



**HAL**  
open science

# Distributed Sub-gradient Method for Delay Tolerant Networks

Riccardo Masiero, Giovanni Neglia

► **To cite this version:**

Riccardo Masiero, Giovanni Neglia. Distributed Sub-gradient Method for Delay Tolerant Networks. [Research Report] 2010. inria-00506485v1

**HAL Id: inria-00506485**

**<https://inria.hal.science/inria-00506485v1>**

Submitted on 27 Jul 2010 (v1), last revised 24 Aug 2010 (v3)

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

# *Distributed Subgradient Method for Delay Tolerant Networks*

Riccardo Masiero — Giovanni Neglia

N° 7345

July 2010

— Networks and Telecommunications —

A large, light gray, stylized letter 'R' is positioned to the left of the text.

*R*apport  
de recherche



## Distributed Subgradient Method for Delay Tolerant Networks

Riccardo Masiero<sup>\*†</sup>, Giovanni Neglia<sup>†</sup>

Theme : Networks and Telecommunications  
Networks, Systems and Services, Distributed Computing  
Équipe-Projet MAESTRO

Rapport de recherche n° 7345 — July 2010 — 35 pages

**Abstract:** In this report we consider that nodes in a Delay Tolerant Network (DTN) may collaborate to minimize the sum of local objective functions, depending in general on some parameters or actions of all the nodes in the network. If the local objective functions are convex, it can be adopted a recently proposed computation framework, that relies on local sub-gradient methods and consensus algorithms to average each node information. Existing convergence results for this framework can be applied to DTNs only in the case of nodes' synchronous operation and simple mobility models without memory. We address both these issues. First, we prove convergence to the optimal solution for a more general class of mobility processes. Second, we show that, under asynchronous operation, a straight application of the original method would lead to sub-optimal solutions and we propose some changes to solve this problem. As a particular case study, we show how the framework can be applied to optimize the dissemination of dynamic content in a DTN.

**Key-words:** delay tolerant networks, distributed optimization, consensus, sub-gradient

\* Department of Information Engineering, University of Padova 6/B, Via G. Gradenigo, I-35131 Padova (PD), Italy. E-mail: [riccardo.masiero@dei.unipd.it](mailto:riccardo.masiero@dei.unipd.it).

† INRIA-MAESTRO, 2004 route des Lucioles, BP 93, FR-06902 Sophia Antipolis (BIOT), France. E-mails: {[riccardo.masiero](mailto:riccardo.masiero@sophia.inria.fr), [giovanni.neglia](mailto:giovanni.neglia@sophia.inria.fr)}@sophia.inria.fr.

## Méthode de sous-gradient distribuée dans les réseaux tolérant les délais

**Résumé :** Dans ce rapport, nous considérons que les nœuds dans un réseau tolérant les délais (Delay Tolerant Network, DTN) peuvent collaborer afin de minimiser la somme des fonctions objective locales, qui dépendent en général des paramètres ou des actions de tous les nœuds du réseau. Si les fonctions objectif locales sont convexes, il peut être adopté une méthodologie récemment proposé, qui s'appuie sur le calcul du sous-gradient de la fonction locale et les algorithmes de consensus pour faire la moyenne de l'information de chaque nœud. Les résultats de convergence existants pour cette méthodologie peuvent être appliqué aux DTNs uniquement dans le cas de operation synchrone des nœuds et pour des modèles de mobilité simples, sans mémoire. Nous abordons ces deux questions. Tout d'abord, nous prouvons la convergence de la méthode à la solution optimale pour une classe plus générale des processus de mobilité. Deuxièmement, nous montrons que, dans le cas de opération asynchrone, une application directe de la méthode originale conduit à des solutions sous-optimales et nous proposons quelques modifications pour résoudre ce problème. Comme étude de cas particulier, nous montrons comment le cadre peut être utilisés pour optimiser la diffusion de contenus dynamiques dans un DTN.

**Mots-clés :** réseau tolérant les délais, optimisation distribuée, consensus, sous-gradient

## 1 Introduction

In [1], Nedić and Ozdaglar propose a solution for distributed optimization in a network of  $M$  agents (nodes) based on the pioneer work of Tsitsiklis, [2]. In doing so, algorithms based on the consensus paradigm are used in conjunction with the sub-gradient method; throughout this report we will refer to this approach as *distributed sub-gradient method*. In the original approach proposed by [2] and also adopted in [1], all the  $M$  agents in the network are required to exchange information each other within a time  $k \leq B$  to prove the convergence of the proposed algorithm. Whilst this assumption is reasonable for the applications envisioned by Tsitsiklis, e.g., machine parallel computation, is instead likely to be not met in practice in mobile network scenarios.

Our interest is to focus onto a framework where mobile nodes are allowed to exchange information only when meeting among agents occurs, and therefore there is a non null probability that two nodes never encounter; also we would like to require as less synchronization as possible among agents. Very recent progresses in the field, due to the work of Lobel and Ozdaglar [3] as well as the work of Tahbaz-Salehi and Jadbabaie [4], allow us to have convergence results for the above framework also in the case of mobile and probabilistic scenarios like the one described below.

Let us consider a network of  $M$  mobile nodes or agents. Each node  $i$  in the network, has a state vector  $\mathbf{x}^i \in \mathbb{R}^N$  of parameters (or in general, quantities) to be chosen according to some criteria that will be better specify further on. Let us assume also that there is a vector  $\mathbf{x}^*$  such that, if chosen by all the nodes, allow the overall network to reach the maximum gain as regard as some performance metric or function utility. To make the node agree on the , the nodes in the network are let free to communicate (with no delay) when a transmission opportunity occurs, i.e., when two nodes are within the communication range of each other. If all the agents in the network eventually

In the above context we explore a stochastic model for describing the iteration among nodes and proving the desired convergence results.

Furthermore, we will propose a possible interesting scenario of application of the studied framework, like the one provided by [5]. In [5] a Server Provider (SP) optimally decides to which node it should transmit (for a content sharing application) on the basis of some utility function depending on both the meetings occurring in the network and the chosen vector of update rates. We can think of proposing a dual approach to this problem, particularly suitable for Delay Tolerant Network (DTN, see e.g., [6] and [7]) applications; in the proposed application each node in the network would solely decide if requiring information from the BS or not, according to its own utility function and the exchange of information occurred with its encountered neighbors.

This report is structured as follows. In Section 2 we review the main results in [1, 3] on convergence and optimality of the distributed sub-gradient method and in Section 3 we present the motivation of our work. As original contributions, in Section 4 we extend the results in [1] and [3] to more general models of network mobility and in Section 5, we study how the presented framework needs to be extended to cope with asynchronous node operations. Then, in Section 6, we illustrate a possible DTN application exploiting the distributed sub-gradient method. Finally, Section 8 concludes the report and trace the line for the future research.

## 2 Distributed Sub-gradient Method's Overview

In this Section we review the main results in [1, 3] on convergence and optimality of the distributed sub-gradient method when a random network scenario is considered.

Let us consider a set of  $M$  nodes (agents), that want to cooperatively solve the following optimization problem:

**Problem 1** (Global Optimization Problem). *Given  $M$  convex functions  $f_i(\mathbf{x}) : \mathbb{R}^N \rightarrow \mathbb{R}$ , determine:*

$$\mathbf{x}^* \in \operatorname{argmin}_{\mathbf{x} \in \mathbb{R}^N} f(\mathbf{x}) = \sum_{i=1}^M f_i(\mathbf{x}),$$

that we assume to be feasible. The difficulty of the task arises from the fact that agent  $i$ , for  $i = 1, 2, \dots, M$ , only knows the corresponding function  $f_i(\mathbf{x})$ , that is called the *local objective function*. For example  $f_i$  could be a performance metric relative to node  $i$ , and  $f$  could indicate global network performance.

If the functions  $f_i$  are differentiable, each node could apply a gradient method to its function  $f_i$  to generate a sequence of local estimates, but this would lead to  $M$  biased estimates of the solution of problem 1. In [1] and [3], it is shown that if nodes perform a gradient method but are also able to *average* their local estimates, under opportune conditions, these estimates all converge to a point of minimum of  $f$ ,  $\mathbf{x}^*$ .

