Distributed Storage Management of Evolving Files in Delay Tolerant Ad Hoc Networks

Eitan Altman[1], Philippe Nain[1], Jean-Claude Bermond[2]
[1] INRIA, 2004 Route des Lucioles, BP 93 06902 Sophia-Antipolis, France
[2] I3S/INRIA, 2004 Route des Lucioles, BP 93, 06902 Sophia-Antipolis, France
Email: {Eitan.Altman, Philippe.Nain, Jean-Claude.Bermond}@sophia.inria.fr

Abstract*— This work focuses on a class of distributed storage systems whose content may evolve over time. Each component or node of the storage system is mobile and the set of all nodes forms a delay tolerant (ad hoc) network (DTN). The goal of the paper is to study efficient ways for distributing evolving files within DTNs and for managing dynamically their content. We specify to dynamic files where not only the latest version is useful but also previous ones; we restrict however to files where a file has no use if another more recent version is available. There are $N+1$ mobile nodes including a *single* source which at some points in time makes available a new version of a *single* file F. We consider both the cases when (a) nodes do not cooperate and (b) nodes cooperate. In case (a) only the source may transmit a copy of the latest version of F to a node that it meets, while in case (b) any node may transmit a copy of F to a node that it meets. A file management policy is a set of rules specifying when a node may send a copy of F to a node that it meets. The objective is to find file management policies which maximize some system utility functions under a constraint on the resource consumption. Both myopic (*static*) and state-dependent (*dynamic*) policies are considered, where the state of a node is the age of the copy of F it carries. Scenario (a) is studied under the assumption that the source updates F at discrete times $t = 0, 1, \ldots$. During a slot $[t, t+1)$ the source meets any node with a fixed probability. We find the optimal static (resp. dynamic) policy which maximizes a general utility function under a constraint on the number of transmissions within a slot. In particular, we show the existence of a threshold dynamic policy. In scenario (b) F is updated at random points in time, with the consequence that between two meetings with the source a node does not know the age evolution of the version of F it holds. Under Markovian assumptions regarding nodes mobility and update frequency of F, we study the stability of the system (aging of the nodes) and derive an (approximate) optimal static policy. We then revisit scenario (a) when the source does not know parameter N (node population) and q (node meeting probability) and derive a stochastic approximation algorithm which we show to converge to the optimal static policy found in the complete information setting. Numerical results illustrate the respective performance of optimal static and dynamic policies as well as the benefit of node cooperation.

Keywords: Evolving files; Storage systems; Delay-tolerant (ad hoc) networks; Performance evaluation; Optimization.

I. INTRODUCTION

Much work has been devoted for the study of Delay Tolerant Networks (DTNs). Most of the work on protocol design has focused on the use of mobility in order to reach one or more disconnected destinations. The protocols are based on distribution of the file to relay nodes so as to increase the successful delivery probability [2], [3], [4], [9], [10].

In such applications, the DTN becomes a distributed storage system that contains copies of a file that is being transmitted. In this paper we focus on a special type of file that we call "dynamic file" or "evolving file". By that we mean a file whose content may evolve and change from time to time. One (or various) sources wish to make a file available to mobile nodes, and to send updates from time to time. Some examples are:

- a source has a file containing update information such as weather forecast or news headlines. The file changes incrementally from time to time with new information updates.
- a source wishes to make backups of some directories and to store them at another nodes in order to increase the reliability.
- some software updates or patches may be distributed regularly.

Several formats of dynamic files have been standardized:

- the RSS ("Real Simple Syndication" [5]) family of Web feed formats used to publish frequently updated content such as blog entries, news headlines, and podcasts in a standardized format. Updates can originate from various sources.
- another format called the "Atom Syndication Format" has been adopted as IETF Proposed Standard RFC 4287.

We specify to dynamic files where not only the latest version is useful but also previous ones; we restrict however to files where a file has no use if another more recent version is available. For example, consider an evolving file containing the weather forecast for seven consecutive days. If a user needs the weather forecast for the next day then any version of the file from the six last days is useful. The more recent the file is, the more accurate the requested information is. Furthermore, having access to a given file makes all previous files irrelevant to the user.

The goal of our paper is to study efficient ways for distributing evolving files within DTNs and for managing dynamically their content. The obvious way to provide the most up-to-date information is to use epidemic routing (e.g. see [10]) for each new version of F. This however consumes a lot of network resources.

We start with a general description of the model. More details will be given in the subsequent sections. There are $N+1$ mobile nodes including one source node. From now on a *node* designates any mobile node other than the source. At some time epochs the source creates an updated version of a file F. When the source meets a node it may decide to transmit to this node a copy of F. Similarly, when two nodes meet the node which carries the more recent version of F may transmit a copy of this version to the other node. When a node receives a more recent version of F than the one it was carrying (if any) it deletes at once the oldest version of F.

The setting in which only the source may transmit (a copy of) F to another node is called the *non-cooperative* setting, while in the *cooperative* setting any mobile node may transmit to any other node. We assume that transmissions are always successful.

There is a utility $U(k)$ associated with a node in state k, where the state of a node is defined as the age of the copy of F, if any, this node carries. A *file management policy*, or simply a policy, is a set of rules specifying whether the source and a node, or two nodes, should communicate whenever they meet. A policy is *static* (resp. *dynamic*) if the decision to transmit does not (resp. does) depend on the state of the mobile nodes. The objective is to find a file management policy that maximizes the expected system utility given a constraint on the expected number of communications taking place in a slot.

Section II addresses the non-cooperative setting. Time is slotted and there is a fixed probability q that any pair of mobile nodes meets in a slot. At the beginning of each slot the source creates a new version of F, so that each node carrying a copy of F knows that its state has increased by one unit. A copy of F reaching age $K+1$ ($K<\infty$) is immediately deleted. We find the optimal static policy (Proposition 1) and show that there is an optimal dynamic policy of a threshold type (Proposition 2) which we fully characterize (Proposition 3). The performance of the optimal static and dynamic policies are compared (Figures 1-4) for two different utility functions ($U(k)=1$ and $U(k)=1/k$).

Section III investigates the cooperative setting. We develop a continuous-time model in which mobile nodes meet at random times and file F is updated by the source also at random times. The latter assumption implies that nodes do not know when a new version of F is created, which in turn implies that between two consecutive meetings with the source a node does not know the age evolution of the version of F it holds. For this reason, we assume that a node never deletes a file (corresponding to $K=\infty$ in Section II) except when it receives a more recent version of F either from another node or from the source. As a result, the state of a node may grow to infinity, a situation referred to as instability. In Proposition 4 we derive conditions for stability in a Markovian framework. Under more restrictive assumptions, where node meeting times and update times are modeled by independent Poisson processes, we derive a "mean-field like" approximation for the expected number of nodes in state $k \geq 1$ in the case where a static policy is enforced. We then use this result to quantify in Figures 5-6 the benefit of having nodes to cooperate.

The deployment of optimal policies derived in Sections II-III requires that the source has a complete information on the network (node mobility, number of nodes). In Section IV we release this assumption. We focus on the noncooperative setting and restrict to static policies, and we assume that the source does not know the number of nodes N and does not know the meeting probability q. By using the theory of stochastic approximations, we construct an algorithm which converges to the optimal static policy found in Section II. Section V concludes the paper.

