicitly considering b:

$$\begin{aligned} D^{SS} &= \mathbb{E}\left[RTT_i\left(\log_x \frac{W_i}{4} + 1\right)\right] \\ &\approx \mathbb{E}[RTT]\left(\log_x \frac{\mathbb{E}[W]}{4} + 1\right). \end{aligned} \tag{23}$$

Note that the expected value in (23) is applied to a non-linear expression. We have already assumed RTT_i and W_i as independent random variables. Moreover, if the distribution of W is concentrated around its mean, as shown in Section VI, then the approximation on the derivation of the expected value in (23) is acceptable.

The final expression of the throughput, also considering TOs, is shown in (24). It is worth remarking that some parameters, for instance $\mathbb{E}[W]$ provided in Section IV-A, are also used here, but they only consider CA and FR phases, without taking SS into account, thus providing an approximated estimate. We will show in Section VI that (18) provides a throughput estimation with a very small error w.r.t. (24) so that the improvement brought by (24) is almost negligible in the scenario under consideration.

V. The Relation between Burst Loss Rate and Event Loss Rate: the BLR Model

In this section, a simple but effective model is proposed in order to glue together the Burst Loss Rate (BLR) value experienced at MAC layer and p and q values seen at transport

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2018.2865150, IEEE Transactions on Aerospace and Electronic Systems

$$T_{Full} = \frac{(1 - p_{TOFR} - p_{TOCA})(S^{CA} + S^{FR}) + p_{TOCA}(S^{SS} + S^{CA}) + p_{TOFR}(S^{SS} + S^{CA} + S^{FR})}{(1 - p_{TOFR} - p_{TOCA})(D^{CA} + D^{FR}) + p_{TOCA}(D^{SS} + D^{CA} + D^{TO}) + p_{TOFR}(D^{SS} + D^{CA} + D^{FR} + D^{TO})}. \quad (24)$$

layer (*cross-layer* approach). An analytical model is proposed in [34], able to estimate the achievable throughput and BLR when CRDSA is in use with a given number of replicas. This model takes into account the iterative way in which SIC works on received RA blocks, but it just models the dynamics at the MAC layer, without considering the interactions with upper layer protocols. In this section, we propose a model that explicitly provides the relation among BLR at MAC layer with p and q at transport layer; to the best of our knowledge, such relation is still unexplored in the literature. In the following, we refer to *collisions* as to MAC bursts erased after SIC.

The collisions erase transmitted data, triggering the retransmissions at transport layer. Further to retransmitting data, TCP lowers the sending rate to decrease the probability that the same event occurs again, as described in Section IV. In the following, we provide two examples to ease the understanding of how the same number of segment losses can have different effects, according to the TCP behavior. As visible in Figure 3, if two consecutive collisions occur within the same CWND, a single loss event occurs (in Figure 3a), otherwise two loss events occur (in Figure 3b); thus, different p values are experienced. Let us call p_1 and p_2 the rates of loss events in Figures 3a and 3b, respectively: in the provided examples, it results that $p_1 < p_2$. Conversely, the q value reflects the number of segment losses within a CWND and, if q_1 and q_2 are the loss rates in Figures 3a and 3b, it results that $q_1 > q_2$. Equation (14) has a large sensitivity to the p value and a reduced sensitivity to the q value, thus the example in Figure 3b is subject to a larger reduction of the sending rate than the one in Figure 3a because $p_2 > p_1$.

Because of the different dynamics at MAC and at transport layer, which happen on very different time-scales, and because of the fact that TCP operates differently according to its internal state (SS, CA or FR), it is difficult to model the loss event rate p. In order to overcome this problem and to estimate the TCP throughput, the ***BLR model*** is proposed in what follows. For the sake of simplicity, we assume that one TCP segment fits exactly into one time slot ($r = 1$) and that spurious TCP retransmissions do not occur[9]. It follows that:

$$q \equiv BLR, \quad (25)$$

which means that each collision in the RA channel erases a single TCP segment. Moreover, Equation (25) confirms the consistency of Equation (14), which is defined at least when $q \leq 0.2$. In fact, when CRDSA++ operates at load levels up to the optimal working point [7], then it results that $q < 0.1$.

Hence, in the case of a finite CWND size and in virtue of Equation (25), it follows that:

$$p = \frac{q}{\mathbb{E}[\delta]} \approx \frac{q}{1 + (\mathbb{E}[W] - 4)\, q} \approx \frac{BLR}{1 + (\mathbb{E}[W] - 4)\, BLR}, \quad (26)$$

recalling that $\mathbb{E}[\delta]$ is the expected number of segments lost in a loss event. It can be observed that $\mathbb{E}[W]$ in the right side of (26) still depends on p as shown in (14). However, after a few mathematical manipulations, Equation (26) can be reformulated as $A(q)p^2 + B(q)p + C(q) = 0$, for which it is possible to find[10] a solution for $p = p(q) \in [0, 1]$. Therefore, using (25) and (26) we can obtain both p and q values on the basis of the single MAC-layer parameter BLR. These values can be used in (18) or (24) to obtain a TCP throughput estimation as a function of BLR. Formally speaking, if F is the function of the throughput estimation, $T = F(p, q, RTT)$ according to either (18) or (24), we can rewrite it as $T = F(BLR, RTT)$ thanks to the ***BLR model***. In Section VI-B, this model is validated by means of simulations.

VI. Performance Evaluation

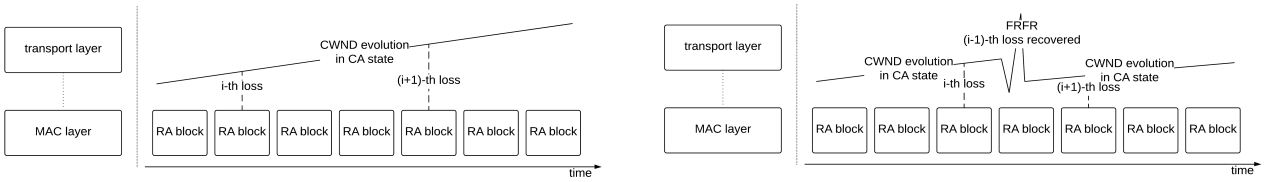
The performance evaluation shown here is based on the NS-3 simulator, implementing a DVB-S2/RCS2 satellite network by means of the modules described in [28]. A large number of nodes composing a WSN are connected to an RCST, which acts as a gateway forwarding received traffic in the satellite uplink. Multiple RCSTs (and thus multiple WSNs) are instantiated in our simulator (the actual numbers can be read in the following of this section). In other words, in order to simulate PUB/SUB-compliant data exchanges, data packets are sent to the gateway, then the local connection is closed; for the sake of simplicity and to avoid burdening the simulator, such a transmission does not make use of the TCP protocol in the simulator. We recall that it is assumed that local transmissions are always successful, thus neglecting any complexity on this side. On the other hand, a persistent TCP connection via satellite RA uplink is initiated by an RCST at the reception of the first data packet and kept alive until the end of each simulation round. Large and detailed statistics are collected per RCST at MAC and transport layers, then exploited in what follows to evaluate the achievable performance and to validate the proposed analytical model. The TCP flavor in use is NewReno. The available implementation in NS-3 is based on the SBS variant. Note that, as shown in [15], the performance of SBS is not significantly different from that of the Impatient variant in the conditions here under investigation, i.e., limited congestion window sizes and independent loss events. The main settings of the simulator are provided in Table IV.