In particular, a time slotted system is assumed, where, at the end of a slot, each node  $i$  communicates its local estimate to a subset of all the other nodes, and then updates the estimate according to the following equation<sup>1</sup>:

$$\mathbf{x}^i(k+1) = \sum_{j=1}^M a_{ij}(k) \mathbf{x}^j(k) - \gamma(k) \mathbf{d}^i(k), \quad (1)$$

where the vector  $\mathbf{d}^i(k) \in \mathbb{R}^M$  is a sub-gradient<sup>2</sup> of agent  $i$  objective function  $f_i(\mathbf{x})$  computed at  $\mathbf{x} = \mathbf{x}^i(k)$ , the scalar  $\gamma(k) > 0$  is the step-size of the sub-gradient algorithm at iteration  $k$ , and  $a_{ij}(k)$  are non-negative weights, such that  $a_{ij}(k) > 0$  if and only if node  $i$  has received node  $j$ 's estimate at the step  $k$  and  $\sum_{j=1}^M a_{ij}(k) = 1$ . We denote by  $\mathbf{A}(k)$  the matrix whose elements are the weights, i.e.  $[\mathbf{A}(k)]_{ij} = a_{ij}(k)$ .

We observe that the first addend in the right side of 1 corresponds to averaging according to a consensus algorithm.

[1] proves that the iterations (1) generate sequences converging to a minimum of  $f$  under the following set of conditions:

1. the step-size  $\gamma(k)$  is such that  $\sum_{i=1}^{\infty} \gamma(k) = \infty$  and  $\sum_{i=1}^{\infty} \gamma(k)^2 < \infty$ ;
2. the gradient of each function  $f_i$  is bounded;
3. each matrix  $\mathbf{A}(k)$  is symmetric (then doubly stochastic);

<sup>1</sup>In this paper all the real valued vectors are assumed to be column vectors.

<sup>2</sup> $\mathbf{d}^i \in \mathbb{R}^N$  is a sub-gradient of the function  $f_i$  at  $\mathbf{x}^i \in \operatorname{dom}(f_i)$  iff  $f_i(\mathbf{x}^i) + (\mathbf{d}^i)^T(\mathbf{x} - \mathbf{x}^i) \leq f_i(\mathbf{x})$  for all  $\mathbf{x} \in \operatorname{dom}(f_i)$ .

4. it exists  $\eta > 0$ , such that  $a_{ii}(k) > \eta$  and, if  $a_{ij}(k) > 0$ , then  $a_{ij}(k) \geq \eta$ ;
5. the information of each agent  $i$  reaches every other agent  $j$  (directly or indirectly) infinitely often;
- 6') there is a deterministic bound for the intercommunication interval between two nodes.

We better formalize conditions 5 and 6' (resp. 6''). Consider the graph  $(\mathcal{V}, E_\infty)$ , where  $\mathcal{V}$  is the set of nodes and the edge  $(i, j)$  belongs to  $E_\infty$  if nodes  $i$  and  $j$  communicate infinitely often (i.e. if  $a_{ij}(k)$  is positive for infinite values  $k$ ). Condition 5 imposes that the graph  $(\mathcal{V}, E_\infty)$  is (strongly) connected. Condition 6' requires that there is a positive integer constant  $B$ , such that two nodes communicating infinitely often, communicate at least once every  $B$  slots, i.e. if  $(i, j) \in E_\infty$ , then  $\max\{a_{ij}(k), a_{ij}(k+1), \dots, a_{ij}(k+B-1)\} > 0$ .

In [3], the assumption on the deterministic bound on the inter-meeting time is removed, but matrices are required to be independently and identically distributed. In fact, condition 6' is replaced by the following one:

- 6'') matrices  $\mathbf{A}(k)$  are i.i.d. random matrices.

In such case,  $(i, j) \in E_\infty$  if and only if  $E[a_{ij}(k) > 0]$ . Note also that, when the matrices  $\mathbf{A}(k)$  are random (like in 6''), condition 5 requires the matrix  $E[\mathbf{A}(k)]$  to be irreducible and aperiodic.

Both papers address also the case when the gradient step-size does not vanish, but it is kept constant ( $\gamma(k) = \gamma$ ). In this case, the sequence of estimates  $\mathbf{x}^i$  does not converge in general to a point of minimum of  $f$ , but it may keep oscillating around one of such point. It is possible to bound the difference between the values that  $f$  assumes at the points of a smoothed average of  $\mathbf{x}^i$  and the minimum of  $f$ . As it is intuitive, the smaller  $\gamma$ , the smaller such difference.

We are going to provide an intuitive explanation of why the result holds, and an outline of the proofs in [1,3]. This will be useful for our following extensions. We first formulate (1) in matrix form as follows:

$$\mathbf{X}(k+1) = \mathbf{A}(k)\mathbf{X}(k) - \gamma(k)\mathbf{D}(k), \quad (2)$$

where

$$\mathbf{X}(k) \stackrel{\text{def}}{=} [\mathbf{x}^1(k+1), \dots, \mathbf{x}^i(k+1), \dots, \mathbf{x}^M(k+1)]^T$$

and

$$\mathbf{D}(k) \stackrel{\text{def}}{=} [\mathbf{d}^1(k+1), \dots, \mathbf{d}^i(k+1), \dots, \mathbf{d}^M(k+1)]^T.$$

This equation iteratively leads to

$$\begin{aligned} \mathbf{X}(k+1) &= \mathbf{A}_{(1)}^{(k)}\mathbf{X}(1) - \sum_{s=2}^k \mathbf{A}_{(s)}^{(k)}\gamma(s-1)\mathbf{D}(s-1) + \\ &\quad - \gamma(k)\mathbf{D}(k), \end{aligned} \quad (3)$$

where  $\mathbf{A}_{(s)}^{(k)}$ , with  $s, k \geq 1$  and  $s \leq k$ , is the *backward matrix product*, i.e.  $\mathbf{A}_{(s)}^{(k)} = \mathbf{A}(k)\mathbf{A}(k-1)\dots\mathbf{A}(s)$ . We note that the first addend in the right side corresponds to a standard consensus algorithm.

We introduce the average of all the nodes estimates,  $\mathbf{y}(k) \in \mathbb{R}^M$ , defined as:

$$\mathbf{y}(k)^T = \frac{1}{M} \mathbf{1}^T \mathbf{X}(k).$$

By (2), we obtain

$$\begin{aligned} \mathbf{y}(k+1)^T &= \frac{1}{M} \mathbf{1}^T \mathbf{A}(k) \mathbf{X}(k) - \mathbf{1}^T \frac{\gamma(k)}{M} \mathbf{D}(k) = \\ &= \mathbf{y}(k)^T - \frac{\gamma(k)}{M} \mathbf{1}^T \mathbf{D}(k). \end{aligned} \quad (4)$$

Assume for a moment that  $\mathbf{x}^i(k) = \mathbf{x}^j(k) = \mathbf{y}(k)$ , for each  $i$  and  $j$ , then subgradients  $\mathbf{d}^i(k)$  are all evaluated in  $\mathbf{y}(k)$  and  $\mathbf{1}^T \mathbf{D}(k)$  is a sub-gradient of the global function  $f$  evaluated in  $\mathbf{y}(k)$ . Thus, the above equation (4) corresponds to a basic iteration of a sub-gradient method for the global function  $f$ . The intuitive explanation of the result is that averaging keeps the estimates of  $\mathbf{x}^i(k)$ , for all  $i$ , close each other (and then close to  $\mathbf{y}(k)$ ) and makes the local sub-gradients updates equivalent to a sub-gradient update for the global function  $f$ .

We illustrate this through a simple toy example that we are going to use different times across this paper. Consider three nodes, labeled as 1, 2 and 3. Their local objective functions are  $f_1(x) = f_2(x) = x(x-1)/2$  and  $f_3(x) = 2x^2$ , where  $x \in \mathbb{R}$ . Then the global function is  $f(x) = \sum_i f_i(x) = 3x^2 - x$ , it has minimum value equal to  $-1/12$  and a unique point of minimum in  $x = 1/6$ . The weight matrices  $\mathbf{A}(k)$  are i.i.d. random matrices. At each step  $\mathbf{A}(k)$  is equal to one of the following three matrices

$$\begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 1 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix}, \quad (5)$$

with probability  $2/3$ ,  $1/3$  and  $1/3$ , respectively. Figure 1 shows the evolution of the estimates at the 3 nodes, when the algorithm is applied with  $\gamma(k) = 1/k$ . We can see that state's estimates tends to couple and then converge to the optimal value. In particular, estimates of node 1 and node 2 are kept closer each other since the first matrix of above is selected with higher probability. Similar results have been obtained with  $\gamma(k) = \gamma \ll 1$ .