Remark on the notation: by convention $\sum_{k=i}^{j} \cdot = 0$ and $\prod_{k=i}^{j} \cdot = 1$ if $i > j$. $\mathbb{R}^{+}$ denotes the set of all nonnegative real numbers.

II. Non-cooperative nodes

In this section we consider the scenario where nodes do not cooperate and may only receive file F from the source. At times $t=0,1,\ldots$ the source creates a new version of file F. In the following, a slot denotes any time-period $[t,t+1)$, $t\geq 0$, and slot t stands for the time-period $[t,t+1)$. There is a probability $q>0$ that a node meets the source in a slot. We define the *meeting times* between the source and a node as the successive slots at which they meet. The meeting times of each node which the source form a sequence of independent and identically distributed (iid) random variables (rvs) and all meeting time processes are assumed to be mutually independent. For sake of simplicity we assume that all transmissions between the source and the nodes initialized in a slot are completed by the end of this slot. This is equivalent to assuming that the transmission time of F is small with respect to the duration of a slot.

When a node receives an updated version of F it deletes at once the previous version of F it was carrying, if any. We define the age of a version of F as the number of slots that have elapsed since this version was generated by the source. We assume that a version of age $K+1$ or more is useless and that a node deletes at once a file that has reached age $K+1$. Therefore, the age of a version of F varies between 1 (the version was generated in the current slot) and K (the version was generated K slots ago). We further assume that $K<\infty$ (see Remark 2.1).

The state of a node is defined as the age of the version of F it carries, if any. A node is in state 0 if it does not carry any version of F. A node in state K at the end of a slot switches to state 0 at the beginning of the next slot.

At equilibrium let $\overline{X}_k$ be the average number of nodes in state

$k = 0, 1, \ldots, K$ at the end of a slot. Note that

$$\sum_{k=0}^{K} \overline{X}_k = N. \quad (1)$$

When the source meets a node in state k with probability a_k it transmits to it the newest version of F. This decision is independent of all past decisions made by the source and is also independent of the meeting time processes. Introduce $p_k := qa_k$, $(k = 0, 1, \ldots, K)$ the probability that a node in state k receives the newest version of F in a slot.

Under the above assumptions the following equilibrium equations hold

$$\overline{X}_0 = \overline{X}_0(1 - p_0) + \overline{X}_K(1 - p_K) \quad (2)$$

$$\overline{X}_k = \overline{X}_{k-1}(1 - p_{k-1}), \quad k = 2, \ldots, K. \quad (3)$$

There is one additional equilibrium equation given by $\overline{X}_1 = p_0\overline{X}_0 + p_1\overline{X}_1 + \cdots + p_K\overline{X}_K$ which we will not consider since it can be derived by summing up all equations (2)-(3). Equations (1)-(3) define a linear system of $K + 1$ equations and $K + 1$ unknowns.

If $p_0 = 0$, namely if $a_0 = 0$ since we have assumed that $q > 0$, the existence of a unique solution to this linear system of equations will depend on the values of $p_1, \ldots, p_K$. For instance, if $p_0 = 0$ and $p_k < 1$ for all $k = 1, \ldots, K$ the solution is unique and given by $\overline{X}_0 = N$ and $\overline{X}_k = 0$ for some $k = 1, \ldots, K$. This result is of course not surprising since in this case each node will reach state 0 with a positive probability and will never leave that state afterward. On the other hand, if $p_0 = 0$ and $p_k = 1$ for all $k = 1, \ldots, K$ the steady-state will depend on the initial state, implying that the solution to the linear system will not be unique. For instance, if $p_0 = 0$ and $p_1 = 1$ then the number of nodes in state 1 in steady-state is equal to the number of nodes in that state at time $t = 0$.

From now on we will assume that $p_0 > 0$ (i.e. $a_0 > 0$) since we are only interested in the situation where the stationary regime does not depend on the initial state.

Define $\mathbf{p} := (p_0, \ldots, p_K)$. From (3) we find

$$\overline{X}_k = \overline{X}_1 \prod_{i=1}^{k-1}(1 - p_i), \quad k = 2, \ldots, K$$

so that, by (2), $\overline{X}_0 = (\overline{X}_1/p_0)\prod_{i=1}^{K}(1 - p_i)$. The above together with (1) yields

$$\overline{X}_0 = \frac{N\prod_{i=1}^{K}(1 - p_i)}{D(\mathbf{p})}, \quad \overline{X}_k = \frac{Np_0\prod_{i=1}^{k-1}(1 - p_i)}{D(\mathbf{p})} \quad (4)$$

for $k = 1, \ldots, K$, where $D(\mathbf{p}) := p_0 \sum_{j=1}^{K}\prod_{i=1}^{j-1}(1 - p_i) + \prod_{i=1}^{K}(1 - p_i))$.

In the particular case when $p_k = p$ for $k = 0, 1, \ldots, K$ then

$$\overline{X}_0 = N(1-p)^K, \quad \overline{X}_k = Np(1-p)^{k-1}, \; k = 1, \ldots, K. \quad (5)$$

Remark 2.1 ($K = \infty$): Formulas (4) hold if $K = \infty$ (i.e. nodes never delete the file they carry) provided that the denominator in (4) is finite as $K \to \infty$. This is so if $\lim_{j\to\infty} p_j > 0$ (Hint: apply d'Alembert's criterion to the infinite series $\sum_{j\geq 1}\prod_{i=1}^{j-1}(1 - p_i)$). Also note from (4) that $\overline{X}_0 = 0$ if $K = \infty$.

Remark 2.2 (Intermittently avaivalable nodes): The situation where nodes are intermittently available can be handled by replacing by $\mathbf{p}$ by $r\mathbf{p}$ in (4), with r the probability that a node is available in a slot.

A. *Performance metrics*

There are several performance metrics of interest which can be derived from (4). One of these is the expected number copies of F given by

$$\overline{X} = \sum_{k=1}^{K} \overline{X}_k = N - \overline{X}_0. \quad (6)$$

Another one is the expected age of the copies given by $(1/X)\sum_{k=1}^{K} k\overline{X}_k$. Of particular interest is to evaluate the power consumption. Since the power consumption, denoted as $Q(\mathbf{p})$ with $\mathbf{p} = (p_0, \ldots, p_K)$, is proportional to the expected number of transmissions during a slot, we will define it as

$$Q(\mathbf{p}) = \gamma\overline{X}_1. \quad (7)$$

Without loss of generality we assume from now on that $\gamma = 1$.

B. *Energy efficient file management policies*

A file management policy is any decision vector $\mathbf{a} = (a_0, \ldots, a_K) \in (0, 1] \times [0, 1]^K$, where we recall that a_k is the (conditional) probability that the source transmits F to a node in state k when it meets such a node. An equivalent definition of a file management policy is any vector $\mathbf{p} = (p_0, \ldots, p_K) \in (0, q] \times [0, q]^K$ since $p_k = qa_k$ for $k = 0, 1, \ldots, K$. Unless otherwise mentioned we will work the latter definition.

