EAI Endorsed Transactions
on Mobile Communications and Applications

Research Article ICST.ORG

Generalized Encoding CRDSA: Maximizing Throughput in Enhanced Random Access Schemes for Satellite

Manlio Bacco[1 2,*], Pietro Cassará[1], Alberto Gotta[1]

[1]National Research Council of Italy (CNR), Institute of Information Science and Technologies (ISTI)
[2]University of Siena, Information Engineering and Mathematic Science Department

Abstract

This work starts from the analysis of the literature about the Random Access protocols with contention resolution, such as Contention Resolution Diversity Slotted Aloha (CRDSA), and introduces a possible enhancement, named Generalized Encoding Contention Resolution Diversity Slotted Aloha (GE-CRDSA). The GE-CRDSA aims at improving the aggregated throughput when the system load is less than 50%, playing on the opportunity of transmitting an optimal combination of information and parity packets frame by frame. This paper shows the improvement in terms of throughput, by performing traffic estimation and adaptive choice of information and parity rates, when a satellite network undergoes a variable traffic load profile.

Received on 13 February 2014; accepted on 24 September 2014; published on 28 December 2014

Keywords: Satellite, Interference Cancellation, DVB-RCS2, CRDSA, Slotted Aloha, Random Access

Copyright © 2014 Manlio Bacco *et al.*, licensed to ICST. This is an open access article distributed under the terms of the Creative Commons Attribution license (http://creativecommons.org/licenses/by/3.0/), which permits unlimited use, distribution and reproduction in any medium so long as the original work is properly cited.

doi:10.4108/mca.2.5.e2

1. Introduction

Satellite access networks provide a wide range of services for civil and military applications. These services include mobile data transfer, localization, satellite television and Internet web traffic. The Second Generation DVB Interactive Satellite Services (DVB-RCS2) [?] was recently introduced as a renewed standard for satellite communications and is specifically designed for Internet based services. Internet traffic is highly dominated by short-lived connections [?] and message exchanges, as far as mobile applications have gained the largest part of the Internet share. As shown in [?], the performance of access scheme are strongly linked to the traffic characteristics and in particular to the size of the file transferred. In addition, the advent of IoT (Internet of Things) services is boosting the bursty nature of the Internet traffic: such an evolution of the traffic nature requires adequate measurements on the multiple access protocols, in order to settle the suitability of either a Random Access (RA) or a Dedicated Access (DA). In this paper, a study on a generalized case of a Coded Slotted Aloha (CSA) is presented, showing that the random choice of information and redundancy lengths can improve the performance, in terms of throughput. In particular, runned simulations show that the design of a load control mechanism can help in obtaining a reasonable level of performances per load level, avoiding the complexity of a DA-like approach.

The rest of the paper is organized as follows: an highlight on the most relevant related works on Slotted Aloha (SA) based access protocols is presented in Section **??**. In Section **??**, the load estimation strategy is shown, as well as an ideal load control strategy based on Cross Entropy (CE) theory. The major contribution of this work is in Section **??**, a simple but efficient load control strategy that works without an a priori heuristic and guarantees a clear performance gain. Section **??** shows the results of the simulations, where the two control strategy are compared. Finally, the conclusions are in Section **??**.

2. Related Work

Many efforts have been made in the field of random access protocols for satellite communications, aiming at maximizing the aggregated throughput. Contention Resolution Diversity Slotted Aloha (CRDSA) [?] has been introduced in DVB-RCS2 standard [?] for data traffic with a dedicated system profile for SCADA (Supervisory Control And Data Acquisition) and M2M (Machine to Machine) communications. The advantage of contention resolution is represented by the increment of the performance in terms of throughput. In fact, CRDSA exhibits almost ideal performances (i.e low collision losses), up to 50-60% of the global access network load, in case of ideal power control [?]. Both

[*]corresponding author's e-mail: manlio.bacco@isti.cnr.it

CRDSA and SA assume the MAC frame duration equal to T_F, during which N_s slots are allocated. The timeslot has a duration of $T_S = T_F/N_s$. Assuming that in each frame u users try to access the shared channel, each user transmits one or more replicas of the same packet in the current MAC frame. In each packet a pointer to the positions of its replicas is included. The pointer is used to locate its own other replicas into the frame and to remove the interfering ones in the collided slots. The contention resolution proceeds iteratively, by removing the decoded signals in the relative collided slots. At the end of the procedure the Packet Loss (P_L) due to the un-cancelled collisions is significantly reduced. By definition, the throughput is given by:

$$T = G(1 - P_L), \qquad (1)$$

where $G = \frac{u}{N_s}$ is the number of packet transmission attempts per frame, i.e., the load of the system, when a single information packet is transmitted per user per frame. In SA, $(1 - P_L)$ is the probability that no others attempts of transmission arises during a timeslot. This value can be calculated through the Poisson distribution:

$$P(k) = G^k \frac{e^{-G}}{k!}$$

at $k = 0$. Let G^* be the supremum of G such that, in the asymptotic setting $u \to \infty$, the throughput T fulfills $T = G$; therefore G^* is the asymptotic peak throughput. In SA, the maximum throughput $T_{SA}^{max} = G^* e^{-G^*} = \frac{1}{e}$, is achieved for $G^* = 1$. When CRDSA operates at a given G, reducing P_L leads to a throughput gain with respect to SA and its non-SIC variants, as in DSA [?] with two or more replicas for each information packet transmission. This improvement is quantified in [?] as the normalized throughput $T_{CRDSA} \approx 0.55$, which is the probability of successful packet transmission per timeslot, whereas the peak throughput for SA is $T_{SA} \approx \frac{1}{e}$. Further improvements can be achieved by exploiting the capture effect [?] [?]. In [?] Irregular Repetition Slotted ALOHA (IRSA) was introduced to provide a further throughput gain over CRDSA. Higher throughput can be achieved by IRSA, allowing the Satellite Terminal (ST) to choose the number of replicas in a random way. As stated in [?], both CRDSA with one replica and CRDSA++ with more replicas can be considered as particular cases of IRSA where the repetition rate, frame by frame, is fixed. In [?], it is shown that a variable repetition rate does not introduce significant improvements in terms of throughput, if the set of slots per frame N_s is limited to about a hundred of slots, that is the range foreseen in DVB-RCS2 standard [?]. In [?], the throughput $T_{IRSA} \leq 0.8$, but N_s ranges up to 200 slots.

While CRDSA, CRDSA++ and IRSA are based on repetitions, Coded Slotted ALOHA (CSA) encodes an information burst - multiple of a timeslot - prior transmitting [?].

All the protocols so far discussed implicitly impose that the average load in the system is targeted around the G^* value. Otherwise, no improvement is appreciable. A centralised load control should estimate the current system load and dynamically reduce or increase the allocated set of slot N_s, by adapting the frame duration. Since N_s in DVB-RCS2 standard is bounded between 64 and 128 slots [?], the satellite system may perform at G loads under the desired G^*, leading to the under-utilization of the available bandwidth. In CSA, the transmitted burst is made up of $k = 2$ information packets out of n transmitted packets. The others $n - k$ packets are generated by a packet-oriented binary linear block code.

Differently, a generalized case of CSA is presented in this work, named Generalized Encoding CRDSA (GE-CRDSA), where each ST randomly sorts the couple (n, k). Note that $n = r + k$, where r is the number of parity packets. GE-CRDSA aims at optimizing CSA, by allowing each ST to transmit more than two information packets per frame, leading in most cases to reduce the queuing delay with respect to those systems, which consider only a single information packet per frame.

3. Problem Definition and System Overview

Let us consider a SA system, with a MAC frame duration T_F, composed of N_s slots. Let u_i be a Poisson distributed r.v. with mean G, which represents the number of active user terminals in frame i, being G the system load. In any frame, the ST randomly draws, over the set of C coding schemes, one couple (n, k), according to the Probability Mass Function (PMF) defined by the polynomial:

$$\Omega_{(n,k)}(x, y) \triangleq \sum_c \omega_c x^{k_c} y^{n_c}, \qquad (2)$$

where the weight ω_c is the probability of having k_c information packets over a n_c-long codeword, for the c-th coding scheme.

In each frame, the offered information rate (net traffic rate) is given by kG, while the gross traffic rate is given by nG. A flow control algorithm shall account for monitoring the average traffic load during an observation window and for communicating the weight vectors $\mathbf{\Omega} = \{\omega_c;\ c = 1 \cdots C\}$ to balance the traffic load, leading to the maximum system throughput. The optimal choice of the load vector $\mathbf{\Omega}$ was introduced in [?] and is presented in the following sections.