The proofs of the convergence results in [1, 3] share mainly the same outline. A key element is proving that the averaging component (the consensus) of the algorithm converges exponentially fast. More formally, under  $\mathcal{G}'$ , [1] proves that  $\mathbf{A}_{(s)}^{(k)}$  surely converges to the matrix  $\mathbf{J} = 1/M \mathbf{1} \mathbf{1}^T$  and that there are two positive constants  $C$  and  $\beta$  such that  $\|\mathbf{A}_{(s)}^{(k)} - \mathbf{J}\|_\infty \leq C\beta^{k-s}$  for all  $k \geq s$ . Under  $\mathcal{G}''$ , [3] proves almost surely convergence of  $\mathbf{A}_{(s)}^{(k)}$  to  $\mathbf{J}$ , and an exponential convergence rate in expectation, i.e.  $E[\|\mathbf{A}_{(s)}^{(k)} - \mathbf{J}\|_\infty] \leq C\beta^{k-s}$ . Then, similar bounds are established for the distance between  $\mathbf{y}$  and  $\mathbf{x}^*$ , and between  $\mathbf{x}^i$  and  $\mathbf{y}$ , and convergence results (when  $\gamma(k)$  satisfies condition 5) and asymptotic bounds (when  $\gamma(k)$  is constant) follow from the exponential convergence rate of the averaging component. As a consequence of the different kind of convergence for  $\mathbf{A}_{(s)}^{(k)}$  in the two cases, these results hold surely under  $\mathcal{G}'$  and almost surely under  $\mathcal{G}''$ .

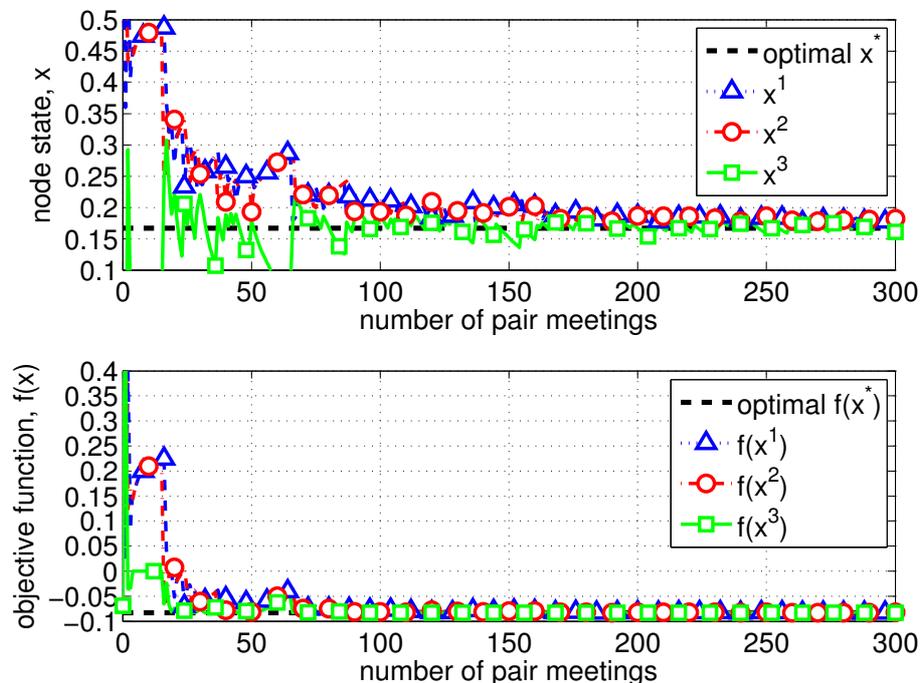


Figure 1: Toy example, convergence of the three estimates. Top: state's estimates for each node. Bottom: objective function value computed in the state's estimates of each node.

### 3 Application to Optimization in DTNs: Motivation

The distributed sub-gradient method presented in Section 2 is particularly appealing in the context of Delay Tolerant Networks, see e.g., [6] and [7]. DTNs are sparse and/or highly mobile wireless ad hoc networks where no continuous connectivity guarantee can be assumed. This intrinsically may lead to the impossibility of gathering to a single data processing point all the information needed to solve network optimization problems using centralized solutions. In this optic distributed methods appear to be therefore the right approach. To be more concrete we briefly present and discuss two possible DTN scenarios in which a global network's function  $f$  has to be optimized.

One central problem in DTNs is related to routing packets towards the intended destination. Common techniques, designed to overcome the absence of a complete route to the destination, rely on multi-copy dissemination of the message in the network. In this context it is natural to define global optimization functions that are able to take into account the trade-off between delivery time and the expense due to an increased use of resources such as buffer space, bandwidth and transmission power. Functions of this kind are convex and can be written as sum of locally measurable quantities, then can be optimized through the above distributed sub-gradient framework, see [8].

The second DTN scenario that we present concerns the dissemination of dynamic content, like news or traffic information. According to [5], a Service Provider (SP) with limited bandwidth, that has to decide the update rate to assign at each node in the network. Nodes can share their content when meet and the overall aim is to maintain the average information in the network as “fresh” as possible. [5] shows that this problem can be formalized as a classical convex optimization problem, and the corresponding global objective function can easily be seen as a sum of local ones. To derive these latter functions, each node needs to collect statistics on its own meeting processes only. Accordingly, in Section 6 we show how to apply the framework presented in Section 2 to this scenario.

From a general perspective, when we want to apply the distributed subgradient method to optimization in DTNs, we can imagine that nodes exchange their local estimates every time they meet and perform the update step in equation (1) at a given sequence of time instants  $\{t_k\}_{k \geq 1}$ . Consequently, the weight matrices  $\mathbf{A}(k)$  originate from the node meeting process. In particular we can consider the contact matrix  $\mathbf{C}(k)$ , where  $c_{ij}(k) = 1$  if node  $i$  has met node  $j$  since last update time instant  $t_{k-1}$ , and  $c_{ij}(k) = 0$  otherwise. We denote as  $\mathcal{C}$  the (finite) set of all the possible  $M \times M$  contact matrices describing the contacts among  $M$  nodes. Each node  $i$  can then calculate its own weights  $a_{ij}(k)$ , for  $j = 1, \dots, M$ , in one of the following two ways (which guarantee that the matrix  $\mathbf{A}(k)$  is doubly stochastic):

**Rule 1** (Updates independent from meetings) For  $j \neq i$ , set  $a_{ij}(k) = 1/M$  if  $c_{ij}(k) = 1$ , otherwise set  $a_{ij}(k) = 0$ . Set  $a_{ii}(k) = 1 - \sum_{j \neq i} a_{ij}(k)$ . This method requires each node to know  $M$ , the total number of nodes in the system.

**Rule 2** (Updates synchronized with meetings) Whenever node  $i$  meets node  $j$ , it also updates its estimate. In this case, set  $a_{ij}(k) = a$  with  $0 < a < 1$ ,  $a_{ii}(k) = 1 - a$  and  $a_{ih} = 0$  for  $h \neq j, i$ .

Now, we envision two main issues to guarantee the convergence of the distributed optimization process in a DTN scenario. The first one is related to the validity of assumptions 6' and 6''. In fact, condition 6' is essentially equivalent to assume that there is a deterministic bound for the intermeeting time of two nodes (that meet infinitely often), and this is for example not the case for all the random mobility models usually considered. Condition 6'' relaxes 6', but requires the independence of meetings occurring in each time slot  $[t_{k-1}, t_k]$ , and meetings under realistic mobility are instead correlated (e.g., if in the recent past  $i$  has met  $j$  and  $j$  has met  $h$ , then the three nodes are close in space and the probability that  $i$  meets  $h$  is higher than a priori). We address this issue in Section 4, where we prove that convergence results hold under more general hypothesis on the stochastic process of the matrices  $\mathbf{A}(k)$ .