Our objective is to find an optimal file management policy $\mathbf{p}$ which maximizes the system utility given a power consumption constraint. More precisely, let $U(k)$ be the utility for having a file of age k in the system. We assume that the mapping $U : \{0, 1, \ldots, K\} \to \mathbb{R}^+$ is non-increasing. Without loss of generality we assume $U(0) = 0$. The system utility is defined as

$$C(\mathbf{p}) = \sum_{k=1}^{K} \overline{X}_k U(k).$$

If $U(k) = 1$ for all k then $C(\mathbf{p}) = \overline{X}$, given in (6).

The optimization problem is the following:

P: Maximize $C(\mathbf{p})$ over the set $(0, q] \times [0, q]^K$ given $Q(\mathbf{p}) \leq V$, where V is a positive constant.

We will solve **P** in two different settings: the *static* setting where management policies are restricted to policies of the form $\mathbf{p} = (p,\dots,p)$ with $p \in (0,q]$, and the *dynamic* setting where the optimization is made over all vectors $\mathbf{p} = (p_0,\dots,p_K) \in (0,q] \times [0,q]^K$.

1) Static optimal policy: .

In the static setting, problem $\mathbf{P}$ becomes (see (5)):

$\mathbf{P}'$: Maximize $C(p) := Np\sum_{k=1}^{K}(1-p)^{k-1}U(k)$ over $p \in (0,q]$ given that $Np \le V$.

The following result holds:

Proposition 1 (Optimal static policy):

If $Nq \le V$ then $p^\star = q$ is the optimal solution; otherwise $p^\star = V/N$ is the optimal solution or, equivalently, $p^\star = \min(q, V/N)$.

Proof. It is enough to show that the mapping $p \to C(p)$ is non-decreasing. Define $U(K+1) = 0$. We have

$$\begin{aligned} C(p) &= \sum_{k=1}^{K}(U(k)-U(k+1))\sum_{j=1}^{k}\overline{X}_j \\ &= N\sum_{k=1}^{K}(U(k)-U(k+1))(1-(1-p)^k) \end{aligned}$$

by using (5). Since the mapping U is non-increasing, $U(k) - U(k+1)$ is non-negative for all k. The proof now follows since the mapping $p \to 1-(1-p)^k$ is non-decreasing for each $k = 1,\dots,K$. ∎

2) Dynamic optimal policy: Let us introduce the new decision variables $x_k = 1 - p_k$ for $k = 1,\dots,K$ and $x_K = (1-p_K)/p_0$. Note that $1-q \le x_k \le 1$ for $k = 1,\dots,K$ and $x_K \ge (1-q)/q$ with equality if and only if $p_0 = p_K = q$. Let $\mathbf{x} = (x_1,\dots,x_K)$. Introduce the set

$$\mathbf{E} = \left\{\mathbf{x} \,:\, \mathbf{x} \in [1-q,1]^{K-1} \times [(1-q)/q,\infty)\right\}.$$

Any vector $\mathbf{x} \in \mathbf{E}$ is called a *policy*. Define the mappings

$$F(\mathbf{x}) = \sum_{k=1}^{K} U(k)\prod_{i=1}^{k-1} x_i, \quad G(\mathbf{x}) = \sum_{k=1}^{K+1}\prod_{i=1}^{k-1} x_i$$

and let $H(\mathbf{x}) := F(\mathbf{x})/G(\mathbf{x})$. Note that $F(\mathbf{x})$ does not depend on the variable x_K. We have $D(\mathbf{p}) = p_0 G(\mathbf{x})$, and so by (4)

$$C(\mathbf{p}) = NH(\mathbf{x}) \quad \text{and} \quad Q(\mathbf{p}) = N/G(\mathbf{x}).$$

In this new notation problem $\mathbf{P}$ becomes $\max_{\mathbf{x}\in\mathbf{E}} H(\mathbf{x})$ subject to the constraint $G(\mathbf{x}) \ge C$, with $C := N/V$.

An admissible policy is any policy such that $G(\mathbf{x}) \ge C$.

Definition 2.1 (Threshold policy):

A policy $\mathbf{x} = (x_1,\dots,x_K) \in \mathbf{E}$ is a threshold policy if either $x_i = 1$ or $x_{i+1} = 1-q$ for $i = 1,\dots,K-2$ and if either $x_{K-1} = 1$ or $x_K = (1-q)/q$.

Any threshold policy $\mathbf{x} = (x_1,\dots,x_K)$ is such that $x_1 \ge \ldots \ge x_{K-1}$. More precisely, it is easily seen that a threshold policy if either of **Type I** or of **Type II** with

Type I: for $k = 1,\dots,K$

$$\mathbf{x}_k(\alpha) := (1,\dots,1,\alpha,1-q,\dots,1-q,(1-q)/q) \qquad (8)$$

where $1-q \le \alpha < 1$ is the k-th entry;

Type II:

$$\mathbf{x}_K(\beta) := (1,\dots,1,\beta) \quad \text{with } \beta \ge (1-q)/q. \qquad (9)$$

In terms of the file management policy $\mathbf{p} = (p_0,\dots,p_K) \in (0,q] \times [0,q]^K$, **Type I** threshold policy $\mathbf{x}_k(\alpha)$, uniquely translates into

$$\mathbf{p}_k(\alpha) := (q,0,\dots,0,1-\alpha,q\dots,q,q) \qquad (10)$$

where $1-\alpha \in (0,q]$ is the $(k+1)$-st entry ($k = 1,\dots,K$) (as already observed $p_0 = p_K = q$ in (10) since this is the only solution of the equation $(1-p_K)/p_0 = (1-q)/q$ when $0 \le p_0, p_K \le q$ with $p_0 \ne 0$). In particular $\mathbf{p}_1(1-q) = (q,\dots,q)$.

Any file management policy

$$\mathbf{p}_K(\beta) = (p_0,0,\dots,0,p_K) \qquad (11)$$

with $(1-p_K)/p_0 := \beta$ corresponds to the unique **Type II** threshold policy $\mathbf{x}_K(\beta)$.

Proposition 2 (Optimality of threshold dynamic policy): Under the assumption that the utility function $U : \{1,\dots,K\} \to \mathbb{R}^+$ is non-increasing there exists an optimal threshold policy.

Proof. Assume that the optimal policy $\mathbf{x}$ is not a threshold policy. Hence, there exists a k, $1 \le k \le K-1$, such that either $x_k < 1$ and $x_{k+1} > 1-q$ if $k \ne K-1$ or $x_{K-1} < 1$ and $x_K > (1-q)/q$ if $k = K-1$.

Assume first that $x_1 \cdots x_{k-1} \ne 0$. Let us show that one can always find $\epsilon_k > 0$ and $\epsilon_{k+1} > 0$ such that $x'_k := x_k + \epsilon_k < 1$, $x'_{k+1} = x_{k+1} - \epsilon_{k+1} > 1-q$ if $k \ne K-1$ (resp. $x'_{k+1} = x_{k+1} - \epsilon_{k+1} > (1-q)/q$ if $k = K-1$) and $G(\mathbf{x}) = G(\mathbf{x}')$, where $\mathbf{x}' = (x_1,\dots,x_{k-1},x'_k,x'_{k+1},x_{k+2},\dots,x_K)$.