3.1. Load Estimation

Load Estimation (LE) techniques in [?] estimates the probability that a slot is in a given state by using the binomial distribution theory. In order to estimate the offered traffic load, a slot state can be in idle (I), successful (S) or collided (C) state, which represent zero, one or two (or more) bursts transmitted in a T_s, respectively. In SA, given the normalized offered load G and the frame length N_s, the expected numbers of slots per state per frame is defined in [?] as:

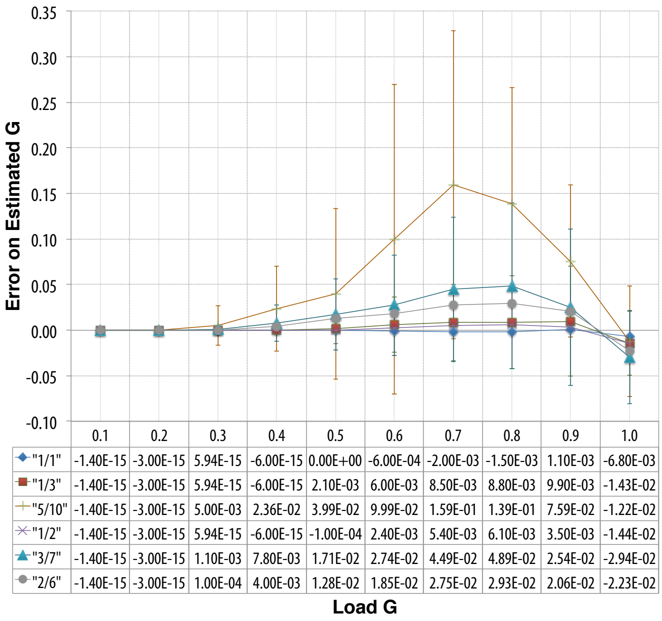
$$s_I(G, N_s) = N_s(1-p)^{GN_s}, \tag{3}$$

$$s_S(G, N_s) = GN_s(1-p)^{GN_s-1}, \tag{4}$$

$$s_C(G, N_s) = N_s - s_I - s_S, \tag{5}$$

where GN_s is the number of bursts per frame and $p = 1/N_s$ is the probability that the burst of a single node is in a given slot. The LE techniques asses the estimated offered load $\hat{G}$ by inversely tracking equations (**??**)-(**??**) from the observed vector $\mathbf{v} = \{i, s, c\}$, where i, s, and c are the number of slots per frame in the I, S, and C states, respectively. Let $\mathcal{H}$ be an LE function that returns $\hat{G}$, which are given in [?] by:

$$\mathcal{H}(\mathbf{v}) = \arg\max_{G} \left(G^s e^{-GN_s} (e^G - 1 - G)^c \right) \tag{6}$$



	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
◆ "1/1"	-1.40E-15	-3.00E-15	5.94E-15	-6.00E-15	0.00E+00	-6.00E-04	-2.00E-03	-1.50E-03	1.10E-03	-6.80E-03
■ "1/3"	-1.40E-15	-3.00E-15	5.94E-15	-6.00E-15	2.10E-03	6.00E-03	8.50E-03	8.80E-03	9.90E-03	-1.43E-02
+ "5/10"	-1.40E-15	-3.00E-15	5.00E-03	2.36E-02	3.99E-02	9.99E-02	1.59E-01	1.39E-01	7.59E-02	-1.22E-02
× "1/2"	-1.40E-15	-3.00E-15	5.94E-15	-6.00E-15	-1.00E-04	2.40E-03	5.40E-03	6.10E-03	3.50E-03	-1.44E-02
▲ "3/7"	-1.40E-15	-3.00E-15	1.10E-03	7.80E-03	1.71E-02	2.74E-02	4.49E-02	4.89E-02	2.54E-02	-2.94E-02
● "2/6"	-1.40E-15	-3.00E-15	1.00E-04	4.00E-03	1.28E-02	1.85E-02	2.75E-02	2.93E-02	2.06E-02	-2.23E-02

Figure 1. Error on load estimation (N_s=100) by using LE techniques combined with the use of linear codes in transmission. Codes in the table are in the form (k, n).

In order to apply Eq. (**??**) in CSA and therefore in GE-CRDSA, you must divide the return value of $\mathcal{H}$ by n, because the physical load is increased by the code length. Therefore, the process of dividing the load estimation by n leads to an estimation of the offered load. In order to obtain the normalized offered load G_{est}, the estimated offered load must be divided by the number of the timeslots N_s. Figure **??** shows the simulation results of Eq. (**??**) with different coding schemes, when $N_s = 100$. The error on load estimation is visible, together with the 0.05, the 0.50 and the 0.95 quantiles of the values collected during a Montecarlo test session. You can also read plot value in the table below the figure, where linear codes are presented as k/n in the first column. It is visible as, when using a linear code with $k > 1$, the estimation is subject to an increasing error.

3.2. Cross Entropy Control Strategy

In order to define a control law to extract the couple (n, k) at the G value that maximizes the aggregated throughput T, the controller needs either the a priori information, as the heuristic shown in Table **??**, or an analytical model as in [?], which provides the average throughput as a function of G, k and n. It is worth noting that a closed analytical form to derive the packet loss (or the throughput) value, given the parameters N_s, k, n and G is still absent, as far as the authors know.

Assuming that any station, in frame f, can randomly choose the couple (n, k) according to the PMF $\Omega_{(n,k)}$, the throughput maximization over all the possible couples (n, k) can be formally stated as:

$$\begin{aligned} &\max\{T\}_{\Omega_{(n,k)}} \\ &\qquad \text{s.t.} \\ &\qquad k \leq k_{max} \\ &\qquad n \leq n_{max} \\ &\qquad P_L \leq \theta \end{aligned} \tag{7}$$

G	(n, k)	T	P_L
0.1	(10, 5)	0.460	$6.598e^{-02}$
0.2	(7, 3)	0.527	$1.222e^{-01}$
0.3	(6, 2)	0.586	$3.288e^{-02}$
0.4	(4, 1)	0.400	$5.00e^{-05}$
0.5	(4, 1)	0.500	$1.23e^{-03}$
0.6	(3, 1)	0.600	$2.93e^{-02}$
0.7	(3, 1)	0.676	$2.952e^{-02}$
0.8	(3, 1)	0.536	$3.083e^{-01}$
0.9	(2, 1)	0.425	$5.319e^{-01}$
1.0	(2, 1)	0.373	$6.361e^{-01}$

Table 1. max normalized Throughput (T) vs. normalized Load (G) using best linear code (n, k), with packet loss P_L (N_s=100).

The solution of the optimization problem (**??**) is given by the PMF $\Omega_{(n,k)}$, used by the ST to draw the number of information and redundancy packets to be transmitted in the next frame f. The constraints of the optimization problem are the maximum allowed number of information packets k and the maximum number of packets n[1] per station per frame and, finally, the maximum value of P_L that can be tolerated. This problem can be addressed through the approaches either analytical or numerical [**?**]. Note that the first approach gives much more information about the behavior of the system.

In [**?**] an exhaustive evaluation of the possible couples (n, k) is presented, in order to achieve the maximum aggregated throughput, when N_s ranges from 100 to 1000 slots. For practical reasons, the main interest is for the N_s range foreseen in DVB-RCS2. Table **??** shows the optimal choice of the couple (n, k) that achieves the maximum aggregated throughput when N_s=100 and G ranges between 0.1 to 1.

Note that the load estimation may be defected by an estimation error, as shown in Figure **??**. In addition, the satellite feedback delay makes the estimation obsolete at least of the satellite latency (more than 250ms). As shown in [**?**] and [**?**], IRSA and CSA protocols can have enhancement in their performance, if a variable repetition rate is used. Note that, the G^* tracking problem is not trivial because the error estimation may lead the system to operate in the range $G > 0.7$, where it is deeply unstable and the collision rate is the prevailing factor in channel rate under-utilization.

Next it is shown how the problem discussed so far can be addressed by an iterative algorithm based on the Cross-Entropy theory. In the proposed formulation, it is assumed that C is the finite set of available coding schemes (n, k). Moreover, it is assumed that the c-th coding scheme holds the probability to be drawn ω_c.

The problem can be stated as follows: finding the PMF $\Omega_{(n,k)}$ of the coding schemes, which maximizes the objective function $\mathbf{O}_T((n, k))$ given by the aggregated average throughput T, assuming that the aggregated average packet loss probability P_L is within a given range.

The approach discussed in this section explores the space of solutions by a random method. Let γ be the maximum value of the objective function for the optimal couple $(n, k)^{opt}$. To solve the stated problem the PMF $\Omega^{opt}_{(n,k)}$ must be identified over the optimization domain. Hence our optimization problem can be viewed also as the maximization probability that our objective function is greater than γ over all the possible solutions for the PMF $\Omega_{(n,k)}$:

$$\mathbb{P}_{\Omega_{(n,k)}}\{\mathbf{O}_T((n,k)) \geq \gamma\} = \mathbb{E}_{\Omega_{(n,k)}}\left\{\mathcal{I}_{\mathbf{O}_T((n,k))} \geq \gamma\right\}. \qquad (8)$$

The weights ω_c able to maximize the probability of the event $\mathbf{O}_T((n, k)) \geq \gamma$ in Eq. (**??**) must be estimated.