The second issue is related to the synchrony in the updates. In fact, the original framework requires all the nodes to update their estimates at the same time instant. This is not always feasible particularly in a disconnected scenario like a DTN. For example, under Rule 2 the reader may have noted that update synchrony requires each node to know when a meeting between two any other nodes in the network is happening. This does not appear to be realistic. Under Rule 1, that requires each node to know the total number of nodes in the system,

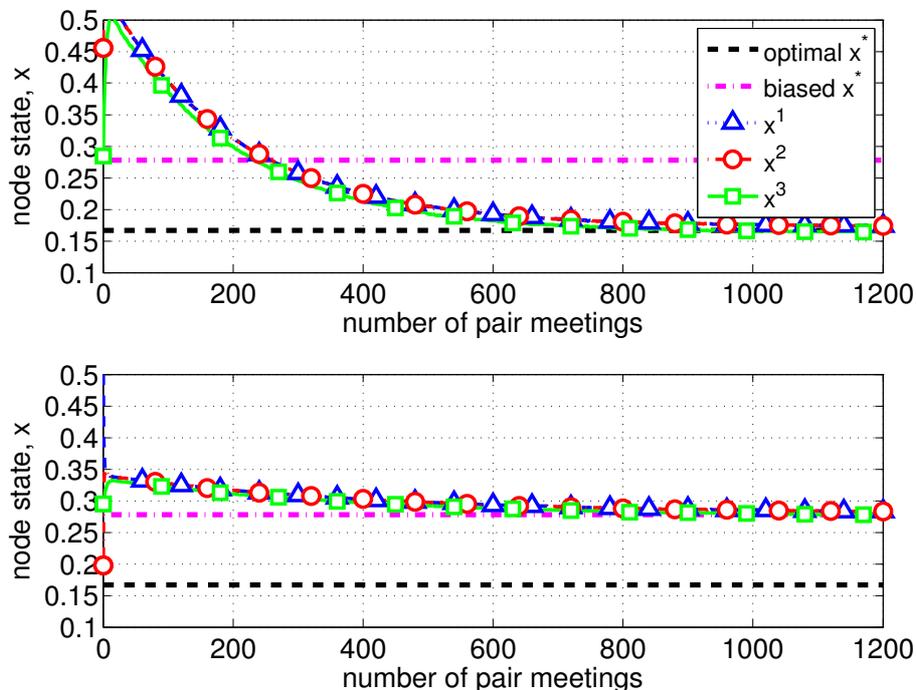


Figure 2: Toy example, convergence of the three estimates in case of fixed step-size  $\gamma = 25 \cdot 10^{-4}$ . Results have been averaged over 100 simulation runs. Top: synchronous updates. Bottom: asynchronous updates.

nodes should also try to keep their internal clocks synchronized in order to be able to perform their updates at “close enough” time instants and this presents some difficulties as well. We show through an example that we cannot simply ignore the issue of update synchrony and a straight application of the algorithm described in the previous section does not lead to correct results. Coming back to the toy example presented in Section 2, we observe that we can think our three matrices in (5) as generated according to Rule 2, when the meeting process has the following characteristics: at each time slot, node 1 and node 2 meet with probability  $2/3$ , node 1 and 3 meet with probability  $1/3$  and node 2 and 3 meet with probability  $1/3$ . Figure 2 shows the evolution of the estimates when the stepsize is constant and equal to  $25 \cdot 10^{-4}$ , both for the synchronous case, where all the nodes update their estimates when a meeting occurs (even the node that is not involved in the meeting), and for the asynchronous case, where only the nodes involved in the meeting do the update. The curves represent the average estimates over 100 different simulations with different pair meetings sequences. We note that whilst in the synchronous case, top figure, all nodes agree on the optimal value to set  $x$ , in the asynchronous case, bottom figure, the estimates still converge, but not to the correct point of minimum for the function  $f$ . We address this issue in Section 5, where we understand the roots of this problem and propose some simple changes to the basic framework to compensate this distortion (but a formal proof is still missing).

In our opinion, these extensions to the basic framework proposed in [1, 3], while motivated in this paper by the DTN scenario, are of wide interest for other possible applications.

## 4 Extension to more General Mobility Models

In our DTN scenario, we consider that the weights are determined from the contacts through a bijective function (as in the case of the two rules presented in Sec. 3). Then condition 5 and 6'' of [3], can be expressed in terms of the sequence of contact matrices as follows: the contact matrices  $\mathbf{C}(k)$  are i.i.d. and  $E[\mathbf{C}(k)]$  is an irreducible aperiodic matrix. In this section, we extend the convergence results to the following, more general, mobility model.

**Assumption 1** (Mobility model). *It exists an irreducible, aperiodic and stationary Markov chain  $\Phi$  with a finite or countable set of states  $S$  and a function  $g : S \rightarrow \mathcal{C}$ , such that  $\mathbf{C}(k) = g(\Phi_k)$ , for each  $\Phi_k \in S$ . Moreover,  $E[\mathbf{C}(k)]$  is an irreducible aperiodic matrix.*

Since there is a bijective correspondence among weight and contact matrices, we observe that under assumption 1, it also exists a function  $\hat{g} : S \rightarrow \mathcal{A}$ , such that  $\mathbf{A}(k) = \hat{g}(\Phi_k)$ . The case when the contact matrices (and then the weight matrices) are i.i.d. is a particular case of our mobility model.

Our proof follows the same outline of [1, 3] presented in Sec. 2. The main issue is to prove the exponential rate of convergence of the backward product  $\mathbf{A}_{(1)}^{(k)}$  to  $\mathbf{J} = 1/M\mathbf{1}\mathbf{1}^T$ .

Before proving the convergence of  $\mathbf{A}_{(1)}^{(k)}$ , we need to recall an ergodic property of the time shift operator  $\theta$  for irreducible, aperiodic and stationary Markov chains. The definitions of measure-preserving and ergodic operators may be found in Appendix B (see also chapter V of [9] for more details).

**Proposition 1.** *Given an irreducible aperiodic and stationary Markov chain  $\Phi$  with finite or countable states, the shift operator  $\theta$  is measure-preserving and ergodic together with all its powers  $\theta^k$ , where  $k \in \mathbb{N}$ .*

*Proof.* For stationary Markov chains the shift operator and its powers are measure-preserving by definition of stationarity. Moreover, irreducible, aperiodic and stationary Markov chains with finite or countable states are mixing (see Theorem 3.1 in [10]). From the definition of mixing, if  $\theta$  is mixing, also the operator  $\theta^k$  is mixing for any given  $k \in \mathbb{N}$ . But every mixing operator is also ergodic by Theorem 2 in [9], then  $\theta^k$  is ergodic for any  $k \in \mathbb{N}$ .  $\square$

We observe that the stochastic process  $\mathbf{A}(k) = \hat{g}(\Phi_k)$  is not in general a Markov chain, because different states of  $S$  may be mapped to the same weight matrix, but nevertheless it is stationary and ergodic.

We will also need the following result, whose proof is in Appendix C.

**Lemma 1.** *(Windowing a Markov chain) Let  $\Phi = \{\Phi_n, n \in \mathbb{N}\}$  be an irreducible, aperiodic and stationary Markov chain. Consider the stochastic process  $\Psi = \{\Psi_n, n \in \mathbb{N}\}$ , where  $\Psi_n = (\Phi_n, \Phi_{n+1}, \dots, \Phi_{n+h-1})$  with  $h$  a positive integer.  $\Psi$  is also an irreducible aperiodic stationary Markov chain.*

Now we have all the instruments to study the convergence of  $\mathbf{A}_{(1)}^{(k)}$ . First, we prove the convergence to  $\mathbf{J}$ . This is a corollary of results in [4].