Set $\delta_k := x'_k x'_{k+1} - x_k x_{k+1} = \epsilon_k x_{k+1} - \epsilon_{k+1}x_k - \epsilon_k\epsilon_{k+1}$. The identity $G(\mathbf{x}') = G(\mathbf{x})$ is equivalent to

$$x_1 \cdots x_{k-1}\,(\epsilon_k + \delta_k A_k) = 0$$

that is $\epsilon_k + \delta_k A_k = 0$, with $A_k := 1 + x_{k+2} + x_{k+2}x_{k+3} + \cdots + x_{k+2}\cdots x_K$.

The equation $\epsilon_k + \delta_k A_k = 0$ rewrites

$$\epsilon_{k+1} = \epsilon_k \frac{1 + A_k x_{k+1}}{A_k(x_k + \epsilon_k)}.$$

So, we can find ϵ_k and ϵ_{k+1} small enough so that they satisfy the conditions.

Observe that $\epsilon_k + \delta_k A_k = 0$ with $\epsilon_k > 0$ yields $\delta_k < 0$ since $A_k > 0$.

Let us finally show that $F(\mathbf{x}') > F(\mathbf{x})$ which will contradict the optimality of $\mathbf{x}$. We have

$$\begin{aligned}\frac{F(\mathbf{x}')-F(\mathbf{x})}{x_1\cdots x_{k-1}} &= \epsilon_k U(k+1)+\delta_k[U(k+2)\\ &\quad +x_{k+2}U(k+3)+\cdots+x_{k+2}\cdots x_{K-1}U(K)]\\ &= (\epsilon_k+\delta_k A_k-\delta_k x_{k+2}\cdots x_K)U(k+1)\\ &\quad +\delta_k[U(k+2)-U(k+1)\\ &\quad +x_{k+2}(U(k+3)-U(k+1))+\cdots\\ &\quad +x_{k+2}\cdots x_{K-1}(U(K)-U(k+1))]\\ &= -\delta_k x_{k+2}\cdots x_K U(k+1)+\delta_k[U(k+2)\\ &\quad -U(k+1)+x_{k+2}(U(k+3)-U(k+1))+\cdots\\ &\quad +x_{k+2}\cdots x_{K-1}(U(K)-U(k+1))] \qquad (12)\end{aligned}$$

where we have used the identity $\epsilon_k + \delta_k A_k = 0$ to derive (12). Since U is non-increasing and $\delta_k < 0$ as noticed earlier, we deduce that the right-hand side of (12) is strictly positive, and therefore $F(\mathbf{x}') > F(\mathbf{x})$.

Assume now that $x_1\cdots x_{k-1} = 0$. This may only happen when $q = 1$ since $1-q \leq x_k \leq 1$ for $k = 1,\ldots,K$. Let $j \in \{1,\ldots,k-1\}$ be the smallest integer such that $x_j = 0$.

If the optimal policy is such that $x_j = 0$ then the value of $x_{j+1},\ldots,x_K$ are irrelevant since $x_j = 0$ implies that $X_{j+1} = \cdots = X_K = 0$ so that both the cost and the constraint will not depend on the values of $x_{j+1},\ldots,x_K$. Assume for instance that $x_{j+1} = \cdots = x_K = 0$ so that policy $\mathbf{x}$ is of the form $\mathbf{x} = (x_1,\ldots,x_{j-1},0,\ldots,0)$. It this is not a threshold policy then one can find $k' \in \{1,\ldots,j-2\}$ such that $x_{k'} < 1$ and $x_{k'+1} > 1-q = 0$. We can then duplicate the same argument used to establish (12) with k replaced by k'. Since $x_1\cdots x_{k'-1} \neq 0$ from the definition of j we conclude that $F(\mathbf{x}') > F(\mathbf{x})$. This completes the proof. ∎

It is actually possible to find the best dynamic file management policy in explicit form, as now shown.

Proposition 3 (Best dynamic file management policy):

Assume that the utility function $U : \{1,\ldots,K\} \to \mathbb{R}^+$ is non-increasing. The following results hold:

(a) if $Nq < V$ the optimal file management policy is $\mathbf{p}_1(1-q) = (q,\ldots,q)$;
(b) if $\frac{Nq}{qk+1} < V \leq \frac{Nq}{q(k-1)+1}$ for some $k = 1,\ldots,K$, the optimal file management policy is $\mathbf{p}_k(q(C-k)) = (q,0\ldots,0,1-q(C-k),q,\ldots,q)$ (see (10));
(c) if $V \leq \frac{Nq}{q(K-1)+1}$ any file management policy $\mathbf{p}_K(C-K) = (p_0,0,\ldots,0,p_K)$ such that $(1-p_K)/p_0 = C-K$ is optimal.

Proof. Since we have shown in Proposition 2 that there exists an optimal threshold policy, we only need to focus on threshold policies as defined in (8)-(9). Easy algebra show that

$$G(\mathbf{x}_k(\alpha)) = k+\frac{\alpha}{q}, \quad k=1,\ldots,K \qquad (13)$$

$$G(\mathbf{x}_K(\beta)) = K+\beta, \qquad (14)$$

so that $G(\mathbf{x}_1(\alpha_1)) \leq \cdots \leq G(\mathbf{x}_{K-1}(\alpha_{K-1})) \leq G(\mathbf{x}_K(\beta))$ for all $\alpha_1,\ldots,\alpha_{K-1} \in [1-q,1)$, $\beta \geq (1-q)/q$. From this we deduce that there are three different cases to consider (recall that $C = N/V$):

(a) $C < G(\mathbf{x}_1(1-q)) = 1/q$ or equivalently $V > Nq$;
(b) $G(\mathbf{x}_k(1-q)) \leq C < G(\mathbf{x}_{k+1}(1-q))$ or equivalently $\frac{Nq}{qk+1} < V \leq \frac{Nq}{q(k-1)+1}$;
(c) $C \geq G(\mathbf{x}_K((1-q)/q))$ or equivalently $V \leq \frac{Nq}{q(K-1)+1}$.

Case (a): In this case any threshold policy satisfies the constraint, so that the optimal policy is the policy which maximizes the cost $H(\mathbf{x})$.

It is shown in Lemma 1 in the appendix that for each $k = 1,\ldots,K$, the mapping $x_k \to H(\mathbf{x})$ is non-increasing for any $\mathbf{x} = (x_1,\ldots,x_K) \in \mathbf{E}$. Therefore, policy $\mathbf{x}_1(1-q) = (1-q,\ldots,1-q,(1-q)/q)$ is optimal, or equivalently (see (10)) the file management policy $\mathbf{p}_1(1-q) = (q,\ldots,q)$ is optimal.

Case (b): Assume that $G(\mathbf{x}_k(1-q)) \leq C < G(\mathbf{x}_{k+1}(1-q))$ for some $1 \leq k \leq K-1$.

By Lemma 1 in the appendix we see that the best threshold policy is the one which saturates the constraint, namely policy $\mathbf{x}_k(\alpha)$ such that $G(\mathbf{x}_k(\alpha)) = C$, that is $\alpha = q(C-k)$. By (13) this policy is unique and is given by $\mathbf{x}_k(q(C-k))$. Equivalently (see (10)), the optimal file management policy is $\mathbf{p}_k(q(C-k))$.