This problem can be viewed as a Stochastic Node Network (SNN) problem with a set of nodes $\{1, \cdots, U\}$ and a set of node characteristics $\{1, \cdots, C\}$ distributed according to $\Omega_{(n,k)}$. The objective is to assign the node characteristics following $\Omega_{(n,k)}$ in a way that the objective function is maximized. Assuming that the nodes of the network sort the coding scheme independently and ω_c is the probability for the couple c to be drawn, the vector $\mathbf{\Omega} = \{\omega_1, \cdots, \omega_C\}$ represent the PMF from which any node draws its coding scheme.

This kind of SNN problems can be resolved through the Cross-Entropy (CE) method using an algorithm. The solution relies on the steps shown in Algorithm **??**.

Algorithm 1 Optimal PMF Algorithm

```
1. Generate the initial distribution $\mathbf{\Omega}_0 = [\hat{\omega}_1^{(0)}, \cdots, \hat{\omega}_C^{(0)}]$
with entries $\hat{\omega}_c^{(0)} = \frac{1}{C}$;
while $D_{KL}\left(\hat{\Omega}^{(t)}_{(n,k)}, \hat{\Omega}^{(t-1)}_{(n,k)}\right) \geq \epsilon$ do
    2. Generate $S$ random vectors $\{(n,k)^1_s \cdots (n,k)^u_s\}$; $s = 1 \cdots S$ of the $u$ coding schemes used by the $u$ users $\{(n,k)^1_s \cdots (n,k)^u_s\} \sim \hat{\Omega}^{(t-1)}_{(n,k)}$;
    3. Evaluate the (1-$\alpha$)-quantile of the sample coding schemes, for which the objective function is greater than $\hat{\gamma}^{(t-1)}$ ;
    4. Update the estimation $\hat{\Omega}^{(t-1)}_{(n,k)}$ to $\hat{\Omega}^{(t)}_{(n,k)}((n,k))$ through the samples that belong to the (1-$\alpha$)-quantile
    5. Evaluate the KL-Distance between the Mass Probability Functions at $(t)$ and $(t-1)$
    t=t+1
end while
```

The $(1-\alpha)$-quantile $\hat{\gamma}^{(t)}$ of the estimated PMF $\hat{\Omega}^{(t)}_{(n,k)}$ can be evaluated through Eq. (**??**).

$$\hat{\gamma} = \max\left\{o \mid \mathbb{P}_{\hat{\Omega}_{(n,k)}}(\mathbf{O}_T((n,k)) \geq o) \geq \alpha\right\}. \qquad (9)$$

The evaluation of the Kullback-Leibler distance (KL-Distance) between $\hat{\Omega}^{(t)}_{(n,k)}$ and $\hat{\Omega}^{(t-1)}_{(n,k)}$ at the step (5) of the Algorithm **??** can be evaluated as follows:

$$D_{KL}(\hat{\Omega}^{(t)}_{(n,k)}, \hat{\Omega}^{(t-1)}_{(n,k)}) = \sum_U \hat{\omega}_c^{(t-1)} \log\left(\frac{\hat{\omega}_c^{(t)}}{\omega\hat{}\,{}^{(\;t-1)}_{c}}\right). \qquad (10)$$

[1] Note that, in case of a single terminal, it would be possible that $k \leq n_{max} \leq N_s$, i.e., the station utilizes the entire frame.

The KL-Distance measures how much the new PMF is different from the previous one.

The update of the estimation $\hat{\Omega}^{(t-1)}_{(n,k)}$ at step (4) by the selected sample can be performed by the Eq. (**??**):

$$\hat{\omega}_c^{(t)} = \frac{\sum_{s=1}^{S} \mathcal{I}_{\{\mathbf{O}_T((n,k)_s) \geq \hat{\gamma}^{(t)}\}} \mathcal{I}_{\{\omega_c\}}}{\sum_{s=1}^{S} \mathcal{I}_{\{\mathbf{O}_T((n,k)_s) \geq \hat{\gamma}^{(t)}\}}}. \tag{11}$$

This equation shows that the weights of the new PMF are the occurrences of the coding scheme of the samples belonging to the $(1-\alpha)$-quantile.

To amplify the CE selection of the weights of the PMF, the entries of the estimated $\hat{\Omega}^{(t)}_{(n,k)}$ are modulated as in the Eq. (**??**).

$$\hat{\omega}_c^{(t)} = \begin{cases} \hat{\omega}_c^{(t-1)} \left(1 + \dfrac{T((n,k)_c)}{\sum_{c=1}^{C} T((n,k)_c)} \right) & , c \in S_{best} \\ (1 - \sum_{c=1}^{C} \hat{\omega}_c^{(t-1)}) \hat{\omega}_c^{(t-1)} & , c \notin S_{best}, c' \in S_{best} \end{cases} \tag{12}$$

In Eq. (**??**), S_{best} is the set of coding scheme samples belonging to the $(1-\alpha)$-quantile. The first row shows how the weights of the coding schemes belonging to the S_{best} are increased proportionally with the increment of throughput given to these coding schemes. Instead, the second row shows how the other weights are decreased proportionally, in a way that the new PMF does not change abruptly w.r.t. the previous one.

3.3. Load Control Strategy

In this section a different technique is presented, with respect to the one described in section **??**. The idea is to find a control algorithm able to perform, at least, as the one based on Cross-Entropy theory, while removing the need for an heuristic, i.e., for a priori known data.

This Load Control (LC) strategy is applied to the described system, in order to maximise the throughput T, given the estimated load G, computed as described in section **??**. Given that u users access to the channel at the same time using the same PMF, the aggregated mean value $(\overline{n}, \overline{k})$ and the code selection probability ω_c, used by the Satellite Terminals (STs), can be evaluated. In order to control and to bind the packet loss P_L perceived by the STs, an observer-like technique is exploited, using a window-based algorithm. The decoding process of the frame provides information to the observer about the coding scheme, the local throughput and the packet loss of the STs. By taking advantage of this information, the statistics computed by using the collected data can be used to control the performance of the system.

Table **??** shows the structure of the typical observation window. In the first column of the table, the time window duration $w_d = (t_e - t_s)$ is shown, where t_s and t_e are the timestamps representing the start and the end time of the window. The remaining columns show the transmitted information length k, the code length n, the estimated load $\hat{G}$, the number of collected samples S_n within the window for the given coding scheme (i.e., the number of STs that have transmitted during w_d using that linear code). The last two columns show the values of P_L and T for that specific linear code, in terms of mean value, given the number of samples S_n. Note that the single window can be viewed as a set of tuples of cardinality equal to the number of different linear codes that each ST can use during the transmission.

w_d (sec)	k	n	$\hat{G}$	S_n	P_L	T
3000	1	1	0.111	10	0.01	0.099
3000	1	3	0.125	100	0.0	0.1
3000	5	10	0.125	100	0.2	0.5

Table 2. Sample data from an observation window.

The proposed control algorithm periodically analyzes the last available window, with a period of LC_p seconds, in order to determine the aggregated mean value $(\overline{n}, \overline{k})$ and the corresponding P_L of the system. The controller will shift the $(\overline{n}, \overline{k})$ value, by changing the weights of the PMF $\Omega_{(n,k)}$, in order to improve the performances.

The main idea behind this control algorithm is to target a given P_L in the shortest possible time. P_L can be reduced by increasing the probability of drawing a code that fosters the P_L closer to the chosen packet loss value. For instance, referring to Table **??** and assuming the target $P_L = 0.18$, the aggregated packet loss can be forced to a value close to the P_L of the last row, increasing the probability to select the code $(n,k) = (n_c, k_c) = (10, 5)$, where (n_c, k_c) is the code (or the code set) that ensures the target will be reached as quickly as possible.

The controller periodically acts as described in Figure **??**: it receives as input the estimated load and the last available observation window. These data are used to estimate P_L, $(\overline{n}, \overline{k})$ and T.