We recall that the G^* load, i.e., the normalized MAC layer load achieving the highest MAC throughput, depends on the number of available time slots per RA block [35], among other factors. Thus, while keeping the other parameters unmodified, the choice of a different waveform can change the number of available time slots per RA block, shifting G^*. For instance, when 64 time slots are available, the G^* value of CRDSA++

[9]If not so, (25) does not hold. In fact, in the presence of spurious retransmissions, $q - BLR = q_{sr}$, where q_{sr} is the rate of spurious retransmissions: the larger q_{sr}, the larger the error due to (25).

[10]Obviously, equation $A(q)p^2 + B(q)p + C(q) = 0$ admits two solutions for p. However, it can be easily proven that when $q \leq 0.2$ there exists one positive solution for p belonging to the interval [0,1], whereas the other one is negative and therefore discarded.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2018.2865150, IEEE Transactions on Aerospace and Electronic Systems



(a) Two collisions occur: a single loss event is experienced at the transport layer because they affect the same CWND. The loss process has parameters p_1 and q_1 in this example.

(b) Two collisions occur: two loss events are experienced at transport layer because the losses are separated by an FR phase, which recovers a previously lost segment. The loss process has parameters p_2 and q_2 in this example.

Fig. 3: Examples of different interactions between the collisions in the RA channel and TCP congestion control algorithm in the presence of the same number of collisions.

TABLE IV: System settings

Name	Value
TCP flavor	NewReno
TCP MSSs	23 / 173 [B]
TCP/IP headers size	7 [B] (w/ ROHC)
Initial RTO value	2 [s]
RA scheme	CRDSA++ (3 replicas)
RA blocks per superframe	1
RA block duration	13 [ms]
number of time slots T_S (WF 14)	64
number of time slots T_S (WF 3)	194
Bandwidth	8012820 [Hz]
Roll off	0.2
Carrier spacing	0.3 [Hz]
Nominal Round Trip Time	0.52 [s]

is ≈ 0.7 [bursts/time slot]; when 194 time slots are available, the G^* value of CRDSA++ is ≈ 0.78 [bursts/time slot] [35]. The overall system has been tested at several increasing load levels, before, close to and after G^*. Two crucial aspects are investigated in this work: the MAC operating point $\hat{G}_T$ when NewReno is in use over an RA satellite channel[11], and the system stability; both issues are discussed in Section VI-C. To the best of the authors' knowledge, the characterization of $\hat{G}_T$ is not available in the literature for CRDSA++, if we exclude the preliminary contribution in [27].

The focus in what follows is on normalized load levels $G \geq 0.45$, because, up to this point, the CRDSA++ throughput is almost equal to the offered load (i.e., the operating point is in the linear part) [31]; beyond this load level, the system works in the quasi-linear part up to the G^* point, and then the MAC throughput dramatically decreases because of a too large BLR. In order to avoid instability effects when $G > G^*$ in RA satellite links [36], for instance, because of an ARQ algorithm running above the MAC layer, a load control algorithm must be used. TCP natively provides such a feature, coupled with automatic retransmissions.

[11]It is important to observe that G^* also corresponds to the normalized MAC layer load achieving the highest MAC throughput when no congestion/flow control is in use above MAC layer (e.g., UDP connections). On the other hand, $\hat{G}_T$ considers the case of TCP/IP on top of it, thus shifting $\hat{G}_T$ to the left of G^* (i.e., $\hat{G}_T \leq G^*$), because of the regulating effect introduced by TCP congestion control algorithms as preliminarily observed in [27].

A. Throughput validation

The TCP-SAT-RA model described in Section IV is now compared with simulation results and with other approaches in the literature as: (*i*) the work in [13], [17] (denoted as PFTK model); (*ii*) the work in [14] (denoted as PWM model); (*iii*) the work in [15] (denoted as DAKH model).

1) Accuracy of the CWND estimation

The CWND size and behavior is a critical aspect to be modeled. While just the expected value is used for the throughput estimation, CWND exhibits a range of different values during the whole TCP connection lifetime. Figure 4a shows the distribution of CWND obtained from simulations, when WF 14 and MSS = 173 bytes are selected; in the following, this scenario is used as an exemplary case. As the load increases (i.e., when the number of RCSTs increases), $\mathbb{E}[W]$ decreases, while the distribution keeps the same shape. A larger G value means higher load on the RA channel and then higher BLRs. Each time two or more RCSTs are involved in a collision, their CWND sizes are shrunk to a lower value, as in Figure 2 after the two lost segments. Therefore, $\mathbb{E}[W]$ decreases as G increases. In Table V, the simulation results are compared with the TCP-SAT-RA model estimations, showing a close match.

The average CWND value has a direct impact on the number of segments sent in a CAFR phase, as visible in (13). The distribution of the number of segments sent per CAFR phase is shown in Figure 4b: this is a geometric distribution, as pointed out in [13]. In a CAFR cycle, the sending rate varies if the connection is in the CA phase or in the FR phase: to underline this, Figure 4c shows the distribution of the number of segments sent in FR only. The distribution in the FR phase (in Fig. 4c) is different from the one in the CAFR phase (in Fig. 4b): the CA phase completely dominates the dynamics of a CAFR cycle, and it exhibits an approximate geometric distribution, while the FR phase has quite a negligible impact on the overall distribution, thus the approximation of a geometric distribution still stands. Table V also shows how the TCP-SAT-RA model can accurately estimate the average number of segments sent in CA and FR phases, along with their average durations.

