Dynamic Control of Coding for Progressive Packet Arrivals in DTNs

Eitan Altman, Lucile Sassatelli, and Francesco De Pellegrini

***Abstract*—In Delay Tolerant Networks (DTNs) the core challenge is to cope with lack of persistent connectivity and yet be able to deliver messages from source to destination. In particular, routing schemes that leverage relays' memory and mobility are a customary solution in order to improve message delivery delay. When large files need to be transferred from source to destination, not all packets may be available at the source prior to the first transmission. This motivates us to study general packet arrivals at the source, derive performance analysis of replication-based routing policies and study their optimization under two-hop routing. In particular, we determine the conditions for optimality in terms of probability of successful delivery and mean delay and we devise optimal policies, so-called *piecewise-threshold policies*. We account for linear block-codes and rateless random linear coding to efficiently generate redundancy, as well as for an energy constraint in the optimization. We numerically assess the higher efficiency of piecewise-threshold policies compared with other policies by developing heuristic optimization of the thresholds for all flavors of coding considered.**

***Index Terms*—Delay tolerant networks, mobile ad hoc networks, optimal scheduling, rateless codes, network coding.**

I. Introduction

DELAY Tolerant Networks (DTNs) leverage contacts between mobile nodes and sustain end-to-end communication even between nodes that do not have end-to-end connectivity at any given instant. In this context, contacts between DTN nodes may be rare, for instance due to low densities of active nodes, so that the design of routing strategies is a core step to permit timely delivery of information to a certain destination with high probability. When mobility is random, i.e., cannot be known beforehand, this is obtained at the cost of many replicas of the original information, a process which consumes energy and memory resources. Since many relay nodes (and thus network resources) may be involved in ensuring successful delivery, it becomes crucial to design efficient resource allocation and data storage protocols. The basic questions are then, sorted in the same order by which we tackle the problem:

Manuscript received February 9, 2012; revised July 5 and August 31, 2012; accepted October 28, 2012. The associate editor coordinating the review of this paper and approving it for publication was S. Liew.

E. Altman is with INRIA, Sophia Antipolis, France (e-mail: eitan.altman@inria.fr).

L. Sassatelli is with the I3S Laboratory at the Université Nice Sophia Antipolis and CNRS, UMR 7271, France (e-mail: sassatelli@i3s.unice.fr).

F. De Pellegrini is with Create-Net, Trento, Italy (e-mail: francesco.depellegrini@create-net.org).

A short version of this article was presented in the Proceedings of INFOCOM mini Conference, San Diego, March 2010 [1].

Digital Object Identifier 10.1109/TWC.2012.122212.120196

(i) **transmission policy:** when the source meets a relay node, should it transmit a packet?
(ii) **scheduling:** if yes, which packet should a source transfer?
(iii) **coding:** should the packets composing the message be encoded according to a specific scheme? If so, what is the resulting joint coding and scheduling?

In the basic scenario, the source has initially all the packets. Under this assumption it was shown in [2] that the transmission policy has a threshold structure: it is optimal to use all opportunities to spread packets till some time σ depending on the energy constraint, and then stop. This policy resembles the well-known "Spray-and-Wait" policy [3]. In this work we assume a more general arrival process of packets: they need not to be simultaneously available for transmission initially, i.e., when forwarding starts, as assumed in [2]. This is the case when large multimedia files are recorded at the source node (from, e.g., a cellular base station) that sends them out (in a DTN fashion) without waiting for the whole file reception.

Contributions. This paper focuses on general packet arrivals at the source and two-hop routing. We distinguish two cases: when the source can overwrite its own packets in the relay nodes, and when it cannot. The contributions are fourfold:

- For work-conserving policies (i.e., the source sends systematically before stopping completely), we derive the conditions for optimality in terms of probability of successful delivery and mean delay.
- In the case of non-overwriting, we prove that the best policies, in terms of delivery probability, are piecewise-threshold. For the overwriting case, work-conserving policies are the best without energy constraint, but are outperformed by piecewise-threshold policies when there is an energy constraint.
- We extend the above analysis to the case where copies are coded packets, generated both with linear block-codes and rateless coding. We also account for an energy constraint in the optimization.
- We illustrate numerically, in the non-overwriting case, the higher efficiency of piecewise-threshold policies compared with work-conserving policies by developing a heuristic optimization of the thresholds for all flavors of coding considered. As well, in the overwriting case, we show that work-conserving policies are the best without any energy constraint.

More in details, assume $\mathbf{t} = (t_1, \ldots, t_K)$ are the arrival times of the K packets at the source, $t_1 \leq t_2 \leq \ldots \leq t_K$. Owing to progressive arrivals, work-conserving policies (i.e., when the source sends a packet with probability 1 all the time) can be suboptimal, in contrast to [2], [4], in the non-overwriting case. When overwriting is allowed, WC policies may be suboptimal when an energy constraint is enforced. Defining couples (s_i, p_i) for $i = 1, \ldots, K$, where s_i are thresholds in $[t_i, t_{i+1}]$ and p_i are probabilities such that the source sends with probability p_i only between t_i and s_i, we are able to prove analytically that piecewise-threshold policies, defined by s_i' with $p_i = 1$, are required in the above cases.

Related Work. The original idea of packet level encoding for multicast was proposed first by [5] dealing with packet level forward error correction (FEC). Several satellites need to receive several data packets, that may need retransmission due to channel errors. Because many sites may need retransmissions, the problem is to avoid the phenomenon of *ACK implosion* due to several sites requesting repairs. [5] builds on Reed–Solomon codes, to transmit H additional packets, so that upon receiving any K of the $K+H$ packets, all stations are able to decode correctly the K information packets. This scheme, and this intuition, do prove useful also in our context. This seminal idea was employed later by several works to combine FEC and acknowledgment-based retransmission protocols, such as [6]. The effort there was to improve timeliness of packet delivery in multicasting multimedia streams which are subject to hard delay constraints. In DTNs the framework is different since the challenge is to overcome frequent disconnections. Papers [7] and [8] propose a technique to erasure code a file and distribute the generated code-blocks over a large number of relays in DTNs, so as to increase the efficiency of DTNs under uncertain mobility patterns. In [8] the performance gain of the coding scheme is compared with simple replication. The benefit of coding is assessed by extensive simulations and for different routing protocols, including two hop routing. In [7], the authors address the case of non-uniform encounter patterns, and they demonstrate strong dependence of the optimal successful delivery probability on the way replicas are distributed over different paths. The authors evaluate several allocation techniques; also, the problem is proved to be NP–hard. The paper [9] addresses the design of stateless routing protocols based on network coding, under intermittent end-to-end connectivity, and the advantage over plain probabilistic routing is proven. In [10] ODE-based models are employed under epidemic routing; in that work, semi-analytical numerical results are reported describing the effect of finite buffers and contact times. The same authors in [11] investigate the use of network coding using the Spray-and-Wait algorithm and analyze the performance in terms of the bandwidth of contacts, the energy constraint and the buffer size.

The structure of the paper is the following. In Sec. II we introduce the network model and the optimization problems tackled in the rest of the paper. Sec. III and Sec. IV describe optimal solutions in the case of work-conserving and not work-conserving forwarding policies, respectively. Sec. V addresses the case of energy constraints. In Sec. VI, the performance analysis of linear block-coding is presented. Rateless coding techniques are presented in Sec. VII. Sec. VIII provides a numerical analysis, and Sec. IX concludes the paper.

TABLE I
MAIN NOTATION USED THROUGHOUT THE PAPER

Symbol	*Meaning*
N	number of nodes (excluding the destination)
K	number of packets composing the file
H	number of redundant packets
λ	inter-meeting intensity
τ	timeout value
$X_i(t)$	fraction of nodes (excluding the destination) having packet i at time t
$X(t)$	summation $\sum_i X_i(t)$
$\widehat{X}_i, \widehat{X}$	corresponding sample paths
z	$:=X(0)$ will be taken 0 unless otherwise stated.
$u_i(t)$	forwarding policy for packet i; $\mathbf{u} = (u_1, u_2, \ldots, u_K)$
u	sum of the u_is
$Z_i(t)$,Z_i	$Z_i(t) = \int_0^T X_i(u)du$, $Z_i = Z_i(\tau)$, $\mathbf{Z}(t) = (Z_1(t), Z_2(t), \ldots)$, $\mathbf{Z} = \mathbf{Z}(\tau)$, $Z = \sum Z_i$
$D_i(\tau)$	probability of successful delivery of packet i by time τ
$P_s(\tau)$	probability of successful delivery of the file by time τ; $P_s(\tau, K, H)$ is used to stress the dependence on K and H
$\mathbb{R}_+$	nonnegative real numbers

II. THE MODEL

The main symbols used in the paper are reported in Tab. I. Consider a network that contains $N+1$ mobile nodes. We assume that two nodes are able to communicate when they come within reciprocal radio range, that communications are bidirectional and that the duration of such contacts is sufficient to one packet in each direction, and that the node buffer size is one packet. Also, let the time between contacts of pairs of nodes be exponentially distributed. The validity of this model has been discussed in [12], and its accuracy has been shown for a number of mobility models (Random Walk, Random Direction, Random Waypoint). Following the notation of [13], let β be the intra-meeting intensity, i.e., the mean number of meetings between any two given nodes per unit of time. Let λ be the inter-meeting intensity (mean number of meetings between one given node and any other nodes within a time unit). We have $\lambda = \beta N$. As we consider a sparse network, i.e., the density of nodes is constant in N, so is λ. That matches to the fact that the number of nodes that fall within radio range per unit of time remains constant if the density of nodes does. A file is transmitted from a source node to a destination node, and decomposed into K packets. The source of the file receives the packets at some times $t_1 \leq t_2 \leq \ldots \leq t_K$. The arrival time of the i-th packet is denoted by t_i. We assume that the transmitted file is relevant during some time τ, i.e., all the packets should arrive at the destination by time $t_1 + \tau$. Furthermore, we do not assume any feedback that allows the source or other mobiles to know whether the file has made it successfully to the destination within time τ.