**Proposition 2** (Convergence of the backward product). *Let assumption 1 hold, then*

$$\lim_{k \rightarrow +\infty} \mathbf{A}_{(1)}^{(k)} = \frac{1}{M} \mathbf{1}\mathbf{1}^T \triangleq \mathbf{J} \quad \text{almost surely (a. s.) .}$$

*Proof.* We observe that  $\mathbf{A}(k)$  is a stationary and ergodic sequence of stochastic matrices with strictly positive diagonal entries. Moreover,  $\mathbb{E}[\mathbf{A}(k)]$  is an irreducible aperiodic matrix, then its eigenvalue with the second largest module has module strictly smaller than 1 ( $|\lambda_2(\mathbb{E}[\mathbf{A}(k)])| < 1$ ). From Theorem 3 in [4], it follows that, with probability one, for each sequence  $\mathbf{A}(k)$  it exists a vector  $\mathbf{v} \in \mathbb{R}^M$ , such that  $\sum_i v_i = 1$  and

$$\lim_{k \rightarrow +\infty} \mathbf{A}_{(1)}^{(k)} = \mathbf{1}\mathbf{v}^T .$$

Note that in general  $\mathbf{v}$  is a random variable, depending on the specific sequence  $\{\mathbf{A}(k)\}_{k \geq 1}$ . The matrices  $\mathbf{A}(k)$  are doubly stochastic, then  $\mathbf{w} = \mathbf{1}/M\mathbf{1}$  is a left eigenvalue corresponding to the unit eigenvalue for all the matrices  $\mathbf{A}(k)$ . Theorem 4 in [4], guarantees that in this case the above vector  $\mathbf{v}$  is a deterministic constant almost surely and in particular is equal to  $\mathbf{w}$ . This concludes the proof.  $\square$

Now we are ready to prove that the convergence rate is almost always exponential.

**Proposition 3.** *Under assumption 1 on the mobility models, if the matrices are doubly stochastic, then for almost all the sequences there exist  $C > 0$  and  $0 < \beta < 1$  (with  $C$  in general depending of the sequence) such that for  $k > s$*

$$\left\| \mathbf{A}_{(s)}^{(k)} - \mathbf{J} \right\|_{\max} \leq C\beta^{k-s} .$$

*Proof.* Given a matrix  $\mathbf{A}$ , consider the coefficient of ergodicity, see e.g., [11]:

$$\tau_1(\mathbf{A}) = \frac{1}{2} \max_{i,j} \sum_{s=1}^M \left| [\mathbf{A}]_{i,s} - [\mathbf{A}]_{j,s} \right| .$$

In the proof of Theorem 3 in [4] is shown that it exists a positive natural  $h$  and  $\eta < 1$  such that

$$\mathbb{P} \left[ \tau_1 \left( \mathbf{A}_{(s+(r-1)h)}^{(s+rh-1)} \right) < \eta \text{ for infinitely many } r \right] = 1 . \quad (6)$$

Then we decompose  $\mathbf{A}_{(s)}^{(k)}$ , in the product of  $i_k$  blocks of size  $h$  and one block of size  $(k+1) \bmod h$  as it follows:

$$\mathbf{A}_{(s)}^{(k)} = \mathbf{A}_{(s+i_k h)}^{(k)} \mathbf{A}_{(s+h(i_k-1))}^{(s+hi_k-1)} \dots \mathbf{A}_{(s)}^{(s+h-1)} .$$

Because of the properties of a coefficient of ergodicity:

$$\begin{aligned} \tau_1 \left( \mathbf{A}_{(s)}^{(k)} \right) &\leq \tau_1 \left( \mathbf{A}_{(s+i_k h)}^{(k)} \right) \prod_{j=1}^{i_k} \tau_1 \left( \mathbf{A}_{(s+h(j-1))}^{(s+hj-1)} \right) \leq \\ &\leq \prod_{j=1}^{i_k} \tau_1 \left( \mathbf{A}_{(s+h(j-1))}^{(s+hj-1)} \right) . \end{aligned}$$

Then, we can write:

$$\log \left( \tau_1 \left( \mathbf{A}_{(s)}^{(k)} \right) \right) \leq \sum_{j=1}^{i_k} \log \left( \tau_1 \left( \mathbf{A}_{(s+h(j-1))}^{(s+hj-1)} \right) \right) .$$

We now consider the Markov chain  $\Phi$ , that “generates” the sequence of matrices  $\mathbf{A}(k)$  underlying the mobility process. Because of Lemma 1,  $\Psi_t = (\Phi_t, \Phi_{t+1}, \dots, \Phi_{t+h-1})$  is an irreducible aperiodic stationary Markov chain, and then all the powers of the shift operator  $\theta$  and in particular  $\theta^h$ , are ergodic. We observe that  $\log\left(\tau_1\left(\mathbf{A}_{(j)}^{(j+h-1)}\right)\right)$  is a function of  $(\Phi_j, \Phi_{j+1}, \dots, \Phi_{j+h-1})$  and then of  $\Psi_j$ . We call such function  $f$ , i.e.,  $f(\Psi_j) \stackrel{\text{def}}{=} \log\left(\tau_1\left(\mathbf{A}_{(j)}^{(j+h-1)}\right)\right)$ . Note that  $f(\Psi_t) \leq 0$  and, from equation (6),  $f(\Psi_t) < \log(\eta) < 0$  infinitely often almost surely. We can then apply Birkhoff’s ergodic theorem to the random sequence  $\{f(\Psi_s), f(\Psi_{s+h}), f(\Psi_{s+2h}), \dots\}$ , following that:

$$\begin{aligned} & \lim_{i \rightarrow \infty} \frac{1}{i} \sum_{j=1}^i \log\left(\tau_1\left(\mathbf{A}_{(s+h(j-1))}^{(s+hj-1)}\right)\right) = \\ & = \lim_{i \rightarrow \infty} \frac{1}{i} \sum_{j=1}^i f(\Psi_{s+hj}) = \mathbb{E}[f(\Psi_t)] < 0 \quad \text{a. s. ,} \end{aligned}$$

therefore,

$$\limsup_{h \rightarrow \infty} \frac{1}{h} \log\left(\tau_1\left(\mathbf{A}_{(s)}^{(s+h)}\right)\right) \leq \mathbb{E}[f(\Psi_t)] < 0 \quad \text{a. s. .}$$

Consider  $\mathbb{E}[f(\Psi_t)] < \zeta < 0$ , then for almost all the sequences it exists  $h_0$ , such that for all  $h \geq h_0$ , it holds:

$$\frac{1}{h} \log\left(\tau_1\left(\mathbf{A}_{(s)}^{(s+h)}\right)\right) \leq \zeta, \quad \text{i.e., } \tau_1\left(\mathbf{A}_{(s)}^{(s+h)}\right) \leq e^{\zeta h} .$$

If we define  $\beta = \exp(\zeta) < 1$ ,  $C = \beta^{-h_0}$  and recall that  $\tau_1\left(\mathbf{A}_{(s)}^{(k)}\right) \leq 1$ , we obtain:

$$\tau_1\left(\mathbf{A}_{(s)}^{(k)}\right) \leq C\beta^{k-s} \quad \text{for } k \geq s . \quad (7)$$

In the above equation, the value of the constant  $h_0$  depends on the specific random sequence and also on  $s$  (while the same value  $\zeta$  can be selected for all the sequences and independently from  $s$ ). We need then to use a corollary of the ergodic theorem about “nearly uniform” convergence that is stated as proposition (1.5) in [12]: if  $f(\cdot)$  is square integrable, then for almost all the sequences we can select  $h_0$  independently from  $s$ . Thus, we can conclude that  $C$  in (7) only depends on the considered sequence.

So far we have established the existence of a geometric convergence result for the ergodic coefficient  $\tau_1$ . The last step of our proof requires us to prove a geometric bound for the distance between  $\mathbf{A}_{(s)}^{(k)}$  and its a. s. limit  $\mathbf{J}$ . In particular we prove that

$$\left| \left[ \mathbf{A}_{(s)}^{(k)} \right]_{uv} - \frac{1}{M} \right| \leq 2\tau_1\left(\mathbf{A}_{(s)}^{(k)}\right) , \quad (8)$$

for each  $u$  and  $v$ . From (8) a geometric bound follows also for  $\|\mathbf{A}_{(s)}^{(k)} - \mathbf{J}\|_{\max}$ . The derivation of (8) has no particular difficulty and mainly follows the proof of Theorem 4.17 in [11], then we have moved it to Appendix D.  $\square$

In [3] a different result it is proven, i.e. that there exist  $\hat{C}$  and  $\hat{\beta}$  such that

$$\mathbb{E} \left[ \left\| \mathbf{A}_{(s)}^{(k)} - \mathbf{J} \right\|_{\max} \right] \leq \hat{C} \hat{\beta}^{k-s} .$$

Then a series of inequalities for the expected values of  $\|\mathbf{y}(k) - \mathbf{x}^i(k)\|_2$  are obtained for all  $i$ . Using Fatou's Lemma, along with the non-negativeness of distances, it is possible to derive inequalities that hold with probability 1. Using Proposition 3, instead, it is possible to obtain the same inequalities directly without the need to consider the expectation.