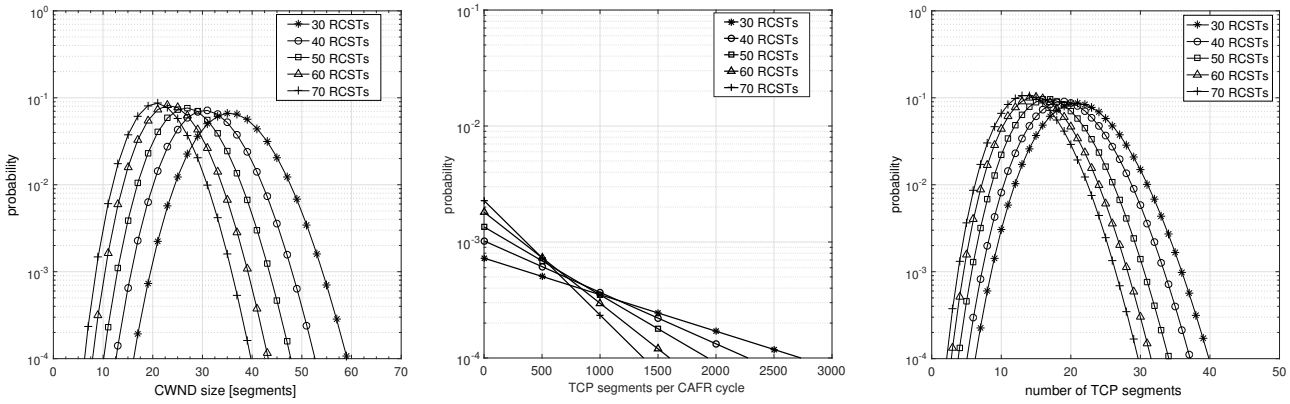
2) Average number of lost segments per loss event

A central part of this work is devoted to the characterization of the loss process experienced on the RA channel when CRDSA++ is in use. Equation (12) provides the analytical

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2018.2865150, IEEE Transactions on Aerospace and Electronic Systems

TABLE V: TCP-SAT-RA estimated values vs. simulation results
WF 14, MSS = 173 bytes, $T_S = 64$

#RCSTs	$\mathbb{E}[W]$ [segments]		S^{CA} [segments], D^{CA} [s]		S^{FR} [segments], D^{FR} [s]	
	simulated	model	simulated	model	simulated	model
30	40.15	41.8	1367.6, 25.5	1394, 28.4	19.3, 0.62	20.5, 0.66
40	33.4	34.8	1003.7, 21.4	978, 22.4	15.6, 0.58	17.1, 0.63
50	28.5	29.6	721.4, 17.9	716.6, 18.6	13.2, 0.58	14.4, 0.61
60	26.1	26	552.1, 15.6	559.2, 16.5	11.7, 0.58	12.6, 0.61
70	21.6	21.9	433.3, 13.9	406.1, 14.2	10.4, 0.58	10.6, 0.62



(a) Probability Mass Function of the CWND size

(b) Probability Mass Function of the number of segments sent per CAFR cycle

(c) Probability Mass Function of the number of segments sent per FR phase

Fig. 4: Simulation results of the scenario WF 14, MSS = 173 bytes, 64 time slots per RA block

expression of $\mathbb{E}[\delta]$, the expected number of losses per loss event. It ranges from 1 to 1.2 in our simulation results, as shown in Table VI, if we exclude the case of WF 14 and MSS = 23 bytes, discussed later. It means that, on average, 1 to 1.2 segments are lost per drop window: the same number of segments is retransmitted in the FR phase, which follows the loss event. Because of the reduced available bandwidth per RCST, the average CWND size is small and, on average, a single loss is experienced per drop window if q is small. The aforementioned average value supports our hypothesis that collisions can be considered as random independent events on an RA channel. Conversely, a strong correlation effect is present in the scenario with WF 14 and MSS = 23 bytes: a single time slot carries up to $r = 5.7$ segments, thus a single collision causes a burst of $f = 6$ losses at transport layer. Note that r is the maximum number of (whole or partial) TCP segments into a single time slot, and not the actual one, because the RLE protocol always tries to ensure the largest occupation of the time slot, but it is limited by the number of data units actually pending for transmission. It means that, if less than f data units are in the queue, a smaller number will be transmitted in a single time slot, thus under-utilizing the available resources. Moving to the performance evaluation, in this case, a single collision represents a very stressful event for a connection, because of the burst of losses experienced at TCP layer. This scenario exhibits very low performance because of this effect, when compared with the scenario WF 14 and MSS = 173 bytes and, generally, its performance is worse than any other configuration with $f = 1$. Because of this, we are not interested to the study of the TCP behavior for $f > 1$ (bursty losses due to encapsulation effects). For the same reason, Table VI does not show the $\mathbb{E}[\delta]$ values for $f > 1$, since its analytical expression is valid only under the hypothesis of independent losses.

3) Comparison among different approaches for estimating the TCP throughput

Figures 5, 6 and 7 compare the envisaged models for the analysis of TCP throughput over an RA satellite link using the CRDSA++ protocol. The relative error $\eta = \left|1 - \frac{T_{est}}{T_{sim}}\right|$ of the estimations is shown, where T_{est} is the throughput resulting from the analytical models under consideration and T_{sim} is the throughput resulting from simulations. These results are obtained with extensive Monte Carlo simulation runs: more than 1 million segments sent per scenario under consideration with a narrow confidence interval. η is computed for every combination of WF and MSS in use, except for the case WF 14 and MSS = 23 bytes ($f > 1$), as explained before. Let us recall that $b = 2$ (delayed ACKs) is adopted in this work; however, the PMW model does not account for it, so the relative error of this analytical model is larger.

Figure 5 shows the value of η for the scenario WF 3 and MSS = 23 bytes: $T_S = 194$ time slots are available for transmission and an RCST can use one time slot per RA block ($s = 1$). A segment fits exactly into one MAC burst ($r = 1$). The TCP-SAT-RA model proposed in this work achieves the lowest η value (i.e., the best results), if we exclude the case with the largest number of RCSTs; the DAKH model and the PFTK one are quite accurate, too, but the former tends to

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2018.2865150, IEEE Transactions on Aerospace and Electronic Systems

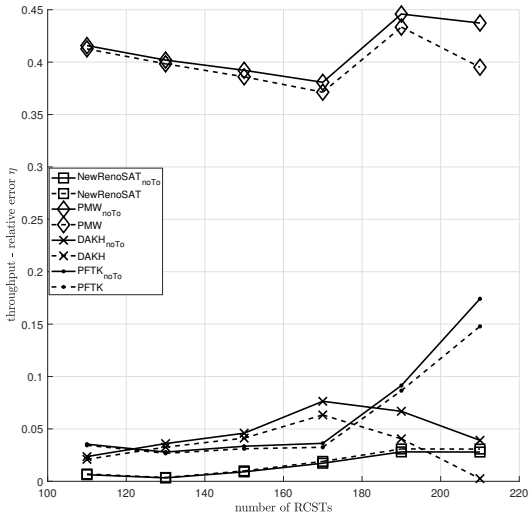


Fig. 5: Comparison (relative error) between simulations and theoretical approaches for TCP NewReno throughput over CRDSA++ for WF 3, MSS = 23 bytes and $T_S = 194$