We consider two-hop routing: a packet can go only through one relay. We distinguish two cases: when the source can overwrite its own packets in the relay nodes, and when it cannot. The possible reason for the source not to be allowed to overwrite its own packets would be to prevent source spoofing in case no authentication system is used between the nodes and an adversarial node would try to impede the transmission. The forwarding policy of the source is as follows. If at time t the source encounters a mobile, it gives it packet i with probability $u_i(t)$ if the overwriting case, and it does so in the non-overwriting case only if the met relay node does not have any packet. Clearly, $u(t) \leq 1$ where $u(t) = \sum_i u_i(t)$. Also, there is an obvious constraint that $u_i(t) = 0$ for $t \le t_i$.

Let $\widehat{\mathbf{X}}^{(N)}(t)$ be a K dimensional vector whose components are $\widehat{X}_i^{(N)}(t)$, for $i = 1, \ldots, K$. Here, $\widehat{X}_i^{(N)}(t)$ stands for the fraction of mobile nodes (excluding the destination) that have at time t a copy of packet i, in a network of size N. Let $\widehat{X}^{(N)}(t) = \sum_{i=1}^K \widehat{X}_i^{(N)}(t)$. This models the spreading process of the i-th packet in the network: in the following we will refer to fluid approximations that describe the dynamics of the fraction of infected nodes. The validity of such approximation is detailed below. In particular, owing to the dependence of this process on β, and hence on N, we cannot readily apply the convergence results of [14]. Below we detail the approach based on the work of Benaïm and Le Boudec in [15], so as to derive analytical expressions of some performance measures.

A. Performance measures: mean-field approximations

Consider time t sampled over the discrete domain, i.e., $t \in \mathbb{N}$. For our case, the drift defined in [15] is

$$f^{(N,i)}(\mathbf{m}) = \mathbb{E}\left(\widehat{X}_i^{(N)}(t+1) - \widehat{X}_i^{(N)}(t) | \widehat{\mathbf{X}}^{(N)}(t) = \mathbf{m}\right) .$$

Owing to the model, we have $f^{(N,i)}(\mathbf{m}) = u_i(t)\beta(1 - \sum_{k=1}^K m_i)$ in the non-overwriting case, and $f^{(N,i)}(\mathbf{m}) = \beta u_i(t)(1-m_i) - \beta m_i(u(t) - u_i(t))$ in the overwriting case. By checking condition **H2** of [15], there exists a function $\epsilon(N)$ such that $\lim_{N\to\infty} \epsilon(N) = 0$ and, for all $m \in [0,1]$, $\lim_{N\to\infty} \frac{f^{(N,i)}(\mathbf{m})}{\epsilon(N)} = f_i(\mathbf{m})$. Indeed, $\epsilon(N) = \beta(N)/\lambda$, fulfills the condition, since $\lim_{N\to\infty} \beta(N) = 0$ and λ is a constant in N. This interaction process is then said to have vanishing intensity [15]. In such cases, it has been shown in [15] that, provided that we change the time scale, the re-scaled process $\tilde{X}_i^{(N)}(r)$ converges for large N, in mean-square, to a deterministic dynamic system which is the solution of a certain Ordinary Differential Equation (ODE). More precisely, $\tilde{X}_i^{(N)}(r)$ is defined as a continuous time process by

$$\begin{cases} \tilde{X}_i^{(N)}(t\epsilon(N)) = \widehat{X}_i^{(N)}(t) & \text{for all } t \in \mathbb{N} \\ \tilde{X}_i^{(N)}(r) \quad \text{is affine on} & r \in [t\epsilon(N); (t+1)\epsilon(N)] \end{cases}$$

Let us define $\tilde{u}_i(r)$ as the re-scaled version of $u_i(t)$, in the same way as for $\tilde{X}_i^{(N)}(r)$. Then $\tilde{X}_i^{(N)}(r)$ converges to a deterministic process $\bar{X}_i^{(N)}(r)$ which is the solution of $\frac{d\mu_i(r)}{dr} = f_i(\mu(r)), \quad \bar{X}_i^{(N)}(0) = z$. Let us denote $\epsilon(N)$ by ϵ in what follows for lighter notation. We can also express the limit $\bar{P}_i(r)$ of the cumulative distribution function (CDF) of the re-scaled delay for packet i to make it successfully to its final destination: $\frac{d\bar{P}_i(r)}{dr} = \frac{\beta}{\epsilon}\bar{X}_i^{(N)}(r)(1-\bar{P}_i(r))$, whereby $\bar{P}_i(r) = 1 - \exp\left(-\frac{\beta}{\epsilon}\int_0^r \bar{X}_i^{(N)}(s)\,ds\right)$. As mentioned in [15] (page 15), the above derivation can be used to approximate the random variable $\widehat{X}_i^{(N)}(t)$ by $X_i^{(N)}(t) = \bar{X}_i^{(N)}(\epsilon t)$ for large values of N. Note that the limit of $\widehat{X}_i^{(N)}(t)$ is zero as $\lim_{N\to\infty} \beta(N) = 0$. As well, we can approximate the CDF of the delivery delay of packet i, denoted by $P_i(t)$, by $\bar{P}_i(\epsilon t)$. Then we readily get the approximation of the probability of reception of the K packets by τ: $P_s(\tau) = \prod_{i=1}^K P_i(\tau)$. with $P_i(t) = \bar{P}_i(\epsilon t) = 1 - \exp\left(-\lambda \int_0^t X_i^{(N)}(s)\,ds\right)$. In the sequel of the paper, we will use the so-defined notation $Z_i(\tau) = \int_0^\tau X_i^{(N)}(s)\,ds$. We can also get an approximation of the mean completion time for large values of N: $\mathbb{E}[D] = \int_0^\infty 1 - P_s(t)\,dt$.

B. Problem statement

We shall now introduce two classes of forwarding policies.

Definition 2.1: We define u to be a *work-conserving* (WC) policy if whenever the source meets a node then it forwards it a packet, unless the energy constraint has already been attained.

Definition 2.2: We define u to be a *piecewise-threshold* policy if the source systematically transmits up to threshold time s_i after receiving packet i, and then stops forwarding until the next packet arrives.

We shall study the following optimization problems:

- **P1.** Find $\mathbf{u}$ that maximizes the probability of successful delivery till time τ (over all kinds of policies).
- **P2.** Find $\mathbf{u}$ that minimizes the expected delivery time over the WC policies.

Policy $\mathbf{u}$ is called *uniformly optimal* for problem P1 if it is optimal for problem P1 for all $\tau > 0$.

Remark 2.1: Note that the forwarding policies we will consider are deployable as the source fully controls the dissemination in a two-hop scheme, provided that information of network parameters such as N and λ is available at the source node. As shown in [16], direct estimation of N and λ can even be avoided provided that stochastic approximation algorithms are employed, but, such techniques are beyond the scope of the present paper.

Energy Constraints: We can denote by $\mathcal{E}(t)$ the energy consumed by the whole network for transmitting and receiving the file during the time interval $[0,t]$. We adopt a linear model, where $\mathcal{E}(t)$ is proportional to the number of transmissions that is $X(t) - X(0)$ because packets are transmitted only to mobiles that do not have any. In particular, let $\varepsilon > 0$ be the energy spent to forward a packet during a contact, including the energy spent to receive the file at the receiver side. We thus have $\mathcal{E}(t) = \varepsilon(X(t) - X(0))$. In the following we will denote x as the maximum number of packets that can be released due to energy constraints. Accordingly, we introduce constrained problems **CP1** and **CP2** obtained from problems P1 and P2, respectively, by restricting to policies for which the energy consumption till time τ is bounded by X_e.

III. Optimal Scheduling

A. An optimal equalizing solution

Theorem 3.1: Fix $\tau > 0$. Assume that there exists some policy $\mathbf{u}(t)$ satisfying $\sum_{i=1}^K u_i(t) = 1$ for all $t \le \tau$ and $\int_0^\tau X_i(t)dt$ is the same for all i's. Then $\mathbf{u}(t)$ is optimal for P1.

Proof. Define the function ζ over the real numbers: $\zeta(h) = 1 - \exp(-\lambda h)$. Denote $\mathbf{Z} = (Z_1, \ldots, Z_K)$ such that $Z_i = \int_0^\tau X_i(v)dv$, and let Z_{total} be $Z_{total} = \sum_{i=1}^K Z_i$ (for lighter notation, we drop the dependence of $\mathbf{Z}$ in τ). We note that

$\zeta(h)$ is concave in h and that $\log P_s(\tau, \mathbf{u}) = \sum_{i=1}^K \log(P_i) = \sum_{i=1}^K \log(\zeta(Z_i))$. It then follows from Jensen's inequality that, for a given fixed Z_{total}, the success probability when using $\mathbf{u}$ satisfies $\log P_s(\tau, \mathbf{u}) \leq K \log(\zeta(Z_{total}/K))$ with equality if Z_i are the same for all i's. Moreover, $\log(\zeta(Z_{total}/K))$ is increasing with Z_{total}, and a WC policy achieves the highest feasible Z_{total} by construction. This implies the theorem. $\diamond$

Note: This theorem can be also proven by proof of Theorem 6.1 by taking $H = 0$.