Let $T^{(i)}$ and $P_L^{(i)}$ be the throughput and the packet loss, respectively, at time t_i. T_{max} represents the maximum throughput achievable for the current load level. The *trend* J is defined as:

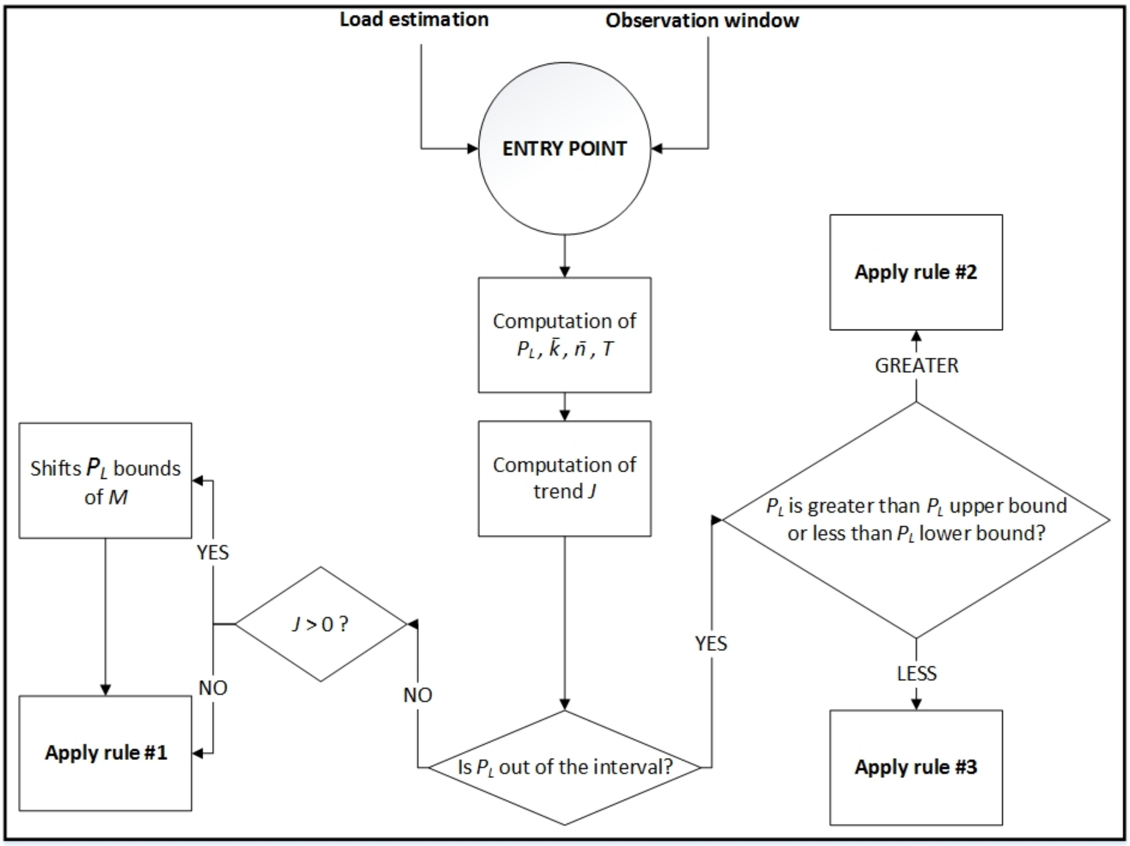

Load estimation
Observation window
ENTRY POINT
Computation of P_L, $\bar{k}$, $\bar{n}$, T
Computation of trend J
Apply rule #2
GREATER
P_L is greater than P_L upper bound or less than P_L lower bound?
LESS
Apply rule #3
YES
Is P_L out of the interval?
NO
$J>0$?
YES
NO
Shifts P_L bounds of M
Apply rule #1

Figure 2. Load Control Algorithm flowchart – the rules are detailed in Algorithm **??**

$$J = sgn((T_{max} - T^{(i)}))|\,(T_{max} - T^{(i)}) - (P_L^{(i)} - P_L^{(i-1)})| \tag{13}$$

This formulation ensures that J is strictly positive only if the T is increasing and its value depends on the difference between $(T^{(i)} - T^{(i-1)})$ and $(P_L^{(i)} - P_L^{(i-1)})$. The so-called *trend* is a metric used to evaluate the system behavior; if $J > 0$, the system is performing better than in the previous time window. The target P_L is upper and lower bounded by the values $P_{L(up)}$ and $P_{L(lo)}$, respectively, given that $P_{L(up)} - P_{L(lo)} > 0$.

The target Packet Loss is defined as follows:

$$P_{L(target)} = (P_{L(up)} - P_{L(lo)})/2. \tag{14}$$

The distance d between the current packet loss P_L and the target packet loss $P_{L(target)}$, defined as:

$$d = |\,P_{L(target)} - P_L| \tag{15}$$

is used in the calculation of the coefficient $g = 0.5d$. The control algorithm adds g to the weight of the selected linear codes and accordingly reduces the other weights. The increment grows linearly with d. Hence, the $P_{L(target)}$ can be quickly reached. The maximum value of g is 0.5, which means an increment of 50% for the extraction probability of the corresponding coding scheme for $d = 1$.

Referring to Figure **??**, the algorithm estimates P_L and T using the last available observation window. Then J is computed. Given the current load level, when $J > 0$ and $P_{L(up)} \leq P_L \leq P_{L(lo)}$, only one of the following conditions is verified in the previous time window: *(i)* the throughput gain is higher than the packet loss gain; *(ii)* the packet loss reduction is higher than the throughput reduction; *(iii)* the throughput has been increased and the packet loss has been reduced. This leads to shift up $P_{L(up)}$ and $P_{L(lo)}$. The bounds are redefined as: $P_{L(lo)} = P_{L(lo)} + M$ and $P_{L(up)} = P_{L(up)} + M$, where M is a tuning factor reported in Table **??**.

The selected codes (n_c, k_c) are the ones able to quickly target the desired value of packet loss: $P_{L(target)}$. The coefficient g, used to vary the PMF shape, is computed in a way that its value is greater if a quick change in the system behavior is desirable and, on the other hand, a little value lets us finely refine the PMF shape, while the current one is near the target. These choices guarantee that abruptly variations of the PMF are not possible and are supported through the empirical analysis of the system behavior, as well as from the results presented in the section **??**.

Algorithm 2 Load Control Algorithm

```
if {$P_{L(lo)} \leq P_L \leq P_{L(up)}$} then
    (RULE #1)
    $\omega_j = \omega_j + f;\ j = (n_c, k_c - 1)$
    $\omega_i = \omega_i + f;\ i = (\frac{\overline{n}}{\overline{k}}(k_c + 1), k_c + 1)$
    $(\overline{n}, \overline{k}) \approx (n_c, k_c)$
    subject to: $2k_c \leq n_c \leq 4k_c$
else if {$P_L \geq P_{L(up)}$} then
    (RULE #2)
    $\omega_j = \omega_j + f;\ j = (k_c, n_c)$
    $k_c = \min\{(\overline{k} - (\overline{k} - k_{min}) * d, k_{min})\}$
    if $k_c \neq k_{min}$ then
        $n_c = (\overline{n}/\overline{k}) * k_c$
    else
        $n_c = \max(n_c - 1, n_{min})$
    end if
    subject to:
    if $G_{est} \leq G_{peak}$ then
        $3k_c \leq n_c \leq 4k_c$
    else
        $2k_c \leq n_c \leq 4k_c$
    end if
else if {$P_L \leq P_{L(up)}$} then
    (RULE #3)
    $k_c = \min(k_c + 1, k_{max})$
    $n_c = (\overline{n}/\overline{k}) * k_c$
    subject to: $2k_c \leq n_c \leq 4k_c$
end if
```

The choice for the numerical values used with the Algorithm **??** is detailed in Table **??**.

Description	**Parameter**	**Value**
Min info size	k_{min}	1
Max info size	k_{max}	10
Min code length	n_{min}	1
Max code length	n_{max}	15
LC chosen k	k_c	$[k_{min}, k_{max}]$
LC chosen n	n_c	$[n_{min}, n_{max}]$
G peak threshold	G_{peak}	0.80
P_L tuning factor	M	0.005
Initial $P_{L(up)}$ value	$P_{L(up)}$	0.03
Initial $P_{L(lo)}$ value	$P_{L(lo)}$	0.01
Ω_i at t=0	Ω_i^0	$1/length(C)$
Control interval	LC_p	3s
Window duration	w_d	3s

Table 3. Simulation and tuning parameters for the LC algorithm

4. Numerical Results

The previous section describes the mathematical framework behind the GE-CRDSA. The algorithm allows the users to randomly choose the number of information and parity packets by using the PMF $\Omega_{(n,k)}$.

In this section, by exploiting the simulation results, it is shown how the PMF, and consequently the choice of (n_c, k_c), affects the throughput and the packet loss for different G values.

An ad-hoc simulator has been developed to test the two algorithms. The system and simulation parameters for the satellite scenario are summarized in Table **??**.

Parameter	**Value**
Bandwidth	8 Mhz
Modulation	QPSK
Phy FEC	1/3
Frame Duration	0.026s
N_s (slots)	100
IC max iterations[2]	20

Table 4. Simulation parameters for the case study

The simulation results are here presented. The LC technique and CE control strategy are compared, initially by forcing them to use the same set of linear codes C, detailed in table **??**. Figures **??** and **??** show the case of increasing G, while Figures **??** and **??** the decreasing one. A random G fluctuation can be obtained from those two profiles. In fact, a real G profile usually exhibits slower variations than those presented here. It is worth noting that the LC outperforms that based on CE theory in terms of throughput at G between 0.6 and 0.4, while the two control techniques show comparable behaviors at other loads.

In Figure **??**, throughput and packet loss are shown for a larger set of codes C, upper bounded by n_{max} and k_{max}, as in Table **??**. In fact, the LC technique is not limited by an a priori information request. Therefore whatever C is, it slowly adjusts the Ω in the described way. On the other hand, the CE Control Strategy (CECS) offers the possibility to immediately use the Ω showing the highest throughput at that G value, because of the heuristic. The LC and CE algorithms use a different codes distribution at each load level. If you refer to $(\overline{n}, \overline{k})$ plots, it is possible to see how the same throughput value can be reached using a different PMF. In Figure **??**, it is also possible to view $P_{L(up)}$ and $P_{L(low)}$ values and how the P_L value is bounded within, for G from 0.7

[2] The interference cancellation (IC) process performs several iterations in order to recover the maximum number of packets from the collision set in each frame. A DSA is equivalent to a CRDSA using a single iteration [**?**]. Trivially, SA is for $k = n = 1$.