We observe that an exponential bound for the convergence of  $\mathbf{A}_{(s)}^{(k)}$  analogous to (7) (in the sense that the constant  $C$  depends on both the random sequence and  $s$ ), was proposed in [13], under assumptions weaker than ours (that only require stationarity and ergodicity of  $\mathbf{A}(k)$ ). Unfortunately, that proof is not correct.

## 5 Asynchronous Updates

In this section, we study how the framework needs to be extended in order to support the case when nodes asynchronously their status. We consider the sequence  $t_k$  of time instants at which one or more nodes perform an update of their estimates. Again, we denote that the estimate of node  $i$  at time  $t_k$  (immediately before the update) is  $\mathbf{x}^i(k)$  and represent all the estimates through the matrix  $\mathbf{X}(k)$ . The evolution of the estimates can still be represented in a matricial form similarly to (2):

$$\mathbf{X}(k+1) = \mathbf{A}(k)\mathbf{X}(k) - \Gamma(k)\mathbf{D}(k) \quad (9)$$

where  $\Gamma(k)$  is a diagonal matrix where the element  $(\Gamma(k))_{ii}$  is simply the stepsize used by node  $i$  at the  $k$ -th update. We denote this stepsize as  $\gamma_i(k)$ . If node  $j$  does not perform any update at that time of the  $t_k$ , then it will simply be  $a_{jj}(k) = 1$ ,  $a_{jh}(k) = 0$  for  $h \neq j$  and  $\gamma_{jj}(k) = 0$ .

In what follows we will first consider the case of decreasing stepsizes, similarly to condition 1, described in Sec. 2. That is, we will consider that for each  $i$ , the sequence  $\{\gamma_i(k)\}_{k \geq 1}$  satisfies:  $\sum_{k=1}^{\infty} \gamma_i(k) = \infty$  and  $\sum_{k=1}^{\infty} \gamma_i(k)^2 < \infty$ .

We can go over the proof in [3] and prove similar results for the new system description. In particular, our proof of the exponential convergence rate of  $\mathbf{A}_{(s)}^{(k)}$  holds clearly also in this case. Bounds for the distance between  $\mathbf{x}^i(k)$  and  $\mathbf{y}(k) = 1/M \mathbf{1}^T \mathbf{X}(k)$  hold with minimal changes, so that we can prove the analogous of Proposition 2 in [3]:

**Proposition 4** (Convergence of Agent Estimates). *Under assumption 1, the estimate of each node converges almost surely to the vector  $\mathbf{y}(k)$ , i.e.*

$$\lim_{k \rightarrow +\infty} \|\mathbf{y}(k) - \mathbf{x}^i(k)\|_2 = 0 \quad \text{a.s., for all } i.$$

The proof is in Appendix E.

The following step in the proof is to use bounds for the distance between  $\mathbf{y}(k)$  and  $\mathbf{x}^*$  (a point of minimum of  $f$ ) to show that  $\lim_{k \rightarrow +\infty} \mathbf{y}(k) = \mathbf{x}^*$ .

In particular the following inequality is derived in [3] (for the synchronous case they are considering):

$$\begin{aligned} \sum_{s=1}^k \gamma(s) [f(\mathbf{y}(s)) - f(\mathbf{x}^*)] &\leq \frac{M}{2} \|\mathbf{y}(1) - \mathbf{x}^*\|_2^2 + \\ &+ 2L \sum_{j=1}^M \sum_{s=1}^k \gamma(s) \|\mathbf{y}(s) - \mathbf{x}^j(s)\|_2 + \frac{L^2}{2} \sum_{s=1}^k \gamma^2(s) \end{aligned}$$

The first term at the right hand side of the above inequality is a constant, the last term is summable because of the assumption on the stepsizes. In [3] it is proven that  $\sum_{s=1}^{\infty} \gamma(s) \|\mathbf{y}(s) - \mathbf{x}^j(s)\|_2 < \infty$  almost surely. Thus they can conclude that

$$0 \leq \sum_{s=1}^{\infty} \gamma(s) [f(\mathbf{y}(s)) - f(\mathbf{x}^*)] < \infty \quad \text{a.s.,} \quad (10)$$

and since  $\sum_{s=1}^{\infty} \gamma(s) = \infty$ , it is possible to conclude that

$$\liminf_{k \rightarrow \infty} f(\mathbf{y}(k)) = f(\mathbf{x}^*) \quad \text{a.s.}, \quad \text{and} \quad \lim_{k \rightarrow \infty} \mathbf{x}^i(k) = \mathbf{x}^* \quad \text{a.s.}.$$

In Appendix, a similar derivation is carried on, leading to the following generalization of (10):

$$\sum_{s=1}^{\infty} \sum_{i=1}^M \gamma_i(s) [f_i(\mathbf{y}(s)) - f_i(\mathbf{x}^*)] < \infty \quad \text{a.s.}, \quad (11)$$

Unfortunately, the different values of  $\gamma_i(s)$  do not allow us to formulate the inequality above in terms of the global function  $f$  as in (10).

We do not have currently a formal result stating which conditions the asynchronous system converges to the optimal solution, but (11) suggests us that us that all the weights  $\gamma_i(k)$  should have on the average the same value. We then propose the following conjecture, that we support later with some examples:

**Conjecture 1.** *When updates are asynchronous, convergence results for subgradient methods hold if  $E[\gamma_i(k)] = E[\gamma_j(k)]$  for each  $i$  and  $j$ .*

Let us see how we can guarantee this condition in different cases. We consider that updates occur after every meeting following rule 2 in Sec. 3. Moreover consider that  $\gamma_i(k) = 1/n_i(k)$ , where  $n_i(k)$  is the total number of updates node  $i$  has performed until the time instant  $t_k$ . If the meeting process follows a Poisson process with total rate  $\mu$  and at each instant the probability that node  $i$  meets another node is  $p_i$ , we expect that by time  $k$ , node  $i$  has had  $p_i k$  meetings (and an equal number of updates). Then the expected value of its stepsize is  $E[\gamma_i(k)] = 1/(p_i k) p_i = 1/k$ . In conclusion if stepsizes follow the rule  $\gamma_i(k) = 1/n_i(k)$ , we expect the asynchronous subgradient mechanism to converges to the optimal solution. Fig. ??? shows that this is the case for our toy example. The simulations for the optimization problem considered in Sec. 6 confirm such convergence.

Let us now revisit the example in Sec. 3 showing that the estimates were not converging to a point of minimum. In the example stepsizes were constant, i.e.  $\gamma_i(k) = \gamma$ . Now, reasoning as above we can conclude that  $E[\gamma_i(k)] = p_i \gamma$ . Hence the expected values are not equal as far as nodes meeting rates (and then update rates) are not equal, as it was in the example where  $p_1 = 5/6$ ,  $p_2 = 5/6$  and  $p_3 = 1/3$ <sup>3</sup>. Intuitively, we expect convergence to be biased towards values closer to the optimum of the local functions of those nodes that perform the updates more often. Equation (11) suggests us that what the distributed mechanism is really doing was to minimize the function  $\sum_i p_i f_i$  rather than  $f = \sum_i f_i$ . This is the case, being that the estimates are converging to ???. If now we want to correct the function, it is sufficient to consider that each node selects its stepsize inversely proportional to its meeting rate. Fig. ??? shows that this correction leads the estimates to converge to the correct results.

<sup>3</sup> Note that meetings always involve two nodes, this is the reason why  $p_1 + p_2 + p_3 = 2$ .