In the non-overwriting case, it is worth noting that the reciprocal of the above theorem is not true: amongst all feasible policies, the policy $\mathbf{u}(\mathbf{t})$ giving the highest delivery probability is not necessarily WC or equalizing the Z_i. It is easy to exhibit counter-examples. Explanation is as follows. The limiting parameters for Z_i are (i) the arrival time of packet i and (ii) the number of already occupied nodes when packet i starts to spread.

- The optimal policy for P1 may not be WC:
  Let $Z_{total} = \sum_{i=1}^K Z_i$. The maximum of Z_{total} is obtained for a WC policy. For fixed Z_{total}, the maximum of $P_s(\tau)$ is indeed obtained by an equalizing policy, but such policy may not belong to the set of feasible policies. In such case, the maximum of $P_s(\tau)$ may be reached by a non-WC policy (non-maximum Z_{total}). For example in the case the last packet arrives much later than the other packets and if $t_K - \tau$ is high enough, then the network must not be saturated so as to allow the last packet to spread enough until τ.
- The optimal policy for P1 may not equalize the Z_i:
  Consider success probability $P_s(\tau) = \prod_{i=1}^K (1 - \exp(-\lambda Z_i))$ and maximization of $P_s(\tau)$ over the policies equalizing the Z_i. Such a policy maximizes the minimum of all Z_i. Hence, constraining the Z_i to be equal will set them to the $\min_i Z_i$, since Z_i is anyway constrained by the arrival time of packet i (e.g., case of $i = K$ arrived only a few instants before τ). That is why constraining to Z_i the Z_j of other packets arrived earlier than packet i is not optimal: it may be better to decrease a bit Z_i while increasing a lot Z_j so that $P_s(\tau)$ gets increased. For example in the case the last packet arrives much later than the other packets and if $t_K - \tau$ is low, then the spreading of packet K will be limited by $t_K - \tau$ and it is not efficient to limit too much the spreading of the other packets (i.e., lowering the Z_i for $i < K$) to let more room than needed by packet K.

B. *Constructing an optimal WC policy*

We propose an algorithm that has the property to generate a policy $\mathbf{u}$ which is optimal not just for the given horizon τ but also for any horizon shorter than τ. Yet optimality here is only claimed with respect to WC policies. We need some auxiliary definitions that we list in order:

- $Z_j(t) := \int_{t_1}^t X_j(r)dr$. We call $Z_j(t)$ the cumulative contact intensity (CCI) of class j.
- $I(t, A) := \min_{j \in A}(Z_j, Z_j > 0)$. This is the minimum non zero CCI over j in a set A at time t.
- Let $J(t, A)$ be the subset of elements of A that achieve the minimum $I(t, A)$.
- Let $S(i, A) := \sup(t : i \notin J(t, A))$ for i in A.
- Define e_i to be the policy that sends packets of type i with probability 1 at time t and does not send packets of other types.

TABLE II
ALGORITHM A

A1	Use $\mathbf{p}_t = e_1$ at time $t \in [t_1, t_2)$.
A2	Use $\mathbf{p}_t = e_2$ from time t_2 till $s(1,2) = \min(S(2, \{1,2\}), t_3)$. If $s(1,2) < t_3$ then switch to $\mathbf{p}_t = \frac{1}{2}(e_1 + e_2)$ till time t_3.
A3	Define $t_{K+1} = \tau$. Repeat the following for $i = 3, ..., K$:
A3.1	Set $j = i$. Set $s(i,j) = t_i$
A3.2	Use $\mathbf{p}_t = \frac{1}{i+1-j} \sum_{k=j}^i e_k$ from time $s(i,j)$ till $s(i, j-1) := \min(S(j, \{1, 2, ..., i\}), t_{i+1})$. If $j = 1$ then end.
A3.3	If $s(i, j-1) < t_{i+1}$ then take $j = \min(j : j \in J(t, \{1, ..., i\}))$ and go to step [A3.2].

Algorithm A in Table II strives for equalizing the less populated packets at each point in time: it first increases the CCI of the latest arrived packet, trying to increase it to the minimum CCI which was attained over all the packets existing before the last one arrived (step A3.2). If the minimum is reached (at some threshold s), then it increases the fraction of all packets currently having minimum CCI, seeking now to equalize towards the second smallest CCI, sharing equally the forwarding probability among all such packets. The process is repeated until the packet arrives: hence, the same procedure is applied over the novel interval. Moreover, it holds the following:

Theorem 3.2: Let $\mathbf{u}^*(t)$ be the policy obtained by Algorithm A when substituting there $\tau = \infty$. Then
(i) $\mathbf{u}^*(t)$ is optimal for P2.
(ii) $\mathbf{u}^*(t)$ is also optimal for P1 in the overwriting case.
(iii) Consider some finite τ. If in addition $\int_0^\tau X_i(t)dt$ are the same for all i's, then $\mathbf{u}^*(t)$ is optimal for P1.

Proof: The proof for (i) is given in the appendix. (ii) Z_{total} is maximized with a WC policy, and the Z_i's can be equalized as much as the t_i's allow because the source can always overwrite packets. (iii) The proof is the same as for Theorem 3.1. $\diamond$

IV. BEYOND WC POLICIES

We have obtained the structure of the best WC policies, and identified cases in which these are globally optimal. We show the limitation of WC policies.

A. *The case K=2*

1) The non-overwriting case: Consider two packets, arriving at the source at t_1 and t_2, respectively. Consider the policy $\mu(s)$ where $0 = t_1 < s \leq t_2$ which transmits packet 1 during $[t_1, s)$, does not transmit anything during $[s, t_2)$ and then transmits packet 2 after t_2. Let us define $X(t) = 1 - \exp(-\beta t)$.

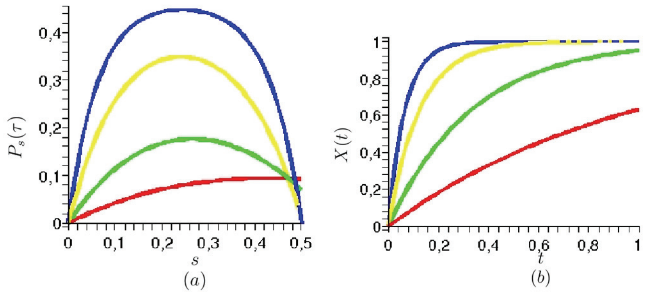

Fig. 1. $K=2$, $\beta = 1,3,8,15$, $t_1 = 0$, $t_2 = 0.8$. (a) Success probability under non WC policy $\mathbf{u}(s)$ as a function of s; top curve corresponds to largest value of β; second top corresponds to second largest β etc. (this order changes only at s very close to 0.5). (b) The evolution of $X(t)$ as a function of t under the best WC policy. The curves are ordered according to β with the top curve corresponding to the largest β etc.

Then it holds

$$X_1(t) = \begin{cases} X(t) & 0 \le t \le s \\ X(s) & s \le t \le \tau \end{cases}$$

$$X_2(t) = \begin{cases} 0 & 0 \le t \le t_2 \\ X(t-(t_2-s)) - X(s) = & \\ e^{-\beta s} - e^{-\beta(t-(t_2-s))} & t_2 \le t \le \tau \end{cases}$$

This gives

$$\int_0^\tau X_1(t)dt = \frac{-1+\beta s + e^{-\beta s}}{\beta} + (\tau - s)(1-e^{-\beta s})$$

$$\int_0^\tau X_2(t)dt = \frac{e^{-\beta s}}{\beta}(\beta(\tau - t_2) - 1 + e^{-\beta(\tau - t_2)})$$

Example 4.1: Using the above dynamics, we can illustrate the improvement that non WC policies can bring. We took $\tau = 1$, $t_1 = 0$, $t_2 = 0.8$. We vary s between 0 and t_2 and compute the probability of successful delivery for $\beta = 1,3,8,15$. The probability of successful delivery under the piecewise-threshold policies $u(s)$ are depicted in Figure 1 as a function of s which is varied between 0 and t_2. The corresponding optimal policies $u(s)$ are given by the thresholds $s = 0.425, 0.265, 0.242, 0.242$ for the β above, respectively.

In all these examples, there is no optimal policy among those that are WC (i.e., with $s = t_2$). We can verify that a WC policy is optimal for all $\beta \le 0.9925$. Note that under any WC policy, as β increases to infinity, $X_1(t_2)$ and hence $X(t_2)$ increase to one. Thus $\int_0^\tau X_2(t)$ tends to zero. We conclude that the success delivery probability tends to zero, uniformly under any WC policy.

2) The overwriting case: WC policies are optimal for P2 and P1 provided that no energy constraint is enforced. WC policies are not necessarily optimal anymore with an energy constraint (e.g., if X_e is lower than $X_1(t_2)$).

B. Time changes and policy improvement

Lemma 4.1: Let $p < 1$ be some positive constant. For any multi-policy $\mathbf{u}(t) = (u_1(t), ..., u_n(t))$ satisfying $u(t) = \sum_{i=1}^n u_i(t) \le p$ for all t, define the policy $\mathbf{v}(t) = (v_1(t), ..., v_n(t))$ where $v_i(t) = u_i(t/p)/p$ or equivalently, $u_i(t) = p\, v_i(tp)$, $i = 1, ..., n$. Define by $X_i(t)$ the state trajectories under $\mathbf{u}(t)$, and let $\overline{X}_i(t)$ be the state trajectories under $\mathbf{v}(t)$. Then $X_i(t) = \overline{X}_i(tp)$, and hence $X(t) = \overline{X}(tp)$.