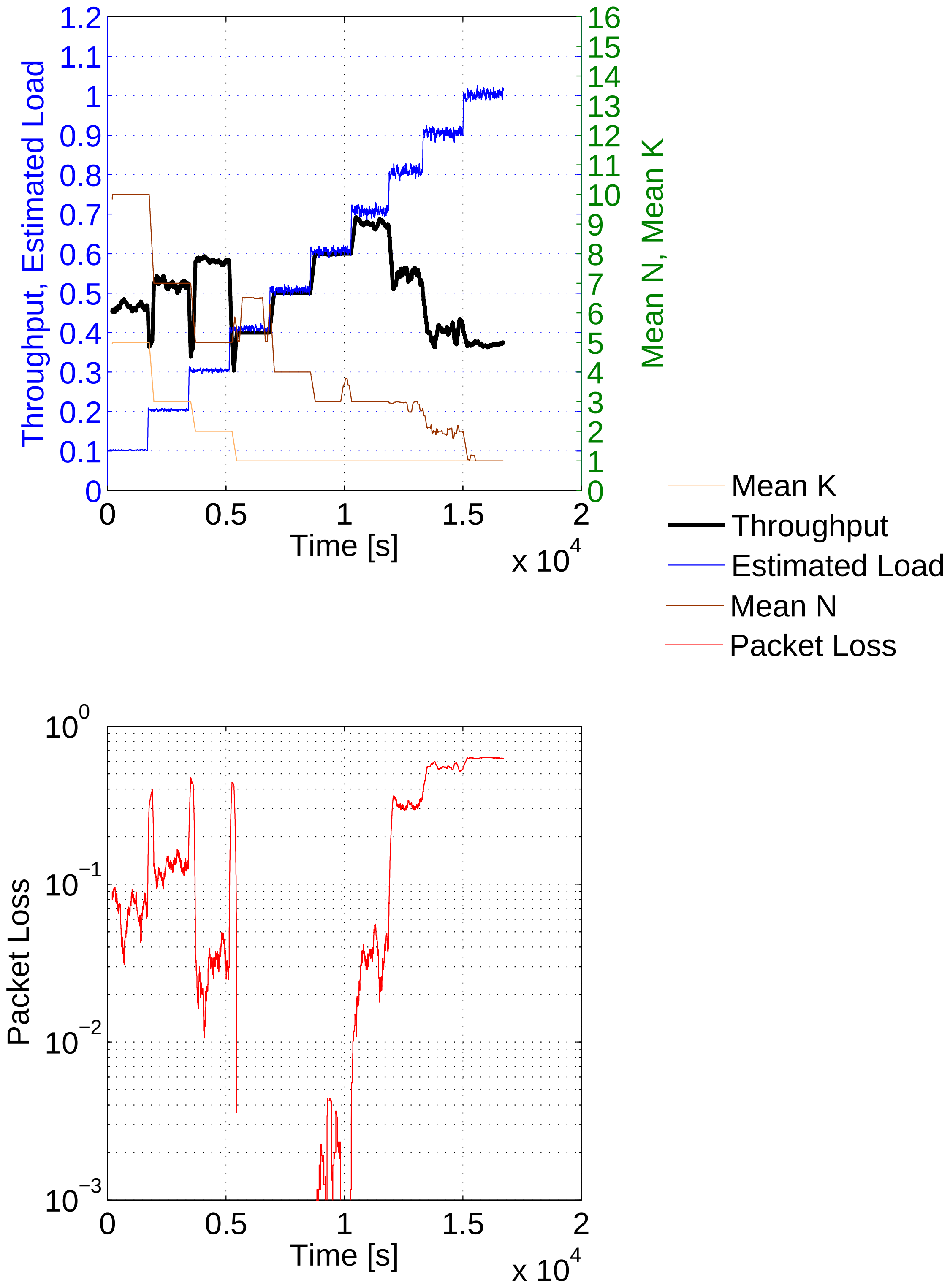


Figure 3. Throughput and Packet Loss for CE algorithm, linear code set C as in table **??**.

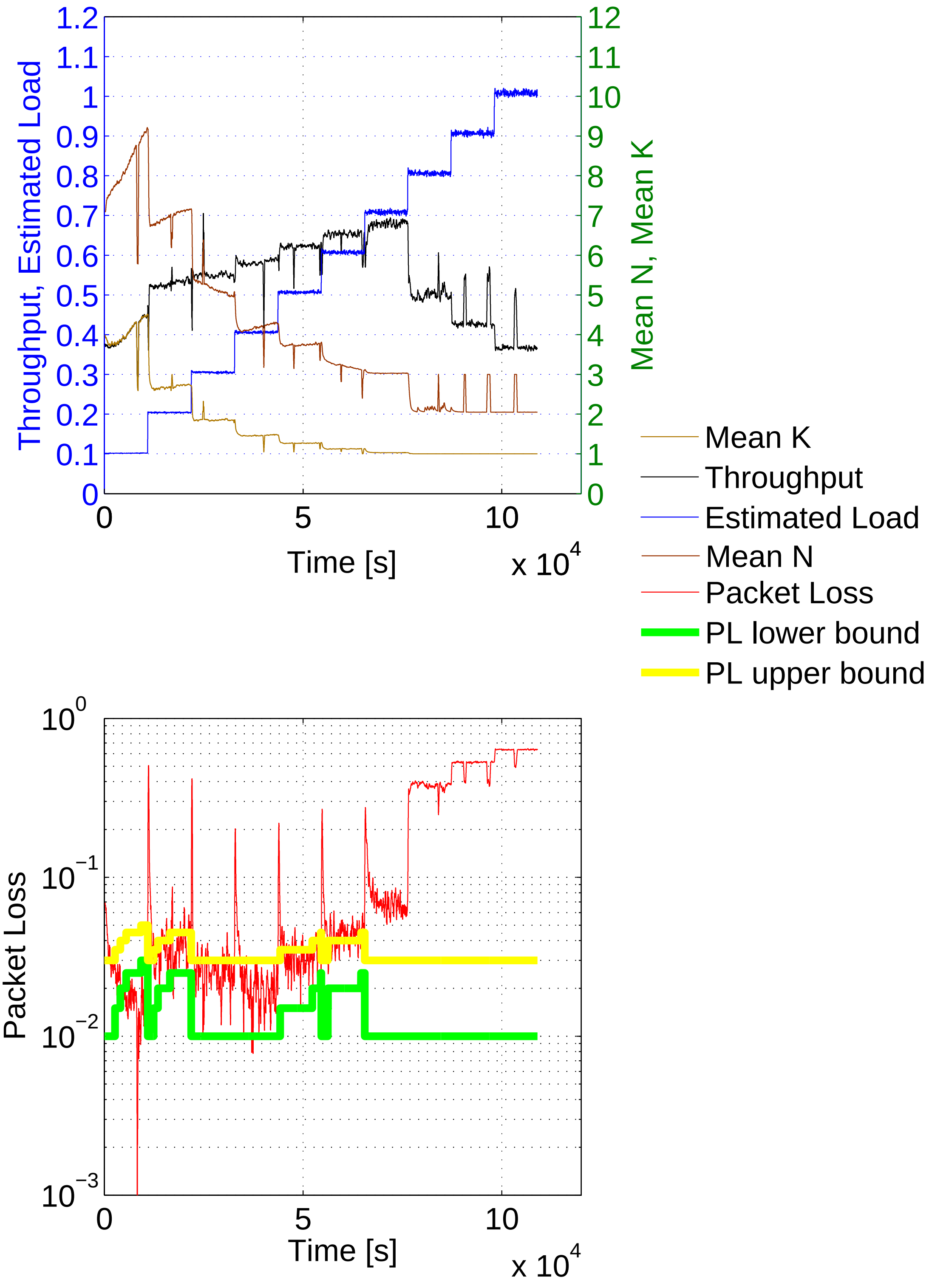


Figure 4. Throughput and Packet Loss for LC algorithm, linear code set C as in table **??**.