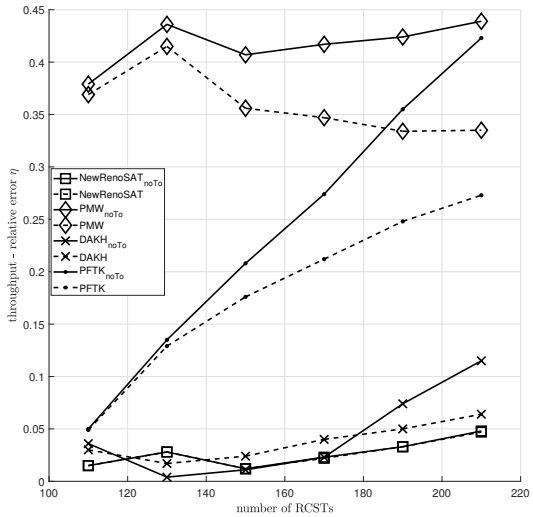


Fig. 6: Comparison (relative error) between simulations and theoretical approaches for TCP NewReno throughput over CRDSA++ for WF 3, MSS = 173 bytes and $T_S = 194$

overestimate the throughput at medium loads and the latter has an η value larger than 0.15 for higher loads.

In Figure 6, the relative error is shown for the scenario WF 3 and MSS = 173 bytes. In this scenario, a time slot carries a fraction of the segment: in fact, $r = 0.175$, so that 5.7 time slots are needed to carry a single segment, which means that the sixth time slot carries the last fraction of a segment and the initial fraction of the next one. A collision involving the sixth time slot causes the loss of two TCP segments, thus triggering two retransmissions for a single collision event. This happens with probability equal to $1/6$: because of this, the error on the estimation is larger than the one in Figures 5 and 7, for each model. In this scenario, there is some correlation among losses, so the hypothesis of complete independence among segment losses is not verified. Since anyway the correlation is small, simulation results show a good match with TCP-SAT-RA and DAKH analytical models.

Figure 7 provides the throughput estimation for the scenario WF 14 and MSS = 173 bytes: one segment fits exactly into a single MAC burst ($r = 1$). TCP-SAT-RA and DAKH models achieve the best results, even if DAKH shows an average relative error that is larger at high loads. Instead, the PFTK model exhibits a larger error because it does not take the FR mechanism into account.

4) Retransmission TimeOuts

In this section, the measure ξ is introduced to quantify the rate of TOs in the simulations:

$$\xi = \frac{\mathbb{E}[\#TOs]}{T_{sim}/\mathbb{E}[RTT]}, \tag{27}$$

where $\mathbb{E}[\#TOs]$ is the average number of TOs per connection and T_{sim} is the simulation length in seconds ($T_{sim}/\mathbb{E}[RTT]$ is the number of rounds in each simulation). Therefore, ξ represents the probability that a TO occurs in a round. The numerical values of ξ per scenario under consideration are shown in the last column of Table VI. We observe two

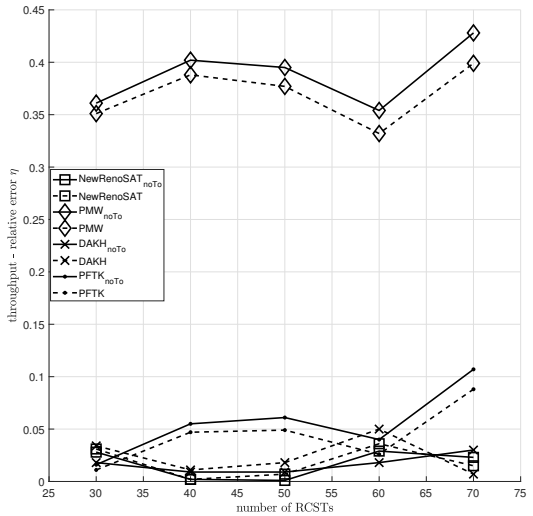


Fig. 7: Comparison (relative error) between simulations and theoretical approaches for TCP NewReno throughput over CRDSA++ for WF 14, MSS = 173 bytes and $T_S = 64$

effects worth of mentioning: if $r = 1$, ξ has a substantially-lower value with respect to the cases where $r \neq 1$; in other words, MAC fragmentation and packing significantly impact on TCP performance. The second effect here discussed is related to the load level: the lowest load level exhibits a ξ value larger than that of higher ones, which is unexpected, at first glance. At high loads, collisions are responsible of sustained segment losses, thus the ξ value reflects this. At medium loads, collisions are responsible for moderate losses, so that ξ shows a lower value. In fact, the FR mechanism is particularly efficient in the presence of a moderate loss rate, and time-consuming TOs are avoided. Instead, at low loads, another effect is present, which makes ξ larger than expected:

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2018.2865150, IEEE Transactions on Aerospace and Electronic Systems

the CWND overshooting problem. A collision is a very rare event at low loads, therefore the overshooting problem is possible: a CWND increasing slowly but continuously over time and whose value goes beyond the Bandwidth-Delay product increases the probability of spurious retransmission TOs if large buffers are implemented at the MAC layer.

B. Validating the BLR model

The *BLR model* discussed in Section V is now validated against simulation results. The scenario WF 14 and MSS = 173 bytes has been chosen because of the very low rate of spurious retransmissions exhibited during the simulation phases.

We compare the simulation results with the estimations provided by the models in (18) and (24), and then we test the accuracy of the BLR model by plugging (25) and (26) into the two previous equations. Practically speaking, we firstly picked the values of p and q measured during the simulations (in Table VI) and then plugged them into (18) and (24). The accuracy of the TCP throughput estimation with respect to the values measured during the simulations is then captured using the relative error η, which is shown in the first two columns of Table VII. On the other hand, as far as the BLR model validation is concerned, we picked the values of BLR measured in the simulations (in Table VI), applied them in (25) and (26) to obtain p and q, which have been in turn used in (18) and (24) to compute the corresponding TCP throughput estimations. The accuracy of the BLR model is given by the relative error η, which is shown in the last two columns of Table VII. The *BLR model* achieves a good precision: in fact, the larger estimation error is lower than 7% (we recall that spurious retransmissions are neglected). Because of this, a fraction of the estimation error depends on the fact that the NewReno implementation in NS-3 does not provide any algorithms to prevent such a phenomenon. A consideration is here in order to motivate the need of the BLR model: while it can be practical to estimate BLR in real systems, the same cannot be said for p when using TCP, because it depends on TCP inner dynamics. Thus, this model offers a practical way for relating p to MAC parameters to estimate TCP throughput.