Proof. We look for $s(t)$ such that $X_i(t) = \overline{X}_i(s(t))$. In the non-overwriting and overwriting cases, respectively:

$$\frac{d\overline{X}_i(s(t))}{dt} = \frac{d\overline{X}_i(s)}{ds}\frac{ds(t)}{dt} = \left[\beta v_i(s)(1-\overline{X}(s))\right]\frac{ds(t)}{dt},$$

$$\frac{d\overline{X}(s(t))}{dt} = \frac{d\overline{X}(s)}{ds}\frac{ds(t)}{dt} = \left[\beta v(s)(1-\overline{X}(s))\right]\frac{ds(t)}{dt}.$$

Taking $s(t) = pt$, we end up with the desired result for all $i = 1, \dots, K$. ◇

An acceleration $\mathbf{v}(t)$ of $\mathbf{u}(t)$ from a given time t' is defined as $v_i(t) = u_i(t)$ for $t \le t'$ and $v_i(t) = u_i(t' + (t-t')/p)/p$ otherwise, for all $i = 1, ..., n$. We now introduce a policy improvement procedure.

Definition 4.1: Consider some policy $\mathbf{u}(t)$ such that $0 < u(t) \le p$ for some $0 < p < 1$ for all t in some interval $S = [a,b]$ and that $\int_b^c u(t)dt > 0$ for some $c > b$. Let $\mathbf{w}(t)$ be the policy obtained from $\mathbf{u}(t)$ by
(i) accelerating it by a factor of $1/p$ between instants a and $d := a + p(b-a)$,
(ii) from time d till time $e := c - (1-p)(b-a)$, use $\mathbf{w}(t) = \mathbf{u}(t + b - d)$. Then use $\mathbf{w}(t) = 0$ till time c.

Lemma 4.2: There exists some policy improvements of $\mathbf{u}(t)$ by $\mathbf{w}(t)$ such that $P_s(\tau = c, \mathbf{w}) > P_s(\tau = c, \mathbf{u})$.

Proof. Let $\mathbf{X}(t)$ and $\overline{\mathbf{X}}(t)$ be the state processes under $\mathbf{u}(t)$ and $\mathbf{w}(t)$, respectively.

- For non-overwriting policies: Consider $\mathbf{w}(t)$ obtained by the improvement of Def. 4.1. We can easily show that $Z_i(c) := \int_a^c X_i(t)dt \le Z_i'(c) := \int_a^c \overline{X}_i(t)dt$ for all i, with strict inequality for some i. Owing to the expression of $P_s(c, \mathbf{u}) = \prod_{i=1}^K \zeta(Z_i(c))$ given in Section II.A, we end up with the lemma for non-overwriting policies.
- For overwriting policies: From a policy $\mathbf{u}(t)$ with the features of Def. 4.1, consider a policy $\mathbf{w}(t)$ obtained by substituting $\mathbf{w}(t)$ by $w(t) = \sum_{i=1}^K w_i(t)$ in Def. 4.1. Then, reasoning with $X(t)$ instead of $X_i(t)$, we can show in the same way as above that $Z'_{total}(c) > Z_{total}(c)$, $X(t) \le X'(t)$ and $X(t) < X'(t)$ for some $t \in [a, c[$. Having $\mathbf{u}(t)$, $w(t)$, $X(t)$ and $\overline{X}(t)$ fixed, we can choose $\mathbf{w}(t)$ such that $X_1'(t) = X_1(t) + X'(t) - X(t)$ and $X_i'(t) = X_i(t)$ for $t = 2, \dots, K$ (it is sufficient to express $\overline{X}_i(t)$ as a function of $v_i(t)$ and $v(t)$ from the ODE $\frac{d\overline{X}_i(t)}{dt} = \beta(v_i(t) - v(t)\overline{X}_i(t))$, and then express v_i by equalizing $\overline{X}_i(t)$ to the desired function). Finally we get $Z_1'(t) > Z_1(t)$ and $Z_i'(t) = Z_i(t)$ for $t = 2, \dots, K$. Whereby the result for overwriting policies. ◇

C. General optimal policies

Theorem 4.1: Let $K \ge 2$. Then an optimal policy for **P1**, named *piecewise-threshold policy*, exists with the following structure:

- (i) There are thresholds, $s_i \in [t_i, t_{i+1}]$, $i = 1, ..., K-1$ for which $u(t) = 1$ for $t \in [t_i, s_i[$. During the intervals $[s_i, t_{i+1})$, $u(t) = 0$.

- (ii) After time t_K it is optimal to always transmit a packet. An optimal policy u satisfies $u(t) = 1$ for all $t \geq t_K$.

Proof. (i) Let $\mathbf{u}(t)$ be an arbitrary policy. Remember that $u(t) = \sum_j u_j(t)$. Assume that it does not satisfy (i) above. Then there exists some $i = 1, ..., K-1$, such that $\mathbf{u}(t)$ is not a threshold policy on the interval $T_i := [t_i, t_{i+1})$. Hence there is a closed interval $S = [a, b] \subset T_i$ such that for some $p < 1$, $u(t) \leq p$ for all $t \in S$ and $\int_b^{t_{i+1}} u(t)dt > 0$. Then $\mathbf{u}(t)$ can be strictly improved according to Lemma 4.2 and hence cannot be optimal.
(ii) Assume that the threshold s_K satisfies $s_K < \tau$. It is straightforward to show that by following $\mathbf{u}(t)$ till time s_K and then switching to any policy that satisfies $u_i(t) > 0$ for $s_K < t \leq \tau$ for all i, $P_s(\tau)$ strictly increases. ⋄

Remark 4.1: The above theorem allows to conclude that the optimal policy for P1, i.e., for maximizing $P_s(\tau)$ over all possible policies, can be searched in the set of the policies given by $\{\mathbf{u}(t), \{s_i\}_i\}$ for $i = 1, \ldots, K$, where $u(t) = 1$ in every interval $[t_i, s_i[$, in $[t_K, \tau[$, and is zero outside. It is worth noting that WC policies are a particular case of piecewise-threshold policies when all the $s_i = t_{i+1}$, for $i = 1, \ldots, K-1$.

V. The Constrained Problem

Let $\mathbf{u}$ be any policy that achieves the constraint $\mathcal{E}(\tau) = \varepsilon X_e$ as defined in Section II-B. We make the following observation. The constraint involves only $X(t)$. It thus depends on the individual $X_i(t)$'s only through their sum; the sum $X(t)$, in turn, depends on the policies u_i's only through their sum $u(t) = \sum_{i=1}^K u_i(t)$.

WC policies. If a policy is WC and has to meet an energy constraint, then it is such that: $u = 1$ till some time s and is then zero. s is the solution of $X(s) = z + X_e$, either in the overwriting or non-overwriting cases. Algorithm A can be used to generate the optimal policy components $u_i(t)$, $i = 1, \ldots, K$: in particular, it will perform the same type of equalization performed in the unconstrained case until the bound is reached and it will stop thereafter.

General policies. Any policy $\mathbf{u}$ that is not as described by (i)-(ii) in Theorem 4.1 (i.e., with some $0 < u(t) < 1$ for a certain time interval) can be strictly improved by using Lemma 4.2. Thus the optimal policies satisfying a certain energy constraint is piecewise-threshold, with thresholds s_i possibly strictly lower than t_{i+1}, for $i = 1, \ldots, K$ and $t_{i+1} = \tau$ for the sake of notation.

VI. Adding Fixed Amount of Redundancy with Optimal Block-Codes

We now consider adding forward error correction: we add H redundant packets and consider the new file that now contains $K + H$ packets. The channel between the source and the destination up to time τ can be seen as an erasure channel. For such channel, maximum-distance separable erasure codes exist, such as Reed-Solomon codes, for which it is sufficient to receive any K packets out of the $K + H$ to ensure successful decoding of the entire file at the receiver (i.e., retrieval of the K information packets). We assume that all redundant packets are available (created at the source) once all K information packets have been received, that is from t_K onwards. The same results would hold if coded packets are created at separate times $t_i > t_K$ for $i = K+1, \ldots, K+H$. Now we introduce the result that specifies how to optimize WC policies when block codes are adopted.

Theorem 6.1: (i) Assume that there exists some policy $\mathbf{u}$ such that $\sum_{i=1}^{K+H} u_i(t) = 1$ for all t, and such that $Z_i(\tau)$ is the same for all $i = 1, ..., K+H$ under $\mathbf{u}$. Then $\mathbf{u}$ is optimal for P1.
(ii) Algorithm A, with $K + H$ replacing K and $\tau = \infty$, produces a policy which is optimal for P2.
(iii) Algorithm A, with $K + H$ replacing K and taking $\tau = \infty$, produces a policy which is also optimal for P1 in the overwriting case.