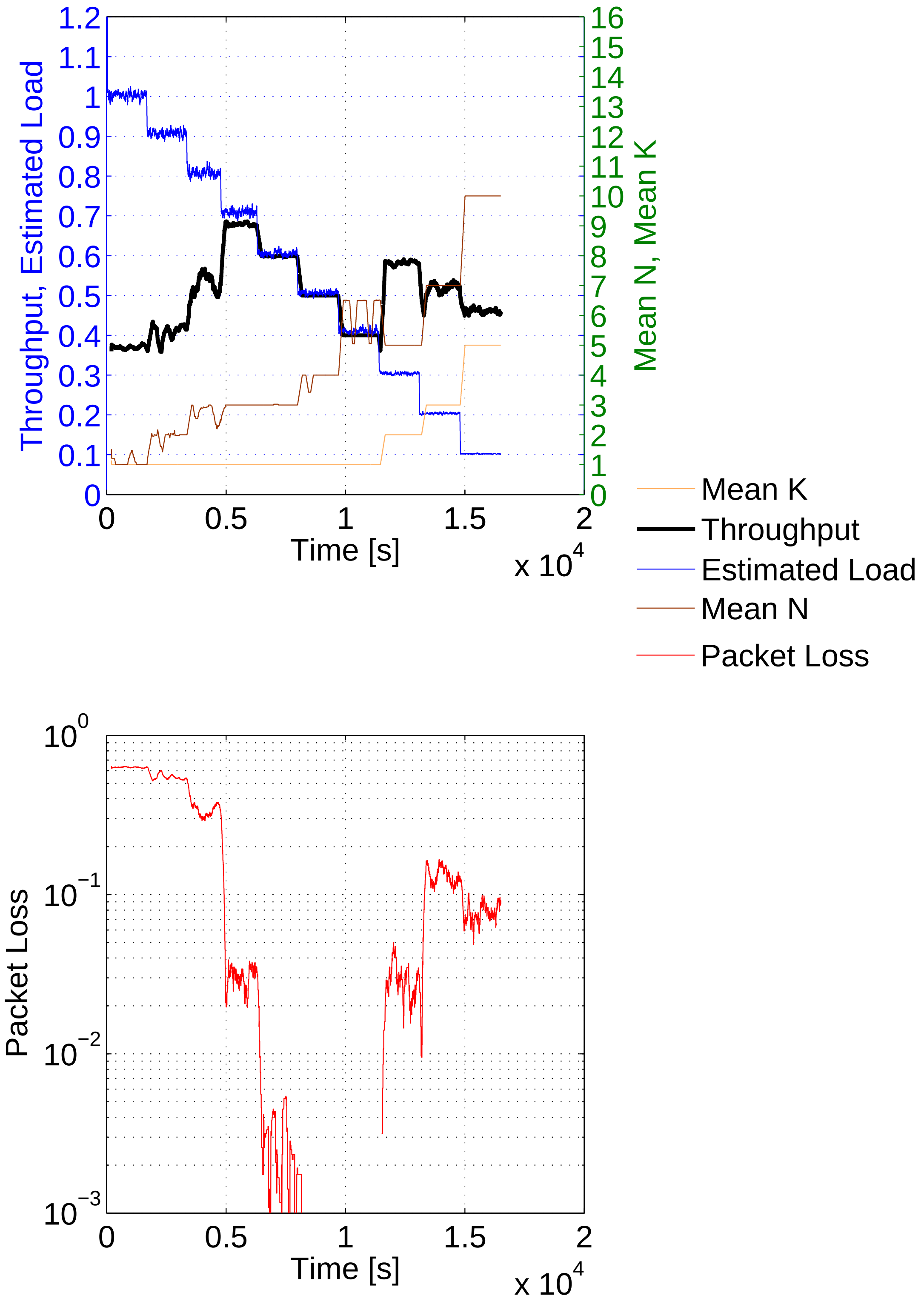


Figure 5. Throughput and Packet Loss for CE algorithm, linear code set C as in Table **??**.

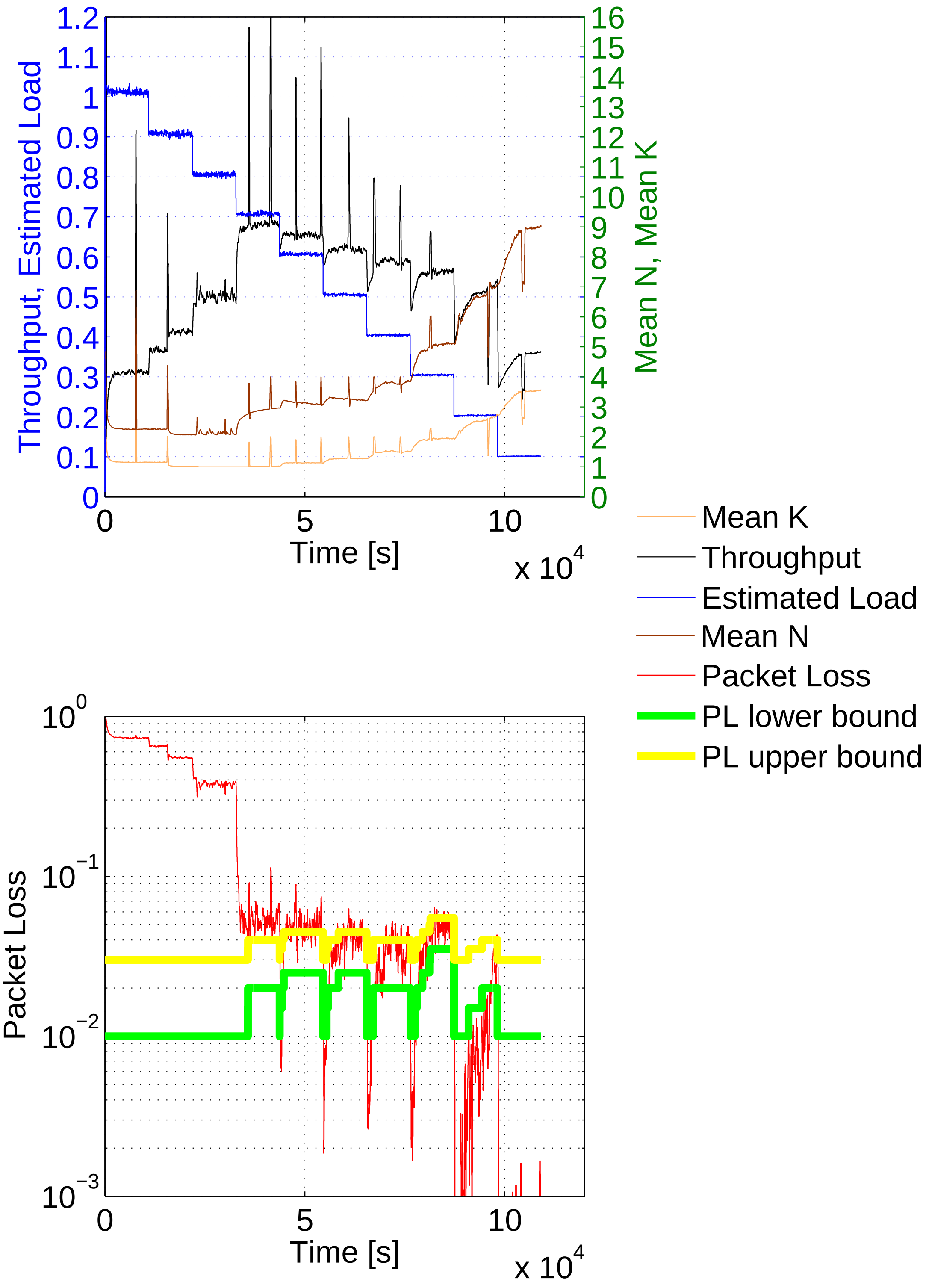


Figure 6. Throughput and Packet Loss for LC algorithm, linear code set *C* as in Table **??**.