Case (c): In this case there is no **Type I** policy which satisfies the constraint $G(\mathbf{x}) \geq C$. Among all **Type II** policies satisfying this constraint the one with the smallest K-th entry is the policy such that $G(\mathbf{x}_K(\beta)) = C$, that is (see (11)) policy $\mathbf{x}_K((C-K)) = (1,\ldots,1,C-K)$. We conclude again from Lemma 1 that this is the optimal policy. Equivalently (see (11)), any file management policy $\mathbf{p}_K(C-K) = (p_0,0,\ldots,0,p_K)$ such that $(1-p_K)/p_0 = C-K$ is optimal. This concludes the proof. ∎

C. Numerical results

Let p_s^* (resp. $\mathbf{p}_d^*$) be the static (resp. dynamic) file management policy which solves the optimization problem $\mathbf{P}$ – as found in Proposition 1 (resp. Proposition 3). Figures (1)-(4) display mappings $q \to \sum_{k=1}^K U(k)\overline{X}_k$ under policies p_s^* and $\mathbf{p}_d^*$ (corresponding curves are referred to as "static" and "dynamic", respectively), for two different utility functions ($U(k) = 1$, $U(k) = 1/k$) and for two different values of the constraint V ($V = 10, 20$). In all figures $N = 100$ and $K = 5$. These results show that the use of the optimal dynamic policy may yield substantial gains (e.g. for $U(k) = 1$ gain of $\approx 22\%$ for all $q \geq 0.2$ – see Fig. 2; gain of $\approx 45\%$ for q close to 1 – see Fig. 1. Gain is halved for $U(k) = 1/k$.). The gain is an increasing function of the meeting probability q.

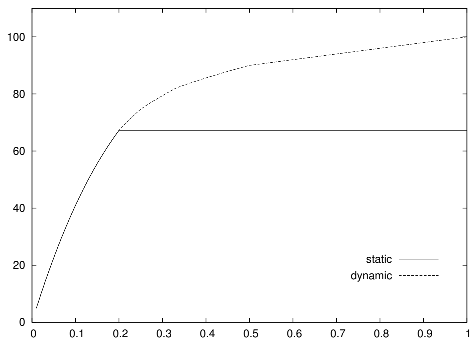

Fig. 1. $q \rightarrow \sum_{k=1}^{5} \overline{X}_k$ under optimal static/dynamic policy: V=20 (N=100, K=5)

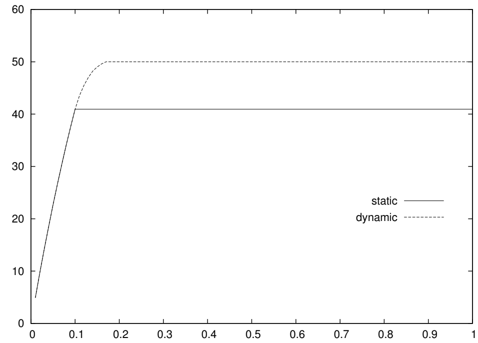

Fig. 2. $q \rightarrow \sum_{k=1}^{5} \overline{X}_k$ under optimal static/dynamic policy: V=10 (N=100, K=5)

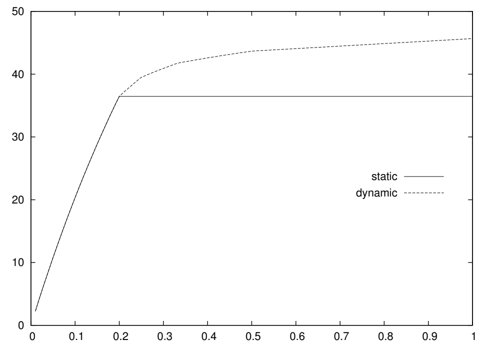

Fig. 3. $q \rightarrow \sum_{k=1}^{5} \overline{X}_k/k$ under optimal static/dynamic policy: V=20 (N=100, K=5)

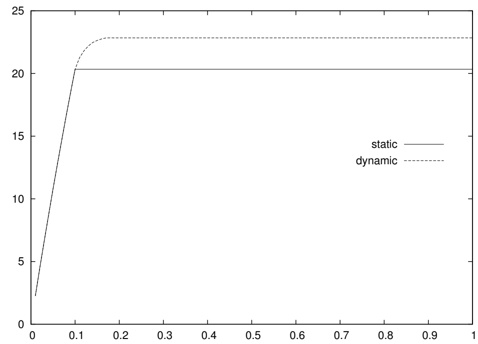

Fig. 4. $q \rightarrow \sum_{k=1}^{5} \overline{X}_k/k$ under optimal static/dynamic policy: V=10 (N=100, K=5)

III. Cooperative nodes

In this section we assume that nodes cooperate in the sense that when two nodes meet the one with the most recent version of F may send a copy to the other one. A node may only delete the version of F it carries when it receives a more recent version from the source or from another node.

The identity of the source is 0 and nodes are labeled $1, 2, \ldots, N$. We observe the system at discrete times $\{t_n\}_{n\geq 0}$, where t_n is the time of the nth event. An *event* is either the meeting of the source with a node, the meeting of two nodes or the creation of a new version of F by the source. Let $\{\xi_n^{i,j}\}_n$ and $\{\zeta_n\}_n$ be $\{0,1\}$-valued rvs where $\xi_n^{i,j} = 1$, $j \neq 0$, if node i meets node j at time t_n, $\xi_n^{i,0} = 1$ if node i meets the source at time t_n, and $\zeta_n = 1$ if the source creates a new version of F at time t_n. We assume that $\zeta_n + \sum_{i,j} \xi_n^{i,j} = 1$ for all n (only one event at time t_n).

Let Y_n^i be the age of the version of F that node i carries just before time t_n. We assume that $Y_0^i \geq 1$ for all i. We introduce the additional $\{0,1\}$-valued rvs $\{a_n^{i,j}(k,l)\}$ and $\{a_n^i(k)\}$, where $a_n^{i,j}(k,l) = 1$ if node i in state k receives a copy of F from node j in state $l < k$ if they meet at t_n, and $a_n^i(k) = 1$ if the source transmits the latest version of F to node i in state k if they meet at t_n.

Denote $\theta_{i,j}(k,l) = P(a_n^{i,j}(k,l) = 1)$ and $\theta_i(k) = P(a_n^i(k) = 1)$.

The following recursions hold ($i = 1, \ldots, N$):

$$
\begin{aligned}
Y_{n+1}^i &= Y_n^i + (1 - Y_n^i)\xi_n^{i,0} a_n^i(Y_n^i) \\
&+ \sum_{\substack{j=1 \\ j\neq i}}^{N} (Y_n^j - Y_n^i)\mathbf{1}_{Y_n^j - Y_n^i < 0}\, \xi_n^{i,j} a_n^{i,j}(Y_n^i, Y_n^j) + \zeta_n.
\end{aligned} \tag{15}
$$

Define the vectors $Y_n = (Y_n^1, \ldots, Y_n^N) \in \mathcal{E} := \{1, 2, \ldots\}^N$, $Z_n := (\{\xi_n^{i,j}\}), \zeta_n)$.