Proof: For $i = 1, \ldots, K + H$, let p_i be the probability of successful delivery of packet i by τ: $p_i = \zeta(Z_i) = 1 - \exp(-\lambda \int_0^\tau X_i(t)dt)$. Let E be a set of $|E|$ pairwise different elements of $\{1, \ldots, K + H\}$. The probability of delivery of the K packets by time τ is given by:

$$P_s(\tau, K, H) = \sum_{Z=K}^{K+H} \sum_{E:|E|=Z} \prod_{i \in E} p_i \prod_{i \notin E} (1 - p_i)$$

Consider any pair of different elements $i, j \in \{1, ..., K+H\}$. Denote $V = \{1, ..., K+H\} \setminus \{i, j\}$. The success probability can be decomposed as

$$\begin{aligned} P_s(\tau, K, H) &= p_i p_j \sum_{\substack{h \subseteq V \\ |h| \geq K-2}} \prod_{r \in h} p_r \prod_{s \in V \setminus h} \Big(1 - p_s\Big) \\ &+ p_i(1 - p_j) \sum_{\substack{h \subseteq V \\ |h| \geq K-1}} \prod_{r \in h} p_r \prod_{s \in V \setminus h} \Big(1 - p_s\Big) \\ &+ (1 - p_i)p_j \sum_{\substack{h \subseteq V \\ |h| \geq K-1}} \prod_{r \in h} p_r \prod_{s \in V \setminus h} \Big(1 - p_s\Big) \\ &+ (1 - p_i)(1 - p_j) \sum_{\substack{h \subseteq V \\ |h| \geq K}} \prod_{r \in h} p_r \prod_{s \in V \setminus h} \Big(1 - p_s\Big) \end{aligned} \quad (1)$$

Equation (1) can be rewritten $P_s(\tau, K, H) = p_i p_j A + p_i(1 - p_j)B + p_j(1-p_i)C + (1-p_i)(1-p_j)D$ with obvious meaning of the notation; observe that A, B, C, D do not depend on p_i (or Z_i) and p_j (or Z_j). Further rearranging the terms we obtain

$$P_s(\tau, K, H) = (A-B-C+D)p_i p_j + (B-D)p_i + (C-D)p_j + D$$

First, observe that $B = C$. Also, $A \geq B \geq D$. For any i and j in $\{1, ..., K+H\}$ we can thus write

$$P_s(\tau, K, H) = g_1 p_i p_j + g_2(p_i + p_j) + g_3 \quad (2)$$

where $g_1 = (A - B - C + D)$, $g_2 = (B - D)$ and $g_3 = D$ are functions only of $\{p_m, m \neq i, m \neq j\}$.
Now, fix all p_m, $m \neq i, j$: optimizing the WC policy with respect to Z_i and Z_j means finding Z_i and Z_j maximizing (2), with $Z_i + Z_j = S = Z_{total} - \sum_{r \neq i,j} Z_r$. Note that S is fixed if Z_{total} and all the Z_r for $r \neq i, j$ are fixed. Thus we

have $p_j = 1 - \frac{e^{-\lambda S}}{1-p_i}$. It follows that $P_s(\tau, K, H) = f_S(p_i)$, where

$$f_S(p_i) = g_1 p_i\Big(1 - \frac{e^{-\lambda S}}{1-p_i}\Big) + g_2\Big(p_i + \big(1 - \frac{e^{-\lambda S}}{1-p_i}\big)\Big) + g_3$$

and it follows

$$\frac{df_S(p_i)}{dp_i} = \big((1-p_i)^2 - e^{-\lambda S}\big)\frac{g_1+g_2}{(1-p_i)^2},$$

As $g_1 + g_2 \geq 0$, the maximum of $f_S(p_i)$ is attained for $p_i = 1 - e^{-\lambda S/2}$. Hence, for a given S, it is optimal to equalize pairwise Z_i and Z_j. Moreover, $f_S(.)$ is increasing with S, and a WC policy achieves the highest feasible Z_{total} by construction. If all Z_i are equal, then $S = 2Z_{total}/(K+1)$ is also maximum for all pairs $\{i, j\}$. This implies (i). Item (ii) can be proven in the same way as in Theorem 3.2: indeed, we can show that $P_s(\tau, K, H)$ is Schur-concave in $\mathbf{Z}$ in the same way as in Lemma 10.1. (iii) Z_{total} is maximized with a WC policy, and the Z_i's can be equalized as much as the t_i's allow because the source can always overwrite packets. ◇

Also, the other results that we had for the case of no redundancy can be obtained here as well (those for P1, CP1 and CP2).

TABLE III
ALGORITHM B

C1 Use $\mathbf{p}_t = e_1$ at time $t \in [t_1, t_2)$.
C2 Use $\mathbf{p}_t = e_2$ from time t_2 till $s(1,2) = \min(S(2,\{1,2\}), t_3)$. If $s(1,2) < t_3$ then switch to $\mathbf{p}_t = \frac{1}{2}(e_1 + e_2)$ till time t_3.
C3 Repeat the following for $i = 3, ..., K-1$:
C3.1 Set $j = i$. Set $s(i,j) = t_i$
C3.2 Use $\mathbf{p}_t = \frac{1}{i+1-j}\sum_{k=j}^{i} e_k$ from time $s(i,j)$ till $s(i,j-1) := \min(S(j,\{1,2,...,i\}), t_{i+1})$. If $j = 1$ then end.
C3.3 If $s(i,j-1) < t_{i+1}$ then take $j = \min(j : j \in J(t,\{1,...,i\}))$ and go to step [C3.2].
C4 From $t = t_K$ to $t = \tau$, use all transmission opportunities to send an RLC of information packets, with coefficients picked uniformly at random in $\mathbb{F}_q$.

VII. Rateless Codes

In this section, we want to identify the possible rateless codes for the settings described in Section II, and quantify the gains brought by coding. Rateless erasure codes are a class of erasure codes with the property that a potentially limitless sequence of coded packets can be generated from a given set of information packets. Information packets, in turn, can be recovered from any subset of the coded packets of size equal to or only slightly larger than K (the amount of additional needed packets for decoding is named "overhead").

A. Rateless coding after t_K

As in the previous section, we assume that redundant packets are created only after t_K, i.e., when all information packets are available. The case when coding is started before receiving all information packets is postponed to the subsection. Since coded packets are generated after all information packets have been sent out, the code must be *systematic* because information packets are part of the coded packets. Amongst rateless codes, LT codes [17] and Raptor codes [18] are near to optimal in the sense that the overhead can be arbitrarily small with some parameters. The coding matrix of each of them has a specific structure in order to reduce encoding and decoding complexity. Only Raptor codes exist in a systematic version. Random network codes [19] are more general rateless codes as generating coded packets relies on random linear combinations (RLCs) of information packets, without any (sparsity) constraint for the matrix of the code. Their overhead can be considered as 0 for high enough finite field order. That is why in this section we provide the analysis of the optimal control for network codes. But, it is straightforward to extend these results to systematic Raptor codes.

After t_K, at each transmission opportunity, the source sends a redundant packet (an RLC of all information packets) with probability $u(t)$. Indeed, from t_K, any sent RLC carries the same amount of information of each information packet, and hence from that time, the policy is not function of a specific packet anymore, whereby $u(t)$ instead of $\mathbf{u}(t)$. In each sent packet, a header is added to describe what are the coefficients, chosen uniformly at random, of each information packet [20]. The decoding of the K information packets is possible at the destination if and only if the matrix made of the headers of received packets has rank K. Note that, in our case, the coding is performed only by the source since the relay nodes cannot store more than one packet.

Theorem 7.1: Let us consider the above rateless coding scheme for coding after t_K.
(i) Assume that there exists some policy $\mathbf{u}(t)$ such that $\sum_{i=1}^{K-1} u_i(t) = 1$ for all t, and such that $Z_i(\tau)$ is the same for all $i = 1, \ldots, K-1$ under $\mathbf{u}(t)$. Then $\mathbf{u}(t)$ is optimal for P1.
(ii) Algorithm B produces a policy which is optimal for P2.
(iii) Algorithm B produces a policy which is also optimal for P1 in the overwriting case.

Proof: Let E be a set made of pairwise different elements from $\{1, \ldots, K-1\}$. Let us take the notation of proof of Theorem 6.1. We have $P_s(\tau, K) = \sum_{E \subset \{1,\ldots,K-1\}} \Big(\prod_{i\in E} p_i \prod_{i \in \{1,\ldots,K-1\}\setminus E}(1-p_i)\Big) Q(|E|)$ where $Q(|E|)$ denotes the probability that the received coded packets, added to the $e = |E|$ received information packets, form a rank K matrix. Let P_m be the probability that exactly m coded packets are received at the destination by time τ. Let consider the probability that, given that $m \geq K - e$ coded packets have been received, these packets form a rank K matrix with the received e information packets. Then $Q(e)$ is expressed as: $Q(e) = \sum_{m=K-e}^{\infty} P_m \prod_{r=0}^{K-e-1}\Big(1 - \frac{1}{q^{m-r}}\Big)$. It can be seen that $Q(e)$ depends on $u_i(t)$, for $i = 1, \ldots, K-1$, through P_m, and hence only through the sum $\sum_{i=1}^{K-1} u_i(t)$, which is 1 (resp. 0) for $t < t_K$ (resp. $t \geq t_K$). Consider a pair $\{i, j\}$ in $\{1, \ldots, K-1\}$, and let $V = \{1, \ldots, K-1\}\setminus\{i, j\}$ and define $Q(\geq a) = \sum_{e=a}^{\infty} Q(e)$ for the sake of notation.