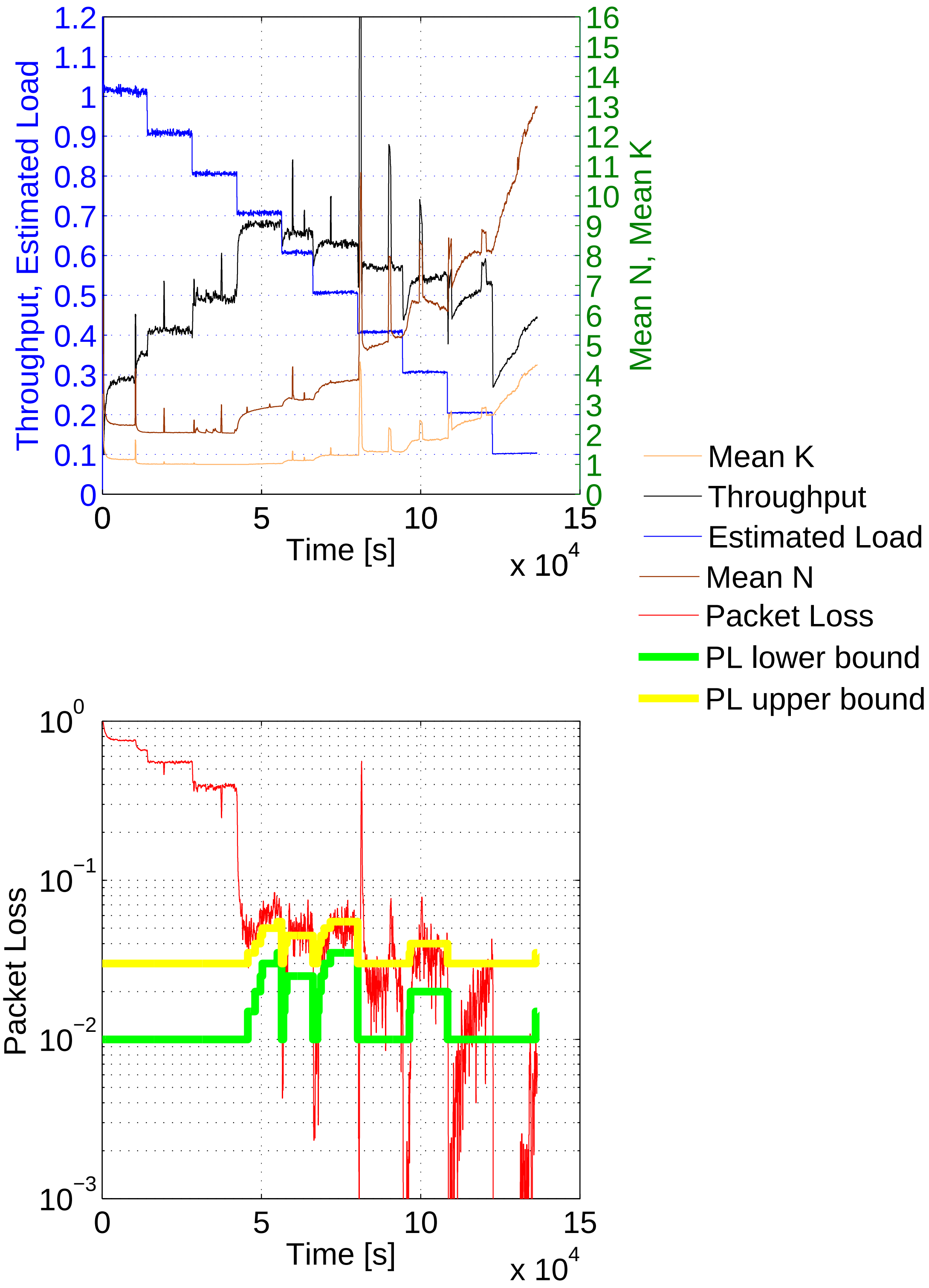


Figure 7. Throughput and Packet Loss for LC algorithm, linear code set *C* as in table **??**.

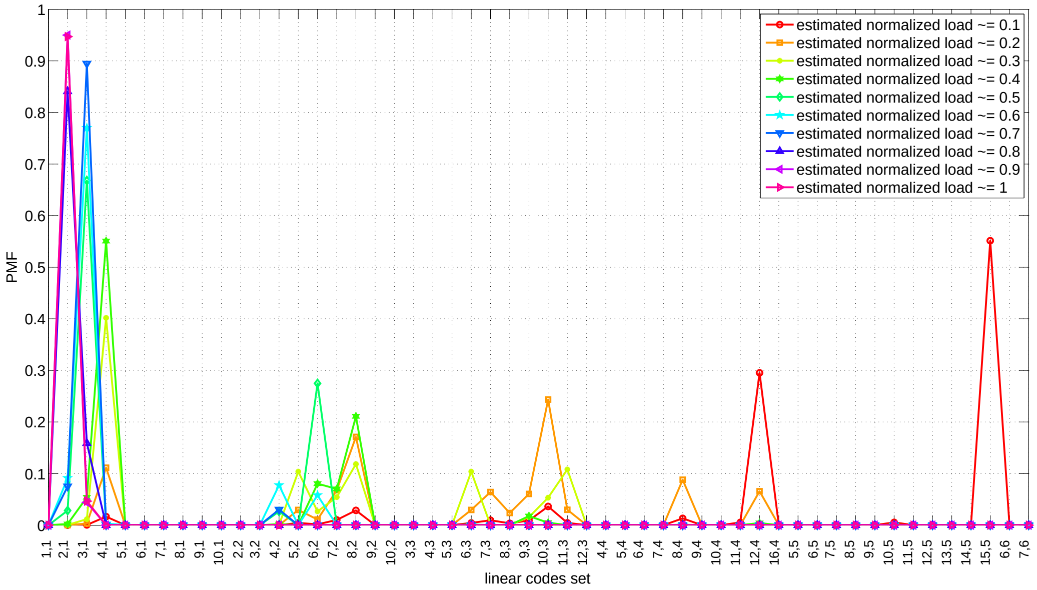


Figure 8. PMF for LC algorithm at different load levels for decreasing G, linear code set C as in table **??**.

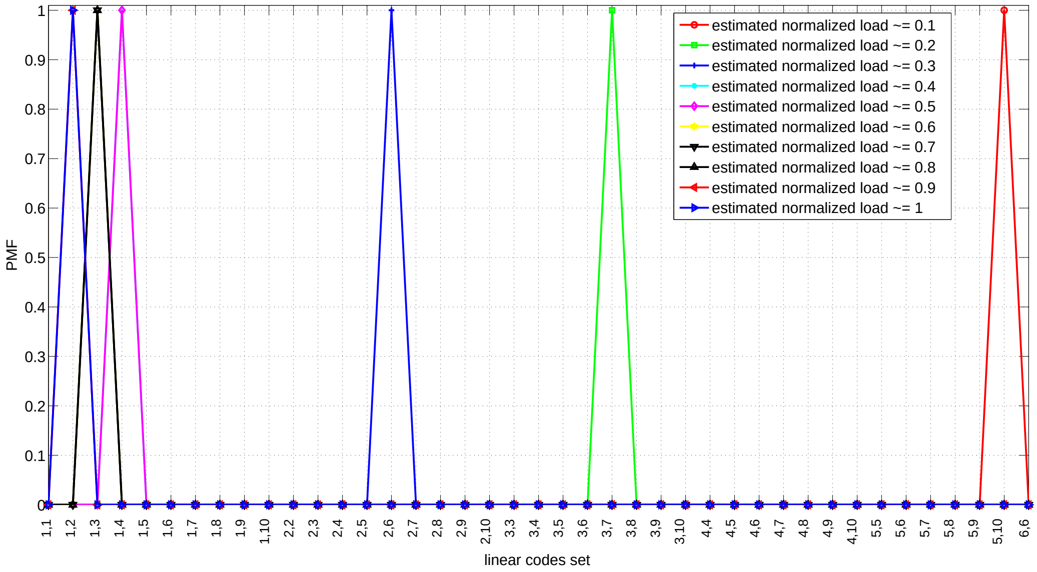


Figure 9. PMF for CE algorithm at different load levels for decreasing G, linear code set C as in table **??**.

to 0.1. The P_L bounds are shifted up in certain points, according to the algorithm described in section **??**.

Figure **??** plots the PMF of the codes that produce the highest throughput in Figure **??** (i.e., at the time instant immediately before a G variation), while Figure **??** shows the PMF of codes when the CECS is selected. Table **??** analytically describes the values of the PMF for the CECS.

Figures **??**, **??**, and **??** show the case of increasing G. In Figure **??** a throughput gain is appreciable at G from 0.4 to 0.6 with respect to Figure **??** paired with both the different choices for $(\overline{n}, \overline{k})$ and P_L with the relative