C. Stability

A critical issue to be taken into account, when dealing with RA protocols, is system stability. CRDSA++, as other RA protocols, has an optimal working point, namely G^*, which exhibits the maximum throughput offered by the MAC protocol. If the system is forced to work at loads $G > G^*$, instability may occur and proper countermeasures are needed, as analysed in [36].

The DVB-RCS2 standard includes a normative load control algorithm, aimed at keeping the system at a target operating point $G_T \leq G^*$. This load control algorithm is not used in our study because the work in [37] has already exhaustively shown that the algorithm above needs to be tuned to each scenario under consideration, thus introducing important scalability and flexibility issues. The DVB-RCS2 load control algorithm can limit the magnitude of load oscillations around G_T according to two different strategies [37]: by reducing the number N of RCSTs or by shifting G_T to a lower value. These oscillations can be responsible for pushing the instantaneous offered load G beyond G^*, thus leading the system towards instability. Both strategies have disadvantages: in an IoT/M2M scenario, a large population of terminals (RCSTs) is common, so that reducing N could be unfeasible. Instead, shifting G_T towards a lower value can be a better approach in our scenario, but there is the need of a centralized NCC, which periodically monitors the aggregate load G measured at the MAC layer. This in turn has to be used by the NCC to select and provide the proper G_T values to the active RCSTs. This interaction requires however many iterations to compensate for the traffic oscillations and adjust the G_T value to prevent instability, as already pointed out in [37]. Figure 8a, described later, shows that the use of TCP on top of CRDSA++ removes the need of a centralized control: TCP can control G_T. Hence, TCP ensures the stability without requiring any layer-2 control mechanisms that can improperly interact with its congestion control algorithm.

Figure 8a shows the number of RCSTs versus the normalized MAC throughput and normalized MAC offered load, when using TCP on top of WFs 14 and 3. We recall that WF 3 offers a greater number of time slots with small payloads, while WF 14 offers a lower number of time slots with larger payloads. Three load intervals can be seen in Figure 8a w.r.t. the number of RCSTs: the first one, at low loads, up to ≈ 100 RCSTs, where WF 14 offers a larger normalized throughput; the second one, at medium loads, from ≈ 100 to ≈ 350 RCSTs, where WF 3 outperforms WF 14 thanks to the greater number of time slots; finally, the third interval, at larger loads (from ≈ 350 RCSTs on), where the throughput offered by the two WFs is comparable. A consideration is here in order: when using WF 3, TCP behaves greedily, showing a clear load peak for about 150 RCSTs, then the load decreases as the number of RCSTs increases. On the contrary, WF 14 shows a more balanced behavior, and a peak is not recognizable.

Let us call $\hat{G}_T$ the load working point for TCP NewReno on top of the CRDSA++ protocol. The following ranges of $\hat{G}_T$ can be read on the y-axis of Figure 8a, according to the WF in use: (*i*) $\hat{G}_T \in [0.45, 0.55]$ for TCP over WF 14 (with 64 available time slots); (*ii*) $\hat{G}_T \in [0.58, 0.63]$ for TCP over WF 3 (with 194 available time slots). When CRDSA++ is in use and assuming no flow control algorithms (i.e., TCP is not used), $G^* \approx 0.7$ when 64 time slots are available per RA block [35], and $G^* \approx 0.78$ when 194 time slots are available per RA block [35]. Thus, Figure 8a shows that $\hat{G}_T \leq G^*$: the TCP congestion control algorithm shift $\hat{G}_T$ towards lower load levels, which exhibit lower loss rates.

To better understand the capability of TCP to self-regulate its transmission rate, let us introduce the offered MAC load per time slot per RCST, contributed by N distinct RCSTs with N corresponding TCP flows, and defined as $\lambda = G/(s\,N\,T_S)$ [bursts/RCST/time slot] (s and T_S being the number of bursts per frame and the number of time slots respectively). The behavior of λ as a function of the number of RCSTs is shown in Figure 8b. In particular, we note that, in the presence of few RCSTs (up to 100, approximately), a reduced number of larger time slots (case WF 14) should be preferred to allow that each RCSTs takes as much advantage as possible from the available

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2018.2865150, IEEE Transactions on Aerospace and Electronic Systems

TABLE VI: Simulation results: the scenarios are described in the first column. $\mathbb{E}[RTT]$ is measured in seconds and $Thr.$ is the average throughput [kbps] per TCP connection from simulations

WF, MSS	$\#RCSTs$	BLR	r	f	q	p	$\mathbb{E}[\delta]$	$\mathbb{E}[RTT]$	$Thr.$	ξ
3, 23	110	1.1E-4	1	1	1.44E-3	4.5E-4	1.07	0.69	12.50	5.5E-4
3, 23	130	2E-4	1	1	1.70E-3	4.86E-4	1.08	0.68	11.90	3E-5
3, 23	150	5.5E-4	1	1	2.80E-3	6.68E-4	1.11	0.62	9.60	6.2E-5
3, 23	170	9.2E-4	1	1	5.13E-3	9.86E-4	1.16	0.61	7.84	5E-4
3, 23	190	1.1E-3	1	1	7.94E-3	1.20E-3	1.20	0.60	6.86	2.5E-3
3, 23	210	2.9E-3	1	1	1.1E-2	6.3E-3	1.20	0.59	2.90	4E-3
3, 173	110	1.2E-4	0.175	1	4.1E-3	1.5E-3	1.09	2.40	11.94	1E-1
3, 173	130	3.5E-4	0.175	1	8.66E-3	3.5E-3	1.14	1.50	11.90	9E-2
3, 173	150	1.4E-3	0.175	1	2.35E-2	1.1E-2	1.16	0.92	10	7E-2
3, 173	170	2.8E-3	0.175	1	2.94E-2	1.72E-2	1.17	0.77	8.50	5E-2
3, 173	190	4.4E-3	0.175	1	3.3E-2	2.7E-2	1.18	0.72	7.30	6E-2
3, 173	210	6.16E-3	0.175	1	3.5E-2	3.5E-2	1.20	0.69	6.37	7E-2
14, 23	30	2.27E-3	5.7	6	1.3E-2	3.7E-4	-	1.28	10.5	5E-2
14, 23	40	2E-3	5.7	6	1.38E-2	3.8E-4	-	1.275	10.48	4E-2
14, 23	50	2.32E-3	5.7	6	1.64E-2	4.2E-4	-	1.23	10.1	4E-2
14, 23	60	2E-3	5.7	6	1.71E-2	4.3E-4	-	1.19	9.93	4E-2
14, 23	70	1.9E-3	5.7	6	1.76E-2	4.7E-4	-	1.15	9.72	4E-2
14, 173	30	6.41E-4	1	1	8.52E-4	7.45E-4	1.03	0.64	65.75	5.3E-4
14, 173	40	9.26E-4	1	1	1.34E-3	1E-3	1.04	0.60	58.50	1E-4
14, 173	50	1.31E-3	1	1	1.75E-3	1.37E-3	1.04	0.58	51.51	1.4E-4
14, 173	60	1.76E-3	1	1	2.30E-3	1.79E-3	1.05	0.58	45.20	3E-4
14, 173	70	2.3E-3	1	1	3.5E-3	2.28E-3	1.07	0.58	38.11	4.6E-4