We have

$$P_s(\tau, K) = p_i p_j \sum_{\substack{h \subseteq V \\ |h| \geq 0}} Q(\geq K-2-h) \prod_{r \in h} p_r \prod_{s \in V \setminus h} \left(1 - p_s\right)$$

$$+ p_i(1-p_j) \sum_{\substack{h \subseteq V \\ |h| \geq 0}} Q(\geq K-1-h) \prod_{r \in h} p_r \prod_{s \in V \setminus h} \left(1 - p_s\right)$$

$$+ (1-p_i)p_j \sum_{\substack{h \subseteq V \\ |h| \geq 0}} Q(\geq K-1-h) \prod_{r \in h} p_r \prod_{s \in V \setminus h} \left(1 - p_s\right)$$

$$+ (1-p_i)(1-p_j) \sum_{\substack{h \subseteq V \\ |h| \geq 0}} Q(\geq K-h) \prod_{r \in h} p_r \prod_{s \in V \setminus h} \left(1 - p_s\right)$$

Therefore, the definition of constants A, B, C and D are different than in proof of Theorem 6.1, but their properties remain the same. Thus, in the same way, we can derive the desired result. The proofs for (ii) and (iii) are the same as in Theorem 6.1. $\diamond$

B. Rateless coding before t_K

We now consider the case where after receiving packet i and before receiving packet $i+1$ at the source, we allow to code over the available information packets and to send the resulting coded packets between t_i and t_{i+1}. LT codes and Raptor codes require that all the information packets are available at the source before generating coded packets. Owing to their fully random structure, network codes do not have this constraint, and allow to generate coded packets online, along the reception of packets at the source. We present how to use network codes in such a setting. The objective is the successful delivery of the entire file (the K information packets) by time τ[1]. Information packets are not sent anymore, only coded packets are sent instead.

Theorem 7.2: (i) Given any forwarding policy $u(t)$, it is optimal for P1 and P2 to send coded packets resulting from RLCs of all the information packets available at the time of the transmission opportunity with probability $u(t)$.
(ii) For any policy $u(t)$, the probability of successful delivery of the entire file is given by

$$P_s(\tau) = \sum_{j=0}^{K-1} \sum_{k_1 > \cdots > k_j} \sum_{l_0 = K-k_1}^{K} \cdots \sum_{l_i = K-k_{i+1}-L_{i-1}}^{K-L_{i-1}} \cdots$$

$$\sum_{l_j = K-L_{j-1}}^{k_j} \sum_{m_0 = l_0}^{\infty} \cdots \sum_{m_j = l_j}^{\infty} \prod_{i=0}^{j} f(l_0, \ldots, l_i, m_i) \,,$$

with $L_i = \sum_{j=0}^{i-1} l_j$, $f(l_0, \ldots, l_i, m_i) = P(m_i)\binom{m_i}{l_i} \prod_{r=0}^{l_i - 1} \left(1 - \frac{1}{q^{K-L_{i-1}-r}}\right)$, where

$$P(m_k) = \exp(-\Lambda_k)\frac{\Lambda_k^{m_k}}{m_k!} \,, \quad \Lambda_k = \lambda \int_0^{\tau} Y_k(t)\, dt \,,$$

$$Y_k(t) = (t \geq t_{k+1})\lambda \int_{t_k}^{\min(t, t_{k+1})} u(v) \exp\left(-\lambda \int_0^{v} u(s)\, ds\right) dv \,.$$

[1]We do not have constraints on making available at the destination a part of the K packets in case the entire file cannot be delivered.

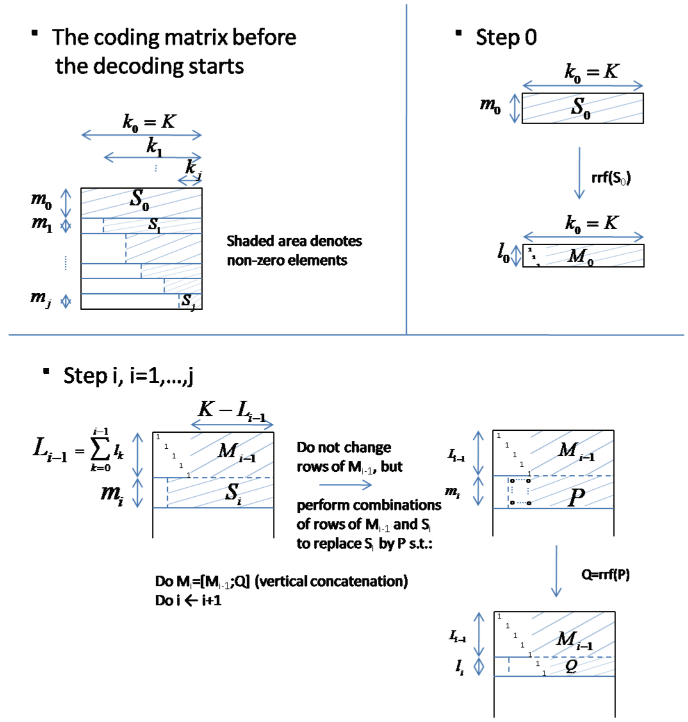

Fig. 2. The decoding process formatted so as to express the successful decoding condition. At each step i, $i = 0, \ldots, j$, l_i is defined as the number of non-zero rows of rrf(P).

Proof: The proof is based on finding the condition for having full-rank decoding matrix, given that the packets of each class arrive according to a Poisson process owing to the assumed mobility model. For all $k = 1, \ldots, K$, let $E(k)$ be $E(k) = \{1, \ldots, k\}$. For short, we say that a coded packet is "a packet over $E(k)$" if the coefficients of the first k information packets are chosen uniformly at random in $\mathbb{F}_q$, while the others are zero. We have the following definitions:

- The received packets are over $E(k_i)$, with $K = k_0 > k_1 > k_2 > \cdots > k_j \geq 1$.
- j is such that $0 \leq j < K$, and denotes the number of pairwise different $k_i \neq K$. We set $k_{j+1} = 0$.
- m_i, $i = 0, \ldots, j$ is the number of received packets over $E(k_i)$.
- S_i is the sub-matrix made of all the received packets over $E(k_i)$.

In order to recover the K information packets, the matrix which has to be full-rank is composed of the S_i, for $i = 0, \ldots, j$. Let rrf(M) denote the reduced row form of any matrix M, i.e., the matrix resulting from Gaussian elimination on M, without column permutation. Let us consider the process described in Figure 2. At each step i, $i = 0, \ldots, j$, l_i is defined as the number of non-zero rows of rrf(P). Therefore, for the coding matrix to have rank K, it is necessary and sufficient to have: $l_0 \geq K - k_1$, $l_1 \geq k_1 - k_2 - (l_0 - (K - k_1))$, $\ldots$, $l_i \geq K - k_{i+1} - L_{i-1}$, $l_j \geq K - L_{j+1}$, with $L_i = \sum_{j=0}^{i-1} l_j$. Therefore we can express $P_s(\tau)$:

$$P_s(\tau) = \sum_{j=0}^{K-1} \sum_{k_1 > \cdots > k_j} \sum_{l_0 = K-k_1}^{K} \cdots \sum_{l_i = K-k_{i+1}-L_{i-1}}^{K-L_{i-1}} \cdots$$

$$\sum_{l_j=K-L_{j-1}}^{k_j} \sum_{m_0=l_0}^{\infty} \cdots \sum_{m_j=l_j}^{\infty} \prod_{i=0}^{j} f(l_0,\dots,l_i,m_i)\,,$$

where $f(l_0,\dots,l_i,m_i)$ is a joint probability: $f(l_0,\dots,l_i,m_i) = P(l_i,m_i|l_0,\dots,l_{i-1}) = P(l_i|m_i,l_0,\dots,l_{i-1})P(m_i)$. We have $P(l_i|m_i,l_0,\dots,l_{i-1}) = \binom{m_i}{l_i}\prod_{r=0}^{l_i-1}\left(1-\frac{1}{q^{K-L_{i-1}-r}}\right)$. Also, $P(m_i)$ is the probability that m_i coded packets over $E(k_i)$ have reached the destination (by time τ). This probability is therefore dependent on $u_i(t)$, for $i=0,\dots,j$.

Now we are able to analyze how must the k_i be chosen, for $i=0,\dots,j$, so as to maximize $P_s(\tau)$, when the other system parameters are fixed. For every $i=0,\dots,j$, we can see from the above expression of $P_s(\tau)$ that it is maximized if l_i needs to be as less as possible, i.e., when k_{i+1} and L_{i-1} are maximized. Note that once k_i is set, l_i is only dependent on $u_i(t)$ and the mobility process (transmission opportunities). Thus, for given i, the probability to receive the required l_i is maximized for maximized k_{i+1}. Thus it is optimal to send packets coded over all the available information packets. Therefore $u_i(t)$ is $u(t)$ for i such that k_i is maximum at time t, 0 otherwise. This proves part (i) of the theorem.

Owing to the mobility process and the buffer size limited to one packet, the arrival process of the packets over $E(k)$ at the destination is a non-homogeneous Poisson process of parameter $\Lambda_k(t)$. Let $\Lambda_k = \Lambda_k(\tau)$. We have: $P(m_k) = \exp(-\Lambda_k)\frac{\Lambda_k^{m_k}}{m_k!}$, where $\Lambda_k = \lambda\int_0^\tau Y_k(t)\,dt$. Then the expression of $Y_k(t)$ is derived thanks to a fluid approximation. $\diamond$

C. Discussion on extensions

Complexity: The decoding complexity of random network codes ($O(K^3)$) may not be a problem in the DTN context as the decoding by progressive/incremental Gauss-Jordan elimination as the packets arrive at a slow rate can limit the computational burden per time unit. If one would like to use systematic Raptor codes for coding after t_K, we can show that Theorem 7.1 would hold unchanged.If we consider low complexity rateless codes such as LT codes instead of random codes, the result of Theorem 7.2 would hold for the following reason. Since the symbol error rate of LT codes is given by the probability that an input symbol be "covered" by an output symbol [17], the probability not to recover an input symbol is minimized if the set of input symbols involved in an output symbol encompasses all the input symbols.