## 6 Application to Optimization in DTNs: a Case Study

In this Section we present a possible DTN scenario in which the distributed sub-gradient method can be exploited. In doing so we strongly relate to the framework proposed in [5]. In details, as explained in Section 3, we consider a mobile network of  $M$  users all interested in collecting the same dynamic information content. Let us to index the considered nodes from 1 to  $M$ , and, accordingly, indicate with  $\mathcal{N} = \{1, \dots, M\}$  the set of such nodes in the network. The information update in the network is performed by a Service Provider (SP) with limited bandwidth  $\mu$ , expressed in  $[\text{sec}^{-1}]$ . In other words the update process that inject information in the network is assumed to be Poisson with parameter  $\mu$ . Further, the users in the network can share their information content when meet other users and on the basis of some rules to be specified. Each node  $i$  in the network has also a local utility function  $u_i$  that is related to the freshness of the content stored at the node itself; roughly speaking, higher the time at which the content stored at node  $i$  was generated by the SP, higher its utility. Formally, at a given instant  $\bar{t}$  we call  $t_i(\bar{t})$  the time at which the SP generated the most recent content version available at node  $i$ , thus we can compute the corresponding age of such version as  $Y_i(\bar{t}) = \bar{t} - t_i(\bar{t})$ .  $Y_i(\bar{t})$  can be modeled as a random variable, that for easiness of notation we can indicate also with  $Y_i$ , omitting  $\bar{t}$ . Our objective is to maximizing the function

$$f(\mathbf{x}) = \sum_{i=1}^M f_i(\mathbf{x}) = \sum_{i=1}^M E_{\mathbf{x}}[u_i(Y_i)] , \quad (12)$$

where  $u_i : \mathbb{R}^+ \rightarrow \mathbb{R}$  are non-increasing functions and the state  $\mathbf{x}$  is an  $[M \times 1]$  rate allocation vector such that  $\sum_{i=1}^M x_i \leq \mu$  and  $x_i \geq 0$  for all  $i \in \mathcal{N}$ . In this scenario, Ioannidis et al. [5] prove that equation (12) is concave and therefore they propose to obtain the optimal  $\mathbf{x}$  using a projected gradient descent algorithm, see e.g., [14]. In general the function in (12) cannot be written in closed form and therefore an unbiased estimator for its gradient is required to iteratively perform the algorithm's step:

$$\mathbf{x}(k+1) = \Pi(\mathbf{x}(k) + \alpha_k \nabla f(\mathbf{x}(k))) , \quad (13)$$

where  $\{\alpha_k\}_{k \geq 0}$  is a positive sequence of parameters such that  $\sum_k \alpha_k = \infty$ ,  $\lim_{k \rightarrow \infty} \alpha_k = 0$ , and  $\Pi$  is the projection on the feasible set for  $\mathbf{x}$ .

According to [5] the gradient of (12) can be estimated both in a centralized and a distributed fashion. However, for illustrative purpose we simplify the file sharing scenario in [5] moving from an all-to-all unrestricted multi-copy protocol to an all-to-all two-hops protocol, see e.g., [15]. In the unrestricted multi-copy protocol each node in the network can copy its more recent content to the other nodes it meets; diversely, according to the two-hops protocol, each node can copy to other nodes only the information it received directly from the SP. Formally, at a given instant  $\bar{t}$  let us call  $t_j^{SP}(\bar{t})$  the time at which the last content downloaded directly from the SP has been injected to node  $j$ , then we define the following protocol's rule:

**Definition 1** (Content Sharing). When a node  $j$  meets a node  $i$  at time  $\bar{t}$ ,  $j$  will copy to user  $i$  the last content it downloaded directly from the SP if this content is more recent than the content stored in  $i$ , i.e.,  $t_i(\bar{t}) < t_j^{SP}(\bar{t})$ .

We assume also that: i) for all  $i \in \mathcal{N}$  the utility function  $u(Y_i)$  is  $\chi\{Y_i \leq \tau\}$ , where  $\chi\{y \leq \tau\}$  is equal to 1 if  $y \leq \tau$  and 0 otherwise, and ii) the meeting process among node pairs is Poisson distributed. In conclusion, considering the above non-decreasing utility functions, the update policy in the definition 1 and Poisson meeting processes, we have

$$f_i(\mathbf{x}) = 1 - \left[ \prod_{j \in \mathcal{N}_i} \frac{x_j e^{-\lambda_{ij}\tau} - \lambda_{ij} e^{-x_j\tau}}{x_j - \lambda_{ij}} \right] e^{-x_i\tau}, \quad (14)$$

where  $\mathcal{N}_i$  is the set of all the nodes in  $\mathcal{N}$  that can meet node  $i$ , whilst  $\lambda_{ij}$  is the meeting rate of the  $(i, j)$  node pair. The function in (12) is then simply obtained summing (14) for all  $i \in \mathcal{N}$ . The rationale to derive equation 14 can be found in Appendix F. The optimization of  $f(\mathbf{x})$  can now be solved both in a centralized manner from the SP or in a distributed fashion exploiting the distributed sub-gradient method framework presented in Section 2. The centralized solution requires that the SP knows the meeting rates among all the node pairs in the network, whilst the distributed approach requires each node to be aware of the total bandwidth available  $\mu$  and the statistic of its own meetings only. In a distributed fashion, we can think that each node  $i$  in the network has its own idea about the optimal allocation rate the SP should adopt, namely  $\mathbf{x}^i$ . The iterative step (13) is therefore performed as follows, see [16]: at each update step  $k$ , for all  $i \in \mathcal{N}$  state  $\mathbf{x}^i$  is 1) updated according to equation (1) and 2) projected into the feasible set  $\sum_{l=1}^M x_l^i \leq \mu$  and  $x_l^i \geq 0$  for all  $l \in \{1, \dots, M\}$ . The local gradient function for each node is easily computed from equation (14) as reported in Appendix F.

To test the performance achievable by the distributed sub-gradient method we simulated meeting events among 10 nodes as follows. Calling  $A$  and  $B$  two distinct regions of the space, nodes can be placed either in  $A$  or in  $B$ . Meeting events among node pairs occur according to a Poisson process of parameter  $\lambda_m$ . At each node  $i \in \mathcal{N} = \{1, \dots, 10\}$  is given a weight  $w_i$  and, if a meeting event takes place, two nodes  $i$  and  $j$  in the same region are selected proportionally to their weights. In details, node  $i$  is selected with probability  $w_i / \sum_{z \in \mathcal{N}} w_z$  and node  $j$ , in the same region of  $i$ , is selected with probability  $w_j / \sum_{z \in \mathcal{N}_{\sim i}} w_z$ , where  $\mathcal{N}_{\sim i}$  indicates the set of all the nodes different than  $i$  but in the same region of  $i$ . To guarantee full connectivity of the network, we also allow a uniformly chosen node to switch the region where staying according to a Poisson process of parameter  $\lambda_d < \lambda_m$ . In this manner we are able to generate a sequence of meeting events that is both stationary and ergodic, according to the theoretical framework extension provided in Section 4. In fact at each meeting event we allow the two nodes involved in the meeting to update their state as explained thus at each step of the process observed the matrices  $\mathbf{A}(k)$  (see Section 2) can be viewed as generated from an ergodic and stationary Markov chain. Furthermore, assign different weights  $w_i$  to nodes allow us to have not only asynchronous update, but also potentially unbalanced contribution to the overall maximization of the network, see toy example's discussion in Section 3.

Figures 3–5 show simulation results for  $\lambda_m = 1$ ,  $\lambda_d = 0.1$ ,  $\tau = 20$  and weights proportionally decreasing with the node indexes in  $\mathcal{N}$ .

Fig. 3 depicts the optimal allocation rate for each user. When the available bandwidth is very low the best solution is to continuously give updates to the node that has the higher contact rate, i.e. node 1 since the above choice of weights; for high bandwidths, instead, the optimal policy is to allocate updates uniformly among all the nodes in the network. In the intermediates case, as expected from the work [5], interestingly the optimal choice is to gradually diminishing the amount of fresh information to give directly to node with the highest contact rate and, in turn, provide more bandwidth to the nodes with less contact rates, i.e., nodes with higher indexes in our simulation.