TABLE VII: WF 14, MSS = 173 bytes, $T_S = 64$. Relative error η when comparing the accuracy of the model presented in Section IV and, in the last two columns, when plugging the *BLR model* into (18) and (24)

$\#RCSTs$	$TCP-SAT-RA_{noTO}$	$TCP-SAT-RA$	$BLRmodel_{noTO}$	$BLRmodel$
30	0.028	0.031	0.063	0.069
40	0.002	0.002	0.058	0.067
50	0.001	0.007	0.040	0.047
60	0.029	0.036	0.015	0.021
70	0.023	0.015	0.046	0.051

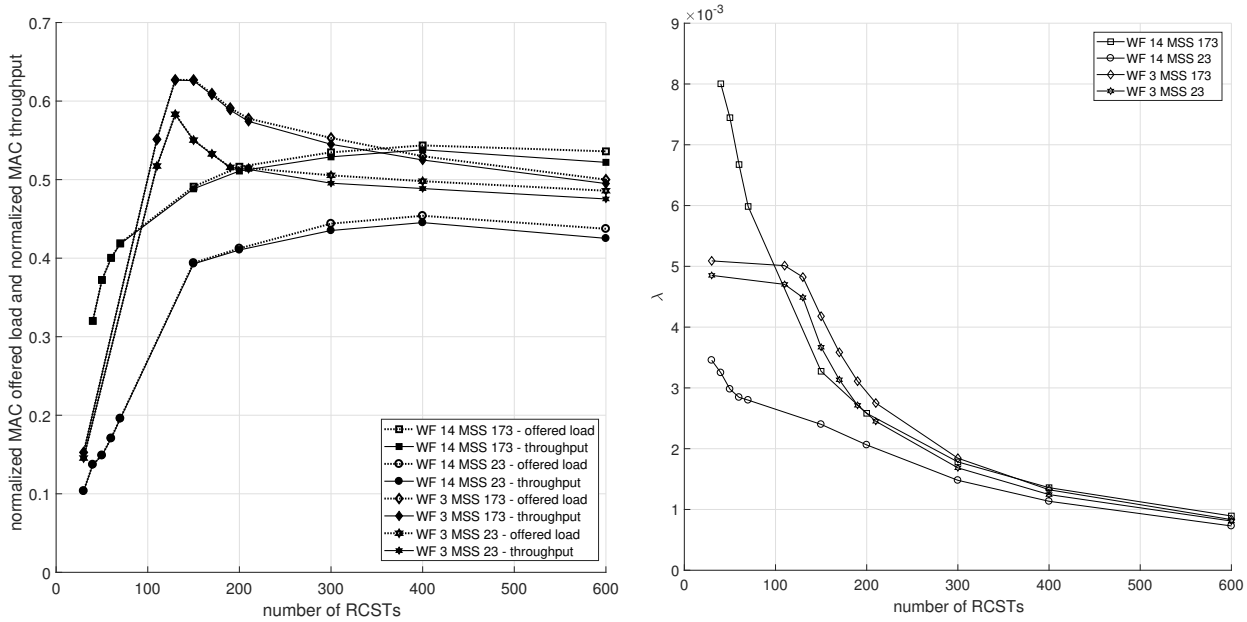
resources. On the other hand, when the number of RCSTs increases, a WF that offers more time slots (case WF 3) should be preferred to allow that a larger population uses the channel. It is worth noting in Figure 8b that λ has approximately the same value for $N > 400$, independently from WF and MSS. The previous recommendations are recollected in Table VIII, which summarizes the proposed guidelines in the presence of reliable M2M communications over a RA satellite link: it must be noted that, in the presence of high loads, the choice must be made accordingly to the average payload length. In fact, if the latter is larger than that provided by WF 3, a WF providing larger time slots should be chosen in order to reduce the fragmentation of layer-3 datagrams at MAC layer (e.g., WF 14); conversely, if the average payload length is comparable with that provided by WF 3, this WF must be preferred to WF 14 because of the large number of time slots, which allow accommodating a large number of simultaneous data exchanges. Finally, Figure 8b confirms that the scenario with WF 14 and MSS = 23 bytes experiences a low performance and a significant under-utilization of the available resources: thus, the choice of a WF and a MSS value that leads to $f > 1$ should be avoided.

VII. Conclusions

In this work, the performance of TCP has been analyzed over a satellite random access MAC. Our proposed throughput estimation model, namely TCP-SAT-RA, accurately fits the simulation results for the satellite scenario under consideration, where losses are only due to MAC collisions. Furthermore, our approach has been compared with other models in the literature, showing that it achieves a lower estimation error. Our simulation results support the hypothesis that collisions can be considered as random independent events on an RA channel. Finally, a simple but effective *cross-layer* model has been proposed to estimate the loss event rate p and the segment loss rate q at transport layer; this is the *BLR model* that has allowed a close match with simulation results, when used in the TCP-SAT-RA model.

In addition, the use of TCP enforces stability in RA channels, removing the need of the DVB-RCS2 load control mechanism, which is strongly scenario-dependent. TCP acts in a distributed way, without the need of a centralized entity in charge of enforcing complex load control strategies. The use of different waveforms and MSS sizes have been analyzed to identify the combination providing the best performance in the IoT/M2M scenario under consideration, also depending on the number of RCSTs. Furthermore, it has been shown that any system configuration leading to the fragmentation of TCP segments at MAC layer should be avoided because of poor performance.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2018.2865150, IEEE Transactions on Aerospace and Electronic Systems



(a) Aggregate normalized MAC throughput and aggregate offered load vs. number of RCSTs.