Multi-hop routing: Our work holds also for multi-hop routing, because the optimality condition stated in Th. 3.1 relies on the concavity of the function $\zeta(h) = 1-\exp(-\lambda h)$. The source cannot keep track of the exact $Z_i(t)$ anymore but it can still get an approximation of the $Z_i(t)$: either by implementing the analytical fluid model, or by an external observer [21]. As well, Section IV.B can be extended to the case of multi-hop, based on what has been done in [4]. Furthermore, if network coding is used (relays re-encode source coded packets), then all the results hold, only the thresholds s_i obtained from the optimization procedure will change.

Buffer size: The case of multi-packet memory can be derived, such as in [13] (Section 6). For the sake of clarity, we focused on single packet memory.

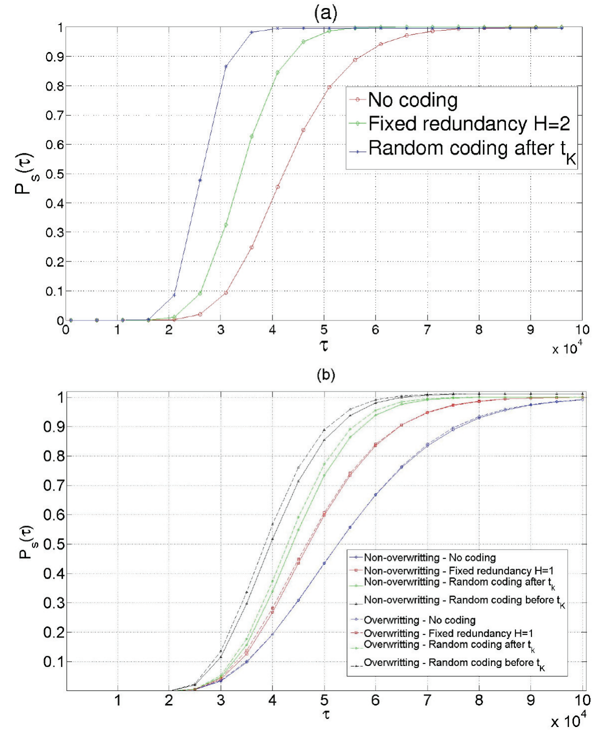

Fig. 3. $P_s(\tau)$ for WC policies based on Algo. A, under various coding schemes. Parameters are $N = 100$, (a) $\beta = 2.10^{-5}$, $K = 10$, $\mathbf{t} = (119, 1299, 1621, 1656, 3112, 3371, 4693, 5285, 5688, 7942)$, non-overwriting case. (b) $\beta = 8.10^{-6}$, $K = 4$, $\mathbf{t} = (1000, 5000, 7000, 20000)$.

VIII. Numerical Analysis

A. Numerical analysis of WC policies

Comparing the delivery delay CDFs for the different coding schemes, it can be seen that coding allows simplification of the scheduling policy, with or without energy constraint. Figure 3 is an example of numerical comparison between the four coding schemes for WC policies designed with Algo. A. We can conclude that (i) as soon as coding is performed, it saves the source maintaining states $Z_i(t)$, and (ii) the sooner coding is performed at the source, the better. We verify that a WC policy with overwriting performs better, as we have demonstrated the optimality of a WC policy in such case.

B. Numerical optimization of piecewise-threshold policies

Figure 4 shows the success probability of piecewise-threshold policies, under various coding schemes and for $K=3$, as well as the variation of best s_1 against τ. For the overwriting case, the value of s_1 is not plot because constant equal to $t_2 = 7000$. For the sake of simplicity in order to study the threshold evolution, we have considered that $\mathbf{u}(t)$ between $[t_i, s_i[$, for all $i=1,\dots,K-1$, is given by Algo. A. For each value of τ, the two-dimensional optimization of the parameters (s_1, s_2) pertains to the class of nonlinear optimization problem, for which we have used Differential Evolution [22]. The optimization of the s_i can be performed

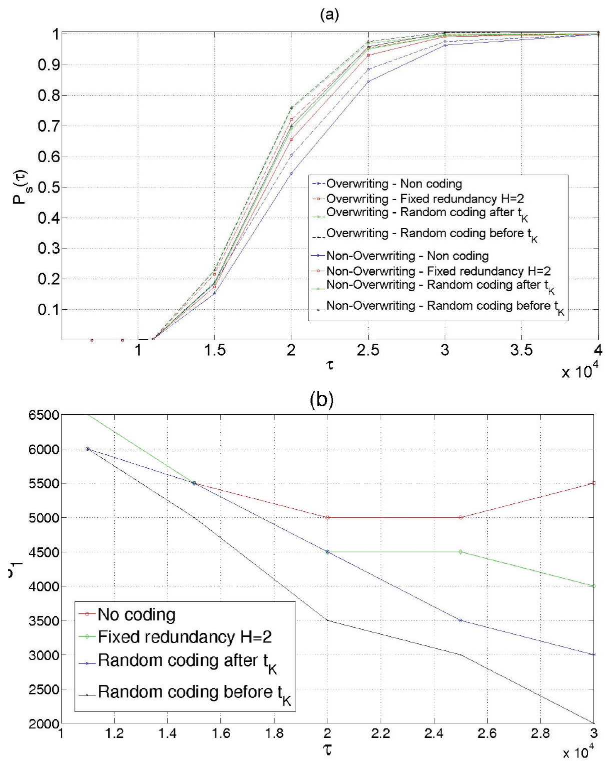

Fig. 4. Parameters are $N = 100$, $\beta = 2.10^{-5}$, $K = 3$, $\mathbf{t} = (1, 7000, 10000)$.(a) $P_s(\tau)$ for an optimized piecewise-threshold policy. (b) Value of threshold $s_1(\tau)$ in the non-overwriting case.

offline provided that the t_i, for $i = 1, \ldots, K$, are known in advance. If not, some heuristics can be used to estimate the inter-arrival times of packets based on, e.g., previous arrivals, or on τ, such as τ/K or T/K with some $T < \tau$. We do not comment s_2 as the settings are such that s_2 is equal to t_3 for all coding schemes and all values of τ. We can interpret the variations of s_1 thanks to the comments made after Theorem 3.1. Let us first give some general comments holding for all coding schemes. We can see that s_1 is relatively high when τ is close to t_3. Indeed, in such case, the limiting parameter for $Z_3(\tau)$ (and $Z_2(\tau)$) is $\tau - t_3$ (or $\tau - t_2$) instead of $X_1(s_1)$, and hence it is better to spread as many packet 1 as possible. Then s_1 starts to decrease: as τ increases, $\tau - t_3$ stops progressively being the limiting factor for $Z_3(\tau)$, and that becomes to be $X_1(s_1)$ which has hence to be limited. When τ increases even more, s_1 can get higher again because copies of packet 3 have time to spread enough, even though they spread slower because of higher $X_1(s_1)$ from $t = t_3$. Now regarding the specificities of each coding scheme, we can notice that the sooner coding is performed, the lower s_1 for all values of τ. As proven in the above section, it is better to propagate coded packets resulting from combination over as many as possible original packets. Therefore, when coding is used, coded packets shall be sent out later to be more useful. That can be a reason for s_1 to be that low for coding schemes. In the overwriting case, the thresholds for all coding schemes are maximum, as a WC policy is optimal in the non-constrained case for such kind of policy.

IX. Conclusions

We have addressed the problem of optimal transmission and scheduling policies in DTN with two-hop routing under memory and energy constraints, when the packets of the file to be transmitted get available at the source progressively. We solved this problem when the source can or cannot overwrite its own packets, and for WC and non WC policies. We extended the theory to the case of fixed rate systematic erasure codes and rateless random linear codes. Our model includes both the case when coding is performed after all the packets are available at the source, and also the important case of random linear codes, that allows for dynamic runtime coding of packets as soon as they become available at the source.

X. Appendix

Definition 10.1: (Majorization and Schur-Concavity [23]) Consider two n-dimensional vectors $d(1), d(2)$. $d(2)$ majorizes $d(1)$, which we denote by $d(1) \prec d(2)$, if

$$\sum_{i=1}^{k} d_{[i]}(1) \leq \sum_{i=1}^{k} d_{[i]}(2), \quad k = 1, ..., n-1,$$

$$\text{and} \sum_{i=1}^{n} d_{[i]}(1) = \sum_{i=1}^{n} d_{[i]}(2),$$

where $d_{[i]}(m)$ is a permutation of $d_i(m)$ satisfying $d_{[1]}(m) \geq d_{[2]}(m) \geq ... \geq d_{[n]}(m)$, $m = 1, 2$. A function $f : \mathbb{R}^n \to \mathbb{R}$ is Schur concave if $d(1) \prec d(2)$ implies $f(d(1)) \geq f(d(2))$. *Separable* Schur concave functions are defined in the following result [23, Proposition C.1 on p. 64].

Lemma 10.1: Assume that a function $g : \mathbb{R}^n \to \mathbb{R}$ can be written as the sum $g(\mathbf{d}) = \sum_{i=1}^{n} \psi(d_i)$ where ψ is a concave function from $\mathbb{R}$ to $\mathbb{R}$. Then g is Schur concave.

Theorem 10.1: $\log P_s(\tau, \mathbf{u})$ is Schur concave in $\mathbf{Z} = (Z_1, ..., Z_K)$. If $\mathbf{Z} \prec \mathbf{Z}'$ then $P_s(\tau, \mathbf{u}) \geq P_s(\tau, \mathbf{u}')$.