Assumptions A1:

(1) $\{Z_n\}_n$ is an iid sequence of rvs. Define $q_i := P(\xi_n^{i,0} = 1)$, $q_{i,j} := P(\xi_n^{i,j} = 1)$, and $r := P(\zeta_n = 1)$;
(2) $r > 0$, $q_i > 0$, $q_{i,j} > 0$ for all $i \neq j$;
(3) the probability that two nodes communicate when they meet only depends on their identity and state, namely $P(a_n^{i,j}(Y_n^i, Y_n^j) = 1 | \{Y_m, Z_m\}_{m\leq n}) = \theta_{i,j}(Y_n^i, Y_n^j)$ for all $i \neq j$;
(4) the probability that the source communicate with another node when they meet only depends on the node identity and state, that is, $P(a_n^i(Y_n^i) = 1\,|\,\{Y_m, Z_m\}_{m\leq n}) = \theta_i(Y_n^i)$ for all i.

A. Stability

Proposition 4 (Stability of $\{Y_n\}_n$):

Assume that **A1** holds. Then, $\{Y_n\}_n$ is an homogeneous, irreducible and aperiodic Markov chain on $\mathcal{E}$. It is positive recurrent if there exist an integer M_0 and $\theta > 0$ such that $\theta_i(k) \geq \theta$ for all $k \geq M_0$, $i = 1, \ldots, N$.

Proof. Only the positive recurrence property does not trivially follow from **A1**. We will show it by applying Foster's criterion (see e.g. [7]) to $\{Y_n\}_n$. Consider the Lyapounov function $f : \mathcal{E} \to \mathbb{R}^+$ defined by $f(\mathbf{y}) = \sum_{i=1}^N y_i$ with $\mathbf{y} = (y_1, \ldots, y_N)$. We need to show that there exists a finite set $\mathcal{F} \subset \mathcal{E}$ such that $\Delta(\mathbf{y}) := E[f(Y_{n+1}) - f(Y_n) \,|\, Y_n = \mathbf{y}]$ is finite on $\mathcal{F}$ and that there exists $\epsilon > 0$ with $\Delta(\mathbf{y}) \leq -\epsilon$ for $\mathbf{y} \in \mathcal{E} - \mathcal{F}$.

Take $y_i \geq \max(2, M_0)$, $i = 1, \ldots, N$. We have

$$\begin{aligned}\Delta(\mathbf{y}) &= \sum_{i=1}^N (1 - y_i) q_i \theta_i(y_i) \\ &\quad + \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N (y_j - y_i)\, \mathbf{1}_{y_j < y_i}\, q_{i,j} \theta_{i,j}(y_i, y_j) + Nr \\ &\leq \theta \sum_{i=1}^N (1 - y_i) q_i + Nr. \end{aligned} \quad (16)$$

Fix $\epsilon > 0$. One can always find an integer $M_1 \geq \max(2, M_0)$ such that the r.h.s. of (16) is less than $-\epsilon$ as long as $y_i \geq M_1$ for any $i = 1, \ldots, N$. Hence, Foster's criterion applies with $\mathcal{F} := \{\mathbf{y} \in \mathcal{E} : y_i \leq M_1 - 1\}$ since $\Delta(\mathbf{y})$ is finite on $\mathcal{F}$ and is less than a negative constant on $\mathcal{E} - \mathcal{F}$. ∎

We will show in a companion paper that the stability of $\{Y_n\}_n$ can be investigated in a much more general framework than the Markovian framework.

B. Quantitative performance

We make additional assumptions in order to compute $\overline{X}_k$, the expected number of files of age $k \geq$ in steady-state. We assume that the source and node $i = 1, \ldots, N$ (resp. any pair of nodes i and j, $i \neq j$) meet according to a Poisson process with rate $\lambda > 0$ and that the source creates a new version of F at each occurrence of a Poisson process with rate $\mu > 0$. These $N(N+1)/2 + 1$ Poisson processes are assumed to be mutually independent. We further assume that $\theta_i(k) := a_k > 0$ and $\theta_{i,j}(k, l) := b_{k,l}$ for any i, j, k, l. In other words, when two nodes (i.e. source or nodes) meet the probability that a transmission occurs only depends on the node state and not on their identity. By Proposition 4 we observe that the system is stable (in this setting $q_i = q_{i,j} = \lambda/\nu$ and $r = \mu/\nu$ with $\nu := \lambda N(N+1)/2 + \mu$).

Let $X_k(t)$ be number of nodes in state k at time t. Set $\overline{X}_k(t) := E[X_k(t)]$. We have the Kolmogorov equations

$$\begin{aligned}\frac{d\overline{X}_1(t)}{dt} &= -\mu \overline{X}_1(t) + \lambda \sum_{k \geq 2} a_k \overline{X}_k(t) \\ &\quad + \lambda \sum_{l \geq 2} b_{1,l} E[X_1(t) X_l(t)] \end{aligned} \quad (17)$$

$$\begin{aligned}\frac{d\overline{X}_k(t)}{dt} &= \mu \overline{X}_{k-1}(t) + \lambda \sum_{l \geq k+1} b_{k,l} E[X_k(t) X_l(t)] \\ &\quad - \lambda \sum_{l=1}^{k-1} b_{k,l} E[X_k(t) X_l(t)] \\ &\quad - (\lambda a_k + \mu) \overline{X}_k(t), \quad k \geq 2. \end{aligned} \quad (18)$$

Let $X_k := \lim_{t \to \infty} X_k(t)$ (a.s.) and $\overline{X}_k = E[X_k]$. From (17)-(18) we find

$$\mu \overline{X}_1 = \lambda \sum_{k \geq 2} a_k \overline{X}_k + \lambda \sum_{l \geq 2} b_{1,l} E[X_1 X_l] \quad (19)$$

$$\begin{aligned}\mu \overline{X}_{k-1} &+ \lambda \sum_{l \geq k+1} b_{k,l} E[X_k X_l] \\ &= \lambda \sum_{l=1}^{k-1} b_{k,l} E[X_k X_l] + (\lambda a_k + \mu) \overline{X}_k, \quad k \geq 2. \end{aligned} \quad (20)$$

We will consider two cases.

Case (a): $b_{k,l} = 0$ for all k, l. This corresponds to the non-cooperative setting studied in Section II. We find (Hint: use $\sum_{k \geq 1} X_k = N$)

$$\overline{X}_k = \frac{N \prod_{j=2}^k \frac{\mu}{\mu + \lambda a_j}}{\sum_{j \geq 1} \prod_{i=2}^j \frac{\mu}{\mu + \lambda a_i}}, \quad k \geq 1 \quad (21)$$

If we perform the change of variable $\mu/(\mu + \lambda a_i) = 1 - p_{i-1}$ in (21) we retrieve the corresponding results (4) found in the discrete-time setting with $K = \infty$ (see Remark 2.1), thereby showing that this model is the continuous-time analog of the discrete-time model.