(b) Offered load λ per RCST per time slot vs. number of RCSTs.

Fig. 8: MAC offered load and throughput for different WFs and MSSs.

TABLE VIII: Suggested configurations when TCP-based M2M data exchanges occur on top of CRDSA++ in DVB-RCS2-compliant RA satellite channels.

	Low loads (up to about 100 active RCSTs)	**Medium loads (up to about 400 active RCSTs)**	**High loads (more than 400 active RCSTs)**
suggested configuration	reduced time slots number with large payloads (e.g., WF 14)	large time slots number with small payloads (e.g., WF 3)	the WF must be chosen accordingly to the average payload length

REFERENCES

[1] J. Kim, J. Lee, J. Kim, and J. Yun, "M2M service platforms: survey, issues, and enabling technologies," *Communications Surveys & Tutorials, IEEE*, vol. 16, no. 1, pp. 61–76, 2014.

[2] O. D. R. Herrero and R. De Gaudenzi, "High efficiency satellite multiple access scheme for machine-to-machine communications," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 4, pp. 2961–2989, 2012.

[3] S. Scalise, C. P. Niebla, R. De Gaudenzi, O. del Rio Herrero, D. Finocchiaro, and A. Arcidiacono, "S-MIM: a novel radio interface for efficient messaging services over satellite," *Communications Magazine, IEEE*, vol. 51, no. 3, pp. 119–125, 2013.

[4] P. Cerwall, P. Jonsson, R. Möller, S. Bävertoft, S. Carson, I. Godor, P. Kersch, A. Kälvemark, G. Lemne, and P. Lindberg, "Ericsson mobility report," *available: www.ericsson.com/res/docs/2015/ericsson-mobility-report-june-2015.pdf*, June 2015.

[5] A. Laya, L. Alonso, and J. Alonso-Zarate, "Is the random access channel of LTE and LTE-A suitable for M2M communications? A survey of alternatives," *Communications Surveys & Tutorials, IEEE*, vol. 16, no. 1, pp. 4–16, 2014.

[6] R. De Gaudenzi, O. Del Rio Herrero, G. Gallinaro, S. Cioni, and P. D. Arapoglou, "Random access schemes for satellite networks, from VSAT to M2M: a survey," *International Journal of Satellite Communications and Networking*, vol. 36, no. 1, pp. 66–107, 2018.

[7] E. Casini, R. D. Gaudenzi, and O. Herrero, "Contention Resolution Diversity Slotted ALOHA (CRDSA): An enhanced random access scheme for satellite access packet networks," *IEEE Transactions on Wireless Communications*, vol. 6, no. 4, pp. 1408–1419, 2007.

[8] C. Liu and E. Modiano, "An analysis of TCP over random access satellite links," in *IEEE Wireless Communications and Networking Conference*, vol. 4, March 2004, pp. 2033–2040.

[9] M. Bacco, A. Gotta, C. Roseti, and F. Zampognaro, "A study on TCP error recovery interaction with Random Access satellite schemes," in *Advanced Satellite Multimedia Systems Conference and Signal Processing for Space Communications Workshop*. IEEE, 2014, pp. 405–410.

[10] M. Collina, M. Bartolucci, A. Vanelli-Coralli, and G. E. Corazza, "Internet of Things application layer protocol analysis over error and delay prone links," in *Advanced Satellite Multimedia Systems Conference and Signal Processing for Space Communications Workshop*. IEEE, 2014, pp. 398–404.

[11] M. Bacco, T. De Cola, G. Giambene, and A. Gotta, "Advances on Elastic Traffic via M2M Satellite User Terminals," in *International Symposium on Wireless Communication Systems*. IEEE, 2015, pp. 226–230.

[12] M. Bacco, M. Colucci, and A. Gotta, "Application Protocols Enabling Internet of Remote Things via Random Access Satellite Channels," in *International Conference on Communications*. IEEE, 2017, pp. 1–6.

[13] J. Padhye, V. Firoiu, D. F. Towsley, and J. F. Kurose, "Modeling TCP Reno performance: a simple model and its empirical validation," *IEEE/ACM Transactions on Networking (ToN)*, vol. 8, no. 2, pp. 133–145, 2000.

[14] N. Parvez, A. Mahanti, and C. Williamson, "An Analytic Throughput Model for TCP NewReno," *IEEE/ACM Transactions on Networking (ToN)*, vol. 18, no. 2, pp. 448–461, 2010.

[15] R. Dunaytsev, K. Avrachenkov, Y. Koucheryavy, and J. Harju, "An Analytical Comparison of the Slow-but-Steady and Impatient Variants of TCP New Reno," in *Wired/Wireless Internet Communications*. Springer, 2007, pp. 30–42.

[16] K. Zhou, K. Yeung, and V. Li, "Throughput modeling of TCP with slow-start and fast recovery," in *Global Telecommunications Conference*, vol. 1, Nov 2005, p. 5 pp.

[17] Z. Chen, T. Bu, M. Ammar, and D. Towsley, "Comments on modeling TCP Reno performance: a simple model and its empirical validation," *IEEE/ACM Transactions on Networking (ToN)*, vol. 14, no. 2, pp. 451–453, 2006.

[18] M. Luglio, C. Roseti, and F. Zampognaro, "Transport Layer Optimization for Cloud Computing Applications via Satellite: Tcp Noordwijk+," *China Communications*, vol. 11, no. 12, pp. 105–119, 2014.

[19] A. Abdelsalam, M. Luglio, C. Roseti, and F. Zampognaro, "TCP Wave: A New Reliable Transport Approach for Future Internet," *Computer Networks*, vol. 112, pp. 122–143, 2017.

[20] N. Kuhn, H. C. Bui, J. Lacan, J. Radzik, and E. Lochin, "On the Trade-off between Spectrum Efficiency with Dedicated Access and Short end-to-end Transmission Delays with Random Access in DVB-RCS2," in *Proceedings of ACM MobiCom workshop on Lowest cost denominator networking for universal access*, Florida, USA, Sep. 2013, pp. 47–52.

[21] M. Bacco, A. Gotta, M. Luglio, C. Roseti, and F. Zampognaro, "A new RA-DA hybrid MAC approach for DVB-RCS2," in *International Black Sea Conference on Communications and Networking*. Constanta, Romania: IEEE, 2015, pp. 215–219.

[22] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP Throughput: a Simple Model and its Empirical Validation," in *ACM SIGCOMM Computer Communication Review*, vol. 28, no. 4. ACM, 1998, pp. 303–314.