The policy $\mathbf{u}^*$ generated by Algorithm A is work conserving by construction. Let $\mathbf{Z}(\tau)$ and $\mathbf{Z}^*(\tau)$ denote the K-dimensional CCI vectors corresponding to a work conserving policy $\mathbf{u}$ and to $\mathbf{u}^*$, respectively. We show in the following that it holds $\mathbf{Z}^*(\tau) \prec \mathbf{Z}(\tau)$ for $\tau \geq 0$. Then Thm. 10.1 implies that $P_s^*(\tau, \mathbf{u}^*) \geq P_s(\tau, \mathbf{u})$, i.e., $\mathbf{u}^*$ is uniformly optimal over work conserving policies. It is then immediate to observe that $\mathbf{u}^*$ is optimal for P2 because it minimizes the expected delivery delay $E[D] = \int_0^\infty (1 - P_s(t))dt$. We now prove that $\mathbf{Z}^*(\tau) \prec \mathbf{Z}(\tau)$ for $\forall \tau \geq 0$. $\mathbf{u}^*$ is generated by Algo A such that for all t:

$\mathbf{u}^*$ maximizes the minimum of the CCIs:

$$\mathbf{u}^* = \arg \max_{\text{WC } \mathbf{u}} \min_{i: t_i \leq t} Z_i(t) \tag{3}$$

$\mathbf{u}^*$ minimizes the highest gap between two CCIs:

$$\mathbf{u}^* = \arg \min_{\text{WC } \mathbf{u}} \max_{i,j: t_i, t_j \leq t} |Z_i(t) - Z_j(t)| \tag{4}$$

For lighter notations, we omit t as well as $t_i, t_j \leq t$ when we refer to any i or j in the remainder of the proof. We have $\sum_{i=1}^{K} Z_{[i]} = \sum_{i=1}^{K} Z^*_{[i]}$. We want to prove that

$\sum_{i=1}^k Z_{[i]} \geq \sum_{i=1}^k Z^*_{[i]}$, $\forall k = 1, \ldots, K-1$.
Owing to property (3) of $\mathbf{u}^*$, we have $\min_i Z_i \leq \min_i Z^*_i$, i.e., $Z_{[K]} \leq Z^*_{[K]}$. Thus, let s_1 and s_2 be such that: $s_1 \leq i$, for $Z_{[i]} \leq Z^*_{[i]}$, $s_2 \leq i < s_1$, for $Z^*_{[i]} \leq Z_{[i]}$, $i < s_2$, for $Z_{[i]} \leq Z^*_{[i]}$.

Let us prove by contradiction that s_2 can be only 1. If s_2 exists, then $Z_{[1]} \leq Z^*_{[1]}$. We would have then $Z_{[1]} - Z^*_{[K]} \leq Z^*_{[1]} - Z^*_{[K]}$. Since $\max_{i,j} |Z_i(t) - Z_j(t)| = Z_{[1]} - Z_{[K]}$, $\mathbf{u}^*$ would not satisfy property (4) anymore. Hence, we cannot have $s_2 \geq 2$. Thus we have: $Z_{[i]} \leq Z^*_{[i]}$ for $s_1 \leq i$ and $Z^*_{[i]} \leq Z_{[i]}$ for $1 \leq i < s_1$. It is then straightforward to show that for all $k = 1, \ldots, K$, $\sum_{i=1}^k Z_{[i]} \geq \sum_{i=1}^k Z^*_{[i]}$, i.e., that $\mathbf{Z}^* \prec \mathbf{Z}$. $\diamond$

REFERENCES

[1] E. Altman, F. De Pellegrini, and L. Sassatelli, "Dynamic control of coding in delay tolerant networks," in *Proc. 2010 IEEE INFOCOM*, pp. 1–5.

[2] E. Altman and F. De Pellegrini, "Forward correction and Fountain codes in delay tolerant networks," in *Proc. 2009 IEEE INFOCOM*, pp. 1–5.

[3] T. Spyropoulos, K. Psounis, and C. Raghavendra, "Efficient routing in intermittently connected mobile networks: the multi-copy case," *ACM/IEEE Trans. Netw.*, vol. 16, no. 1, pp. 77–90, Feb. 2008.

[4] E. Altman, T. Başar, and F. De Pellegrini, "Optimal monotone forwarding policies in delay tolerant mobile ad-hoc networks," in *Proc. 2008 ACM/ICST Inter-Perf.*

[5] J. Metzner, "An improved broadcast retransmission protocol," *IEEE Trans. Commun.*, vol. 32, no. 6, pp. 679–683, June 1984.

[6] J. Nonnenmacher, E. Biersack, and D. Towsley, "Parity-based loss recovery for reliable multicast transmission," *IEEE/ACM Trans. Netw.*, vol. 6, no. 4, pp. 349–361, 1998.

[7] S. Jain, M. Demmer, R. Patra, and K. Fall, "Using redundancy to cope with failures in a delay tolerant network," *SIGCOMM Comput. Commun. Rev.*, vol. 35, no. 4, pp. 109–120, 2005.

[8] Y. Wang, S. Jain, M. Martonosi, and K. Fall, "Erasure-coding based routing for opportunistic networks," in *Proc. 2005 ACM SIGCOMM Workshop Delay-Tolerant Netw.*, pp. 229–236.

[9] J. Widmer and J.-Y. Le Boudec, "Network coding for efficient communication in extreme networks," in *Proc. 2005 ACM SIGCOMM Workshop on Delay-Tolerant Networking*, pp. 284–291.

[10] Y. Lin, B. Liang, and B. Li, "Performance modeling of network coding in epidemic routing," in *Proc. 2007 ACM MobiSys Workshop Mobile Opportunistic Netw.*, pp. 67–74.

[11] Y. Lin, B. Li, and B. Liang, "Efficient network-coded data transmissions in disruption tolerant networks," in *Proc. 2008 IEEE INFOCOM*, pp. 1508–1516.

[12] R. Groenevelt and P. Nain, "Message delay in MANETs," in *Proc. 2005 ACM SIGMETRICS*, pp. 412–413.

[13] X. Zhang, G. Neglia, J. Kurose, and D. Towsley, "Performance modeling of epidemic routing," *Elsevier Comput. Netw.*, vol. 51, pp. 2867–2891, July 2007.

[14] T. G. Kurtz, "Solutions of ordinary differential equations as limits of pure jump Markov processes," *J. Applied Probability*, vol. 7, no. 1, pp. 49–58, 1970.

[15] M. Benaïm and J.-Y. Le Boudec, "A class of mean field interaction models for computer and communication systems," *Performance Evaluation*, 2008.

[16] E. Altman, G. Neglia, F. De Pellegrini, and D. Miorandi, "Decentralized stochastic control of delay tolerant networks," in *Proc. 2009 IEEE INFOCOM*, pp. 1134–1142.

[17] M. Luby, "LT Codes," in *Proc. 2002 IEEE Symp. Foundations Comput. Sciences*, pp. 271–280.

[18] A. Shokrollahi, "Raptor codes," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2551–2567, June 2006.

[19] D. S. Lun, M. Médard, and M. Effros, "On coding for reliable communication over packet networks," in *Proc. 2004 Annual Allerton Conf. Commun., Control, Comput.*, pp. 20–29.

[20] C. Fragouli, J.-Y. Le Boudec, and J. Widmer, "Network coding: an instant primer," *SIGCOMM Comput. Commun. Rev.*, vol. 36, no. 1, pp. 63–68, 2006.

[21] A. Ali, E. Altman, T. Chahed, D. Fiems, M. Panda, and L. Sassatelli, "Estimating file-spread in delay tolerant networks under two-hop routing," in *Proc. 2012 IFIP Netw.*, pp. 277–290.

[22] K. Price and R. Storn, "Differential Evolution: a simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optimiz.*, vol. 11, pp. 341–359, 1997.

[23] A. W. Marshall and I. Olkin, *Inequalities: Theory of Majorization and its Applications*. Academic Press, 1979.

Eitan Altman received the B.Sc. degree in electrical engineering, the B.A. degree in physics, and the Ph.D. degree in electrical engineering from the TechnionIsrael Institute of Technology, Haifa, Israel, in 1984, 1984, and 1990,respectively, and the B.Mus. degree in music composition from Tel-Aviv University, Tel-Aviv, Israel, in 1990. Since 1990, he has been a Researcher with the National Research Institute in Informatics and Control (INRIA), Sophia-Antipolis, France. He is on the Editorial Boards of several scientific journals.Dr. Altman has been the (co)chairman of the program committee of international conferences.He received the Best Paper Award in the Networking 2006 conference and is a coauthor of two papers that have received the Best Student Paper awards at QoFis 2000 and Networking 2002. In October 2012, Dr. Altman was awarded the prize France-Telecom of Académie des Sciences.

Lucile Sassatelli received the M.Sc. and the Ph.D. degree in telecommunication engineering from the University of Cergy-Pontoise, France, in 2005 and 2008, respectively. From November 2008 until September 2009, she was a Postdoctoral Fellow in the Reliable Communications and Network Coding Group (RLE) at Massachusetts Institute of Technology (MIT), Cambridge, USA. Since September 2009, she has been an Assistant Professor at the University Nice Sophia Antipolis, France. Her research activities focus on network coding, mobile and delay-tolerant ad hoc networks and social networks. Dr. Sassatelli has been in the technical program committee of IEEE RAWNET/WNC3 2009 and 2013, WCSP 2012 and VTC 2013.

Francesco De Pellegrini received the Laurea degree in 2000 and the Ph.D. degree in 2004, both in Telecommunication Engineering, from the University of Padova. He is currently employed at CREATE-NET, Trento, Italy. His research interests involve performance evaluation and optimization of networking systems. F. De Pellegrini has been a TPC member of IEEE Infocom and a reviewer for several international networking conferences and journals. Francesco serves in the Steering Programm Commitee of ICST Mobiquitous and Robocomm. Francesco was the Vice-chair for the first edition of ICST Robocomm.