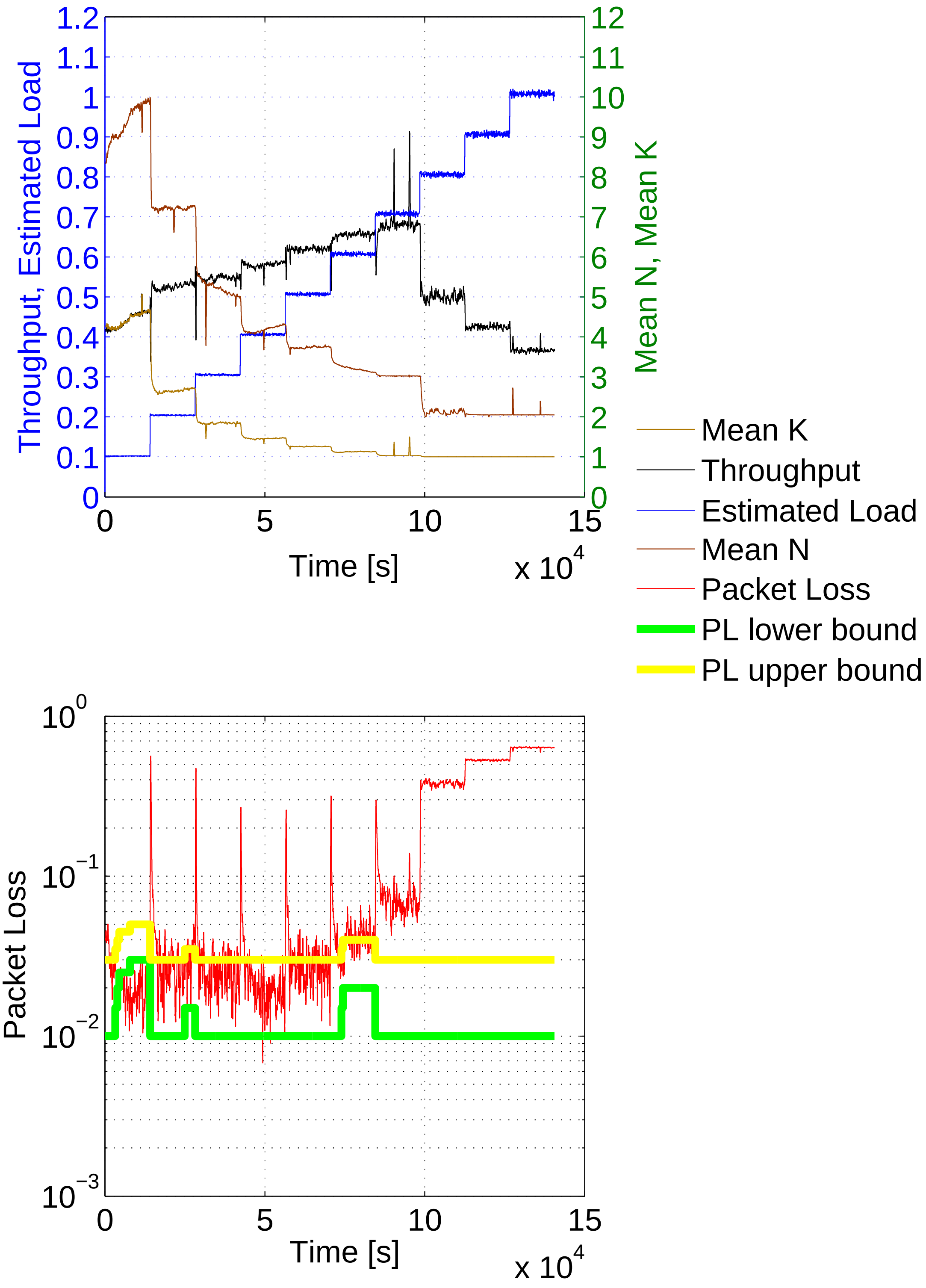


Figure 10. Throughput and Packet Loss for LC algorithm, linear code set C as in table **??**.

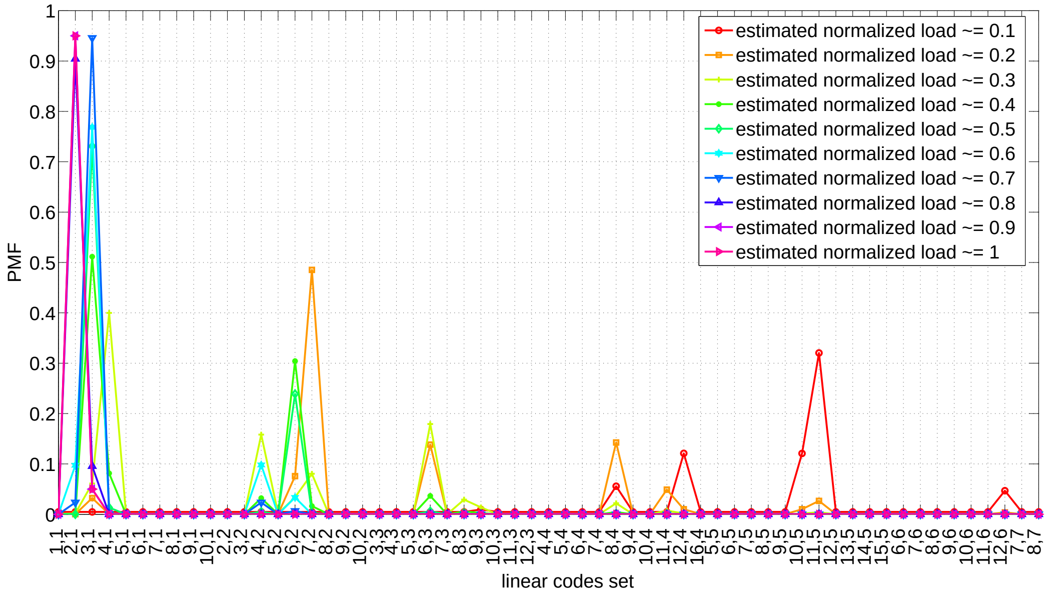


Figure 11. PMF for LC algorithm at different load levels for increasing G, linear code set C as in table **??**.

bounds. The PMF of linear codes of Figure **??** is shown in Figure **??**.

By comparing Figures **??** and **??**, it is noticeable how the load trend (i.e., increasing or decreasing) can lead to different PMF. This is explained by considering the state of the system at the time instant immediately before a load variation, because the control algorithm must quickly react to reduce the transient performance loss. This performance loss can be dependent from a too high packet loss (increasing load) or on a sub-utilization of the available bandwidth (decreasing load). Therefore, the decisions made by the LC algorithms lead to different shapes, whilst the throughput is about the same.

5. Conclusions

Contention resolution algorithms have demonstrated to successfully reduce the collision probability in random access, renewing the application of random access for information delivery. This work shows that by increasing the mean number of information packets sent by a station in each frame, when the system load is poorly loaded, the system throughput can be significantly improved up to the twice of that obtained with a standard CRDSA. In this work, we have shown how the design of a load control mechanism can help in obtaining a reasonable level of performances at each load level, avoiding the complexity of a DAMA-like (Demand Assigned Multiple Access) approach.

Since this study only accounts for colliding packets with the same $SNIR$ (Signal-to-Noise plus Interference Ratio), further improvements in terms of optimal system load G^* and maximum achievable aggregated throughput can be obtained, by considering power unbalancing and capture effect. However, achieving higher loads thanks to capture effect does not impact on the rationale behind the proposed scheme and further improvements could be shown in terms of aggregated throughput. GE-CRDSA does not neglect a load control system in order to track the optimal load correspondent to the maximum achievable throughput, but it relaxes the tracking constraints over a wider range of target loads, reducing the dynamic allocation of the collision set, i.e., the pool of slots dedicated to random access.

References

[1] Digital Video Broadcasting (DVB); Second Generation DVB Interactive Satellite System (RCS2); Part 2: Lower Layers for Satellite standard., Draft ETSI EN 301 545-2 V1.1.1 ed., June 2011.

[2] Labovitz C., Lekel-Johnson S., McPherson D., Oberheide J.,Jahanian F., Karir M., “Atlas internet observatory annual report", in *47th NANOG*, 2009.

[3] Roseti, Cesare, and Erling Kristiansen. "TCP behavior in a DVB-RCS environment." In 24th AIAA International Communications Satellite Systems Conference (ICSSC), San Diego (June 2006). 2006.

[4] Casini, Enrico, Riccardo De Gaudenzi, and Od R. Herrero. "Contention resolution diversity slotted ALOHA (CRDSA): An enhanced random access schemefor satellite access packet networks." Wireless Communications, IEEE Transactions on 6, no. 4 (2007): 1408-1419.

[5] Choudhury Gagan, L., and S. Rappaport Stephen. "Diversity ALOHA–A Random Access Scheme for Satellite Communications." Communications, IEEE Transactions volume 31, no. 3 (1983): 450-457.

[6] Roberts, Lawrence G. "ALOHA packet system with and without slots and capture." ACM SIGCOMM Computer Communication Review 5, no. 2 (1975): 28-42.

[7] Del Rio Herrero, O., Riccardo De Gaudenzi. "A high-performance MAC protocol for consumer broadband satellite systems." in Proc. of 27th AIAA Int. Communications Satellite Systems Conf. (ICSSC), Edinburgh, UK, June 2009.

[8] Liva, Gianluigi. "Graph-based analysis and optimization of contention resolution diversity slotted ALOHA." Communications, IEEE Transactions on 59, no. 2 (2011): 477-487.

[9] Celandroni, N., F. Davoli, E. Ferro, and A. Gotta. "Employing contention resolution random access schemes for elastic traffic on satellite channels." In 18th Ka and Broadband Communications Navigation and Earth Observation Conference, Ottawa, Canada, pp. 24-27. 2012.

[10] Paolini, Enrico, Gianluigi Liva, and Marco Chiani. "High throughput random access via codes on graphs: Coded slotted ALOHA." In Communications (ICC), 2011 IEEE International Conference on, pp. 1-6. IEEE, Kyoto, 2011.

[11] Bacco M., Cassará P., Ferro E., Gotta A., "Generalized Encoding CRDSA: Maximizing Throughput in Enhanced Random Access Schemes for Satellite" in Personal Satellite Services, LNICST vol. 123, pp. 115-122, Springer International Publishing, 2013.

[12] Zanella, Andrea. "Estimating collision set size in framed slotted aloha wireless networks and RFID systems." Communications Letters, IEEE 16, no. 3 (2012): 300-303.

[13] Boyd, Stephen, and Lieven Vandenberghe. Convex optimization. Cambridge university press, 2009.

[14] Botev, Zdravko I., and Dirk P. Kroese. "The generalized cross entropy method, with applications to probability density estimation." Methodology and Computing in Applied Probability 13, no. 1 (2011): 1-27.

[15] Matlab, www.mathworks.com/products/matlab/.