Case (b): $a_k = a > 0$ and $b_{k,l} = b > 0$ for all k, l. Because of the terms $E[X_k X_l]$ equations (19)-(20) cannot be solved. To solve them we will assume that $\mathrm{cov}(X_k, X_l)$ is negligible for $k \neq l$ so that $E[X_k X_l] \approx \overline{X}_k \overline{X}_l$. We conjecture that this approximation (referred to as the "mean-field approximation" – see e.g. [1]) is accurate for large N (the mean-field approach in [6, Theorem 3.1] does not apply here and cannot therefore be used to validate these approximations). With this approximation and the use of the identity $\sum_{k \geq 1} \overline{X}_k = N$, equations (19)-(20) become (with $\rho := \lambda/\mu$)

$$b\overline{X}_1^2 - \overline{X}_1 (bN - a - 1/\rho) - aN = 0 \quad (22)$$

$$b\overline{X}_1^2 - \overline{X}_1 \left(bN - a - 1/\rho - 2b \sum_{l=1}^{k-1} \overline{X}_l \right) + \overline{X}_{k-1}/\rho = 0 \quad (23)$$

for $k \geq 2$. The unique nonnegative root of (22) is

$$\overline{X}_1 = \left(D_1 + \sqrt{D_1^2 + 4abN} \right) / 2b \quad (24)$$

while for $k \geq 2$ we get from (23)

$$\overline{X}_k = \left(D_k + \sqrt{D_k^2 + 4b\overline{X}_{k-1}/\rho} \right) / 2b \quad (25)$$

with $D_k := bN - a - 1/\rho - 2b \sum_{l=1}^{k-1} \overline{X}_l$. (24)-(25) define a recursive scheme allowing the computation of $\overline{X}_k$ for any k.

C. Numerical results

We want to quantify the impact of node cooperation on the system performance in the case where the source has limited power resources. We want to optimize the system utility $\sum_{k=1}^{K} U(k)\overline{X}_k$ under a constraint on the expected number of transmissions *by the source* between the creation of two consecutive version of F. To this end, we will assume that the source transmits to any node that it meets with the probability $a = a^\star$, where $a^\star := \min(1, (1+\rho)V/N\rho)$ is the static policy that solves problem **P** (Proposition 1). Let q_ρ be the probability that the source meets a given node between two creations of a new version of F. We have $q_\rho = \lambda/(\lambda+\mu) = \rho/(1+\rho)$ thanks to the Poisson assumptions. In all experiments reported below we set $N = 100$, $V = 20$ and $K = 5$. Figures 5-6 display the mapping $q_\rho \to \sum_{k=1}^{K} U(k)\overline{X}_k$ (with $U(k) = 1$ in Fig. (5) and $U(k) = 1/k$ in Fig. 6) for three values of the probability b. The value $b = 0$ corresponds to the non-cooperative setting (case (i); curve referred to as "noncooperative") and the values $b = 0.05$, $b = 0.1$ correspond to the cooperative setting (case (ii); curves referred to as "b=0.05" and "b=0.1"). One (obvious) conclusion is that the cooperative setting outperforms the noncooperative setting. Another conclusion is that the impact of b is more pronounced when $U(k) = 1/k$.

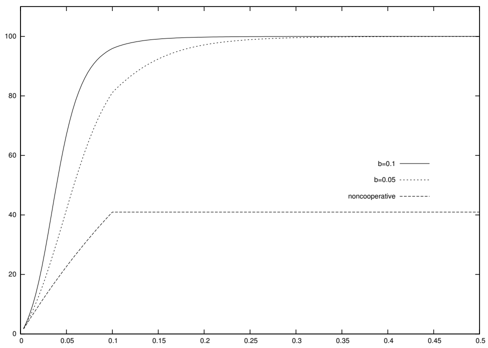

Fig. 5. $q_\rho \to \sum_{k=1}^{5} \overline{X}_k$ ($a = a^\star$, $b \in \{0, 0.05, 0.1\}$, $N = 100$, $V = 10$, $K = 5$)

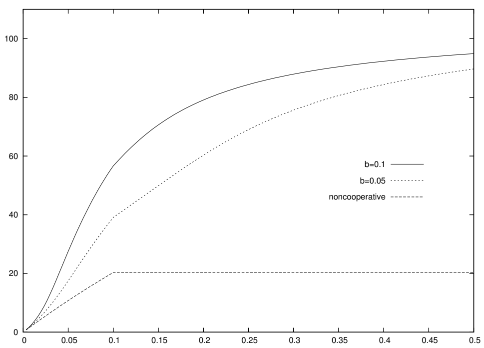

Fig. 6. $q_\rho \to \sum_{k=1}^{5} \overline{X}_k/k$ ($a = a^\star$, $b \in \{0, 0.05, 0.1\}$, $N = 100$, $V = 10$, $K = 5$)

IV. Imperfect state information

In this section we consider the static setting of Section II where nodes do not cooperate. We assume that the source does not know parameters N and q, so that it cannot compute $a^\star := \min(1, V/Nq)$, the (conditional) transmission probability that solves problem **P** (cf. Proposition 1). Instead, we will assume that every $M \geq 1$ slots the source updates the transmission probability a, where M is an arbitrary integer. More precisely, let θ_m be the transmission probability used in slots $mM, \ldots, (m+1)M - 1$. Define the projection operator

$$\Pi_H(u) = \begin{cases} 1 & \text{if } u > 1 \\ u & \text{if } 0 \leq u \leq 1 \\ 0 & \text{if } u < 0. \end{cases}$$

Consider the stochastic recursion

$$\theta_{m+1} = \Pi_H\Big(\theta_m + \epsilon_m(MV - Y_m)\Big) \tag{26}$$

where Y_m is the total number of transmissions in slots $mM, \ldots, (m+1)M - 1$, and $\{\epsilon_m\}_m$ are nonnegative real numbers satisfying

$$\sum_{m\geq 0} \epsilon_m^2 < \infty, \quad \sum_{m\geq 0} \epsilon_m = \infty, \tag{27}$$

Observe that the source knows Y_m for every m. Recursion (26) is motivated by the fact that $a^\star$ is the unique zero of $h(a) := V - \overline{X}_1$ if $h(1) > 0$ and $a^\star = 1$ otherwise, so that the source target is to find the zero, if any, of $h(a)$ (or, equivalently, the zero of $Mh(a)$) in [0,1].

Proposition 5 (Stochastic approximation algorithm):

As $m \to \infty$, θ_m in (26) converges with probability one to $a^\star$, the optimal static policy of Section II-B.1.

Proof. The proof directly follows from the remark after Theorem 2.1 in [8, p. 127]. Let us briefly checked that conditions (A2.1)-(A2.5) of Theorem 2.1 hold. Since $0 \leq Y_m \leq MN$ for all m, condition (A2.1) holds (this condition requires that $\sup_m E|Y_m|^2 < \infty$). By an inductive argument applied to (26) we see that $E[Y_m|\theta_0, Y_i, i < m] = E[Y_m|\theta_m, \theta_i, Y_i, i < m]$. We then note that $E[Y_m|\theta_m, \theta_i, Y_i, i < m] = E[Y_m|\theta_m] := g(\theta_m)$ since the decision by the source to transmit a copy of F to a node only depends on the enforced transmission probability. This implies that condition (A2.2) holds (condition (A2.2) in [8, p. 126] states that $E[Y_m|\theta_0, Y_i, i < m]$ has the form of $g(\theta_m) + \beta_m$ where β_n is a r.v.). We have

$$g(x) = M(V - Nqx)$$

so that conditions (A2.3) (g is continuous) and (A2.5) ($\sum_{m\geq 0} \epsilon_m|\beta_m| < \infty$ w.p.1) are satisfied. Last, condition (A2.4) ($\sum_{m\geq 0} \epsilon_n^2 < \infty$) holds from (27).