[23] Y. Sun, Z. Ji, and H. Wang, "TFRC-satellite: a TFRC Variant with a Loss Differentiation Algorithm for Satellite Networks," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 2, pp. 716–725, 2013.

[24] M. Bacco, P. Cassarà, M. Colucci, and A. Gotta, "Modeling Reliable M2M/IoT Traffic over Random Access Satellite Links in Non-saturated Conditions," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 4, pp. 1042 – 1051, 2018.

[25] N. Celandroni and R. Secchi, "Suitability of DAMA and contention-based satellite access schemes for TCP traffic in mobile DVB-RCS," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 4, pp. 1836–1845, May 2009.

[26] N. Celandroni, F. Davoli, E. Ferro, and A. Gotta, "On Elastic Traffic via Contention Resolution Diversity Slotted Aloha Satellite Access," *International Journal of Communication Systems*, vol. 29, no. 3, pp. 522–534, 2016.

[27] M. Bacco, T. De Cola, and A. Gotta, "TCP New Reno over DVB-RCS2 Random Access Links: Performance Analysis and Throughput Estimation," in *Global Communications Conference*. IEEE, 2015, pp. 1–6.

[28] J. Puttonen, S. Rantanen, F. Laakso, J. Kurjenniemi, K. Aho, and G. Acar, "Satellite Model for Network Simulator 3," in *Proceedings of the 7th International ICST Conference on Simulation Tools and Techniques*, Lisboa, Portugal, March 2014, pp. 86–91.

[29] M. Laner, N. Nikaein, D. Drajic, P. Svoboda, M. Popovic, and S. Krco, *Traffic models for machine-to-machine (M2M) communications: types and applications*. Machine-to-machine communications, architecture, performance and applications, Woodhead Publishing, 04 2014.

[30] A. Arcidiacono, D. Finocchiaro, F. Collard, S. Scalise, F. Lázaro Blasco, R. De Gaudenzi, S. Cioni, N. Alagha, and M. Andrenacci, "From S-band mobile interactive multimedia to fixed satellite interactive multimedia: making satellite interactivity affordable at Ku-band and Ka-band," *International Journal of Satellite Communications and Networking*, vol. 34, no. 4, pp. 575–601, 2016.

[31] O. del Rio Herrero and R. De Gaudenzi, "A high-performance MAC protocol for consumer broadband satellite systems," in *27th IET and AIAA International Communications Satellite Systems Conference (ICSSC)*. IET, 2009, pp. 512–527.

[32] C. Joo and S. Bahk, "Start-up transition behaviour of TCP NewReno," *Electronics Letters*, vol. 35, no. 21, pp. 1818–1820, Oct 1999.

[33] N. Cardwell, S. Savage, and T. Anderson, "Modeling TCP Latency," in *Annual Joint Conference of the Computer and Communications Societies*, vol. 3. IEEE, 2000, pp. 1742–1751.

[34] O. Del Rio Herrero and R. De Gaudenzi, "Generalized Analytical Framework for the Performance Assessment of Slotted Random Access Protocols," *IEEE Transactions on Wireless Communications*, no. 99, pp. 1–13, 2014.

[35] M. Bacco, P. Cassarà, E. Ferro, and A. Gotta, "Generalized Encoding CRDSA: Maximizing Throughput in Enhanced Random Access Schemes for Satellite," in *Personal Satellite Services*. Springer, 2013, pp. 115–122.

[36] A. Meloni and M. Murroni, "Random Access in DVB-RCS2: Design and Dynamic Control for Congestion Avoidance," *IEEE Transactions on Broadcasting*, vol. 60, no. 1, pp. 16–28, 2014.

[37] A. Munari, G. Acar, C. Kissling, M. Berioli, and H. P. Lexow, "Multiple access in DVB-RCS2 user uplinks," *International Journal of Satellite Communications and Networking*, vol. 32, no. 5, pp. 359–376, 2014.

Manlio Bacco received the Ph.D. degree in Information Engineering and Science from the University of Siena in 2016. He works as a researcher at the Institute of Science and Information Technologies (ISTI) since 2012, at the National Research Council (CNR), Pisa, Italy. His research interests include wireless communications, with a focus on satellite and aerial networks, and IoT/M2M communications. He has been participating in European-funded, national-funded, and ESA-funded research and development projects.

Tomaso de Cola was born in Manosque, France, on April 28, 1977. He received the "Laurea" degree (with honors) in telecommunication engineering, in 2001, the Qualification degree as Professional Engineer in 2002 and the Ph. D. degree in Electronic and Computer Engineering, Robotics and Telecommunications in 2010 from the University of Genoa, Italy. From 2002 until 2007, he has worked with the Italian Consortium of Telecommunications (CNIT), University of Genoa Research Unit, as scientist researcher. Since 2008, he has been with the German Aerospace Center (DLR), where he is involved in different European Projects focusing on different aspects of DVB standards, CCSDS protocols and testbed design. In particular, he has been serving since 2015 as Deputy Area Director of the Space Internetworking Services (SIS) area within CCSDS and taking part to the CCSDS Engineering Steering Group (CESG). He is member of the IEEE Communications Society, where he is serving as chair of the Satellite and Space Communications (SSC) technical committee.

Giovanni Giambene received the Ph.D. degree in Telecommunications and Informatics in 1997 from the University of Florence, Italy. He was Technical External Secretary of the European Community COST 227 Action ("Integrated Space/Terrestrial Mobile Networks"). From 1997 to 1998, he was with OTE (Marconi Group) in Florence, Italy, working on a GSM development program. In 1999, he joined the Department of Information Engineering and Mathematical Sciences of the University of Siena, Italy, where, he is associate professor on Networking. He was vice-Chair of the COST 290 Action (2004-2008), entitled "Traffic and QoS Management in Wireless Multimedia Networks" (Wi-QoST). He participated to several EU projects (SatNEx, RADICAL, COST Action WiNeMO, and RESPONSIBILITY). At present, he is involved in the ESA SatNEX IV project. Giambene is IEEE senior member. Since January 2015, he is IEEE Transactions on Vehicular Technology editor. His research interests deal with wireless and satellite networking, cross-layer air interface design, and transport layer performance.

Alberto Gotta is staff member of the Institute of Science and Information Technologies (ISTI), at the National Research Council (CNR), Pisa, Italy. He has been the research leader of the Communication Protocols and Architectures section of the Wireless Network Laboratory at ISTI, from 2015. He is also project reviewer for the Italian Ministry of Economic Development, from 2013. His expertise is mainly related to traffic engineering applied to satellite, M2M, and UAV networks. He has been participating in European-, National-, Regional-, and ESA-funded R&D projects.