Consider the ODE $dx(t)/dt = g(x(t))$. Its solution is $x(t) = (x(0) - V/Nq)e^{-MNqt} + V/Nq$. It has a unique equilibrium point, given by $x_0 = V/Nq$, which is asymptotically stable in the sense of Lyapounov [8, p. 104] (i.e. for each $\delta > 0$, there exists $\eta > 0$ such that if $|x(0) - x_0| < \eta$ then $|x(t) - x_0| < \epsilon$

for all $t \geq 0$). By [8, Remark p. 127] we conclude that $\{\theta_m\}_m$ converges with probability one to $\min(1, V/Nq)$. ∎

Figure 7 below provides a numerical illustration of the convergence of algorithm (26) to the optimal policy $a^\star$ for $M = 1$, $N = 100$, $V = 10$ and $q = 0.2$. In this case $a^\star = 0.5$.

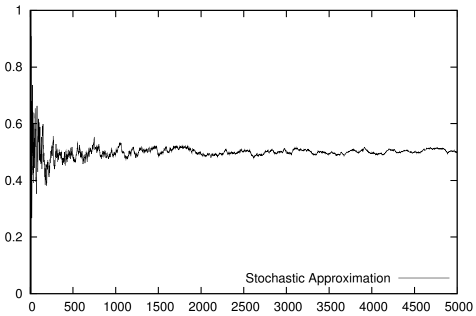

Fig. 7. $m \to \theta_m$: $M = 1$, $a^\star = 0.5$ ($N = 100$, $V = 10$, $q = 0.2$)

V. Conclusion

We have developed simple stochastic models for evaluating the performance of file management policies in DTNs storing dynamic files. Both static and dynamic policies have been investigated. We have shown that using dynamic policies instead of static policies yields substantial gain in the performance; this result holds both in the non-cooperative setting, where only the source is allowed to communicate with the other nodes, and in the cooperative setting where all pairwise communications are possible. Future works include the study of multi-source and multi-file scenarii.

VI. Acknowlegements

The work of E. Altman and P. Nain was partially supported by europeean project IP BIONETS and by Network of Excellence EuroNF. The work of J-C. Bermond was partially supported by europeean project FET AEOLUS and by CRC CORSO with France Telecom.

Appendix

Lemma 1 (Monotonicity of $H(\mathbf{x})$):

For each $k = 1, \ldots, K$, the mapping $x_k \to H(\mathbf{x})$ is non-increasing for any $\mathbf{x} = (x_1, \ldots, x_K) \in \mathbf{E}$.

Proof. First, notice that the mapping $x_K \to H(\mathbf{x})$ is clearly non-increasing since x_K only appears in $G(\mathbf{x})$, the denominator of $H(\mathbf{x})$, and since $G(\mathbf{x})$ is non-decreasing in x_K.

Assume now that $k = 1, \ldots, K$. Let

$$\begin{aligned} B(j) &:= 1 + x_1 + x_1 x_2 + \cdots + x_1 \cdots x_j \\ B_k(j) &:= 1 + x_{k+1} + x_{k+1} x_{k+2} + \cdots + x_{k+1} \cdots x_j \end{aligned}$$

with $B(0) = 1$, $B_k(k) = 1$. Set $U(K+1) = 0$. We have

$$F(\mathbf{x}) = \sum_{j=1}^{K} [U(j) - U(j+1)] B(j-1), \quad G(\mathbf{x}) = B(K)$$

so that

$$\begin{aligned} \frac{\partial}{\partial x_k} F(\mathbf{x}) &= \prod_{j=1}^{k-1} x_j \sum_{j=k+1}^{K} [U(j) - U(j+1)] B_k(j-1) \\ \frac{\partial}{\partial x_k} G(\mathbf{x}) &= B_k(K) \prod_{j=1}^{k-1} x_j . \end{aligned}$$

Therefore

$$\begin{aligned} \frac{\partial}{\partial x_k} H(\mathbf{x}) = \frac{(\prod_{j=1}^{k-1} x_j)^2}{G(\mathbf{x})^2} \Bigg(& \sum_{j=1}^{k} [U(j+1) - U(j)] \\ & \times B(j-1) B_k(K) + \sum_{j=k+1}^{K} [U(j+1) - U(j)] \\ & \times [B(j-1) B_k(K) - B_k(j-1) B(K)] \Bigg). \end{aligned}$$

The first summation is non-positive since U is non-increasing and since $B(j-1) B_k(K) \geq 0$ for all $\mathbf{x} \in \mathbf{E}$. Using again the decreasingness of U a sufficient condition for the second summation to be non-positive is that coefficients $B(j-1) B_k(K) - B_k(j-1) B(K)$ are all non-negative. To see that this is indeed true, note that $B(j) = B(k-1) + x_1 \ldots, x_k B_k(j)$ so that $B(j-1) B_k(K) - B_k(j-1) B(K) = B(k-1)[B_k(K) - B_k(j-1)]$ which is non-negative for all $\mathbf{x} \in \mathbf{E}$. This concludes the proof. ∎

References

[1] M. Benaïm and J-Y. Le Boudec, "A Class of Mean Field Interaction Models for Computer and Communication Systems." Technical Report LCA-Report-2008-010, April 2008. To appear in *Performance Evaluation*.

[2] R. Groenevelt, P. Nain and G. Koole, "Message Delay in Mobile Ad Hoc Networks," Proc. of PERFORMANCE 2005, Juan-les-Pins, France, October 3-7, 2005. In: *Performance Evaluation*, Vol. 62, No. 1-4, pp. 210-228, October 2005.

[3] M. Grossglauser and D.N. Tse, "Mobility Increases the Capacity of Ad Hoc Wireless Networks," *IEEE/ACM Transactions on Networking*, Vol. 10, No. 4, pp. 477-486, August 2002.

[4] Z.J. Haas and T. Small, "A New Networking Model for Biological Applications of Ad Hoc Sensing Networks," *IEEE/ACM Transactions on Networking*, Vol. 14 , No. 1, pp. 27-40, February 2006.

[5] B. Hammersley, *Content Syndication with RSSSharing Headlines and Information.* O'Reilly, 2003.

[6] T.G. Kurtz, "Solutions of Ordinary Differential Equations as Limits of Pure Jump Markov processes," *J. Appl. Prob.*, **7**, pp. 49-58, 1970.

[7] S. Meyn and R.I. Tweedie, *Markov Chains and Stochastic Stability.* Springer Verlag, London, 1993.

[8] H.J. Kushner and G.G. Yin, *Stochastic Approximation and Recursive Algorithms and Applications.* Application of Mathematics, Stochastic Modelling and Applied Probability, Springer Verlag, vol. 35, 2nd Edition, 2003.

[9] G. Sharma, R.R. Mazumdar and N.B. Shroff, "Delay and Capacity Trade-offs in Mobile Ad Hoc Networks: A Global Perspective," *IEEE/ACM Transactions on Networking*, Vol. 15, No. 1, pp. 981-992, 2007.

[10] X. Zhang, G. Neglia, J. Kurose and D. Towsley, "Performance Modeling of Epidemic Routing," *Elsevier Computer Networks*, Vol. 51, No. 10, pp. 2867-2891, July 2007.