In Fig. 4 we draw with a blue line the maximum of the utility function  $f(\mathbf{x})$  corresponding to the optimal rate allocations in Fig. 3. For six different values of the available bandwidth  $\mu$ , we also plot with squared points the utility function values corresponding to the optimal rate allocation achieved by the 10 nodes after several meetings, namely, when the convergence of the distributed sub-gradient method can be considered achieved. The results confirm that with ergodic and stationary meeting process the desired optimal is achievable. Issues related to the actual convergence rate of the method are out of the scope of this paper and will be part of the future research. In Fig. 4 the results for the distributed sub-gradient method have been obtained using decreasing step-sizes for all nodes. In case of constant step-size the reached performance lower due to the distortion introduced by asynchronous updates and different node contact rates, as discussed in Section 5. Diverse experimental results confirm also our overall intuition about the convergence to optimality in case of asynchronous updates: besides using decreasing step-size, we can also reach optimality assigning at each node a fixed step-size inversely proportional to its contact rate.

Finally, we recall that applying the distributed sub-gradient method, at each time instant, each node in the network has a state vector saying the best rate allocation policy according to itself. In particular, each node computes which should be its own update rate from the SP. Fig. 5 shows the evolution on the belief about their own portion of bandwidth for two nodes, i.e., the nodes with highest and lowest contact rates. The evolution lasts along a sequence of  $5 \cdot 10^5$  meetings and  $\mu = 10^{-1.1}$ . We can observe that eventually there is convergence to the desired optimal value for both the nodes. Namely, even both nodes are not maximizing their own local utility function, but the overall one boosting the performance of the file sharing mechanism.

Figure 3: Optimal bandwidth allocation for a network of 10 nodes.

## **7 Acknowledgements**

TBW.

Figure 4: Theoretical maximum of the utility function *vs* utility of the rate allocation obtained with the distributed sub-gradient method for six different bandwidth's values.

## 8 Conclusion

TBW.

Figure 5: Example of state convergence for two nodes when the sub-gradient method is used. Update performed at each pair node meeting, by only the nodes involved in the meetings and applying a decreasing step-size.

## A Derivation of Equation (3)

In this Appendix we show the mathematical steps that lead from equation (2) to equation (3).

From (2) we have that

$$\begin{aligned}
 \mathbf{X}(k+2) &= \mathbf{A}(k+1)\mathbf{X}(k+1) - \mathbf{\Gamma}(k+1)\mathbf{D}(k+1) = \\
 &= \mathbf{A}(k+1)[\mathbf{A}(k)\mathbf{X}(k) - \mathbf{\Gamma}(k)\mathbf{D}(k)] - \mathbf{\Gamma}(k+1)\mathbf{D}(k+1) = \\
 &= \mathbf{A}_{(k)}^{(k+1)}\mathbf{X}(k) - \mathbf{A}_{(k+1)}^{(k+1)}\mathbf{\Gamma}(k)\mathbf{D}(k) - \mathbf{\Gamma}(k+1)\mathbf{D}(k+1) ,
 \end{aligned}$$

that iteratively, replacing  $k + 2$  with  $k + s + 1$ ,  $k + 1$  with  $k + s$  and  $k$  with  $k + s - 1$ , respectively, leads to

$$\begin{aligned}
\mathbf{X}(k + s + 1) &= \mathbf{A}_{(k+s-1)}^{(k+s)} \mathbf{X}(k + s - 1) - \mathbf{A}_{(k+s)}^{(k+s)} \mathbf{\Gamma}(k + s - 1) \mathbf{D}(k + s - 1) + \\
&\quad - \mathbf{\Gamma}(k + s) \mathbf{D}(k + s) = \\
&= \mathbf{A}_{(k+s-1)}^{(k+s)} [\mathbf{A}(k + s - 2) \mathbf{X}(k + s - 2) - \mathbf{\Gamma}(k + s - 2) \mathbf{D}(k + s - 2)] + \\
&\quad - \mathbf{A}_{(k+s)}^{(k+s)} \mathbf{\Gamma}(k + s - 1) \mathbf{D}(k + s - 1) - \mathbf{\Gamma}(k + s) \mathbf{D}(k + s) = \\
&= \mathbf{A}_{(k+s-2)}^{(k+s)} \mathbf{X}(k + s - 2) - \mathbf{A}_{(k+s-1)}^{(k+s)} \mathbf{\Gamma}(k + s - 2) \mathbf{D}(k + s - 2) + \\
&\quad - \mathbf{A}_{(k+s)}^{(k+s)} \mathbf{\Gamma}(k + s - 1) \mathbf{D}(k + s - 1) - \mathbf{\Gamma}(k + s) \mathbf{D}(k + s) = \\
&= \mathbf{A}_{(k+s-2)}^{(k+s)} \mathbf{X}(k + s - 2) - \sum_{l=0}^1 \mathbf{A}_{(k+s-l)}^{(k+s)} \mathbf{\Gamma}(k + s - 1 - l) \mathbf{D}(k + s - 1 - l) + \\
&\quad - \mathbf{\Gamma}(k + s) \mathbf{D}(k + s) = \\
&\quad \vdots \\
&= \mathbf{A}_{(1)}^{(k+s)} \mathbf{X}(1) - \sum_{l=0}^{k+s-2} \mathbf{A}_{(k+s-l)}^{(k+s)} \mathbf{\Gamma}(k + s - 1 - l) \mathbf{D}(k + s - 1 - l) + \\
&\quad - \mathbf{\Gamma}(k + s) \mathbf{D}(k + s) ,
\end{aligned}$$

and finally, replacing  $k + s$  with  $k$ , we have that

$$\mathbf{X}(k + 1) = \mathbf{A}_{(1)}^{(k)} \mathbf{X}(1) - \sum_{l=0}^{k-2} \mathbf{A}_{(k-l)}^{(k)} \mathbf{\Gamma}(k - 1 - l) \mathbf{D}(k - 1 - l) - \mathbf{\Gamma}(k) \mathbf{D}(k)$$

or, replacing in equation (15)  $k - l$  with  $s$

$$\mathbf{X}(k + 1) = \mathbf{A}_{(1)}^{(k)} \mathbf{X}(1) - \sum_{s=2}^k \mathbf{A}_{(s)}^{(k)} \mathbf{\Gamma}(s - 1) \mathbf{D}(s - 1) - \mathbf{\Gamma}(k) \mathbf{D}(k) ,$$

that is exactly (3), as we wanted.

## B Stationarity and Ergodicity: Concepts

Let  $(\Omega, \mathcal{F}, \mathbb{P})$  be a probability spaces. Let also the function  $T : \Omega \rightarrow \Omega$  be a measurable transformation of  $\Omega$  into itself.

**Definition 2** (Measure-Preserving). A measurable transformation  $T : \Omega \rightarrow \Omega$  into  $\Omega$  is measure-preserving if, for every  $B$  in  $\mathcal{F}$ ,  $\mathbb{P}(TB) = \mathbb{P}(B)$ .

**Definition 3** (Stationarity). A random sequence  $\omega \in \Omega$  is stationary if the shift operator is measure preserving.

**Definition 4** (Invariant set). If  $T$  is a measure-preserving transformation,  $B \in \mathcal{F}$  is an invariant set if  $TB = B$ , or equivalently  $\mathbb{P}[(B \setminus TB) \cup (TB \setminus B)] = 0$ .

**Definition 5** (Ergodicity). A measure-preserving transformation  $T$  is ergodic, if, given any invariant set  $B \in \mathcal{F}$ , it holds  $\mathbb{P}(B) = 0$  or  $\mathbb{P}(B) = 1$ .

**Definition 6** (Mixing). A measure-preserving transformation  $T$  is mixing if, for all  $B$  and  $C$  in  $\mathcal{F}$ ,

$$\lim_{n \rightarrow \infty} \mathbb{P}(B \cap T^n C) = \mathbb{P}(B)\mathbb{P}(C).$$

Note that when a random sequence is said to be ergodic tout court, it means that the shift operator is ergodic. Similarly when a random sequence is said to be mixing (or mixing in the ergodic-theoretic sense), it means that the shift operator is mixing.

## C Proof of Lemma 1

In this Appendix we prove lemma 1, reported in the following for reader convenience.

**Lemma.** (*Windowing a Markov chain*) Let  $\Phi = \{\Phi_n, n \in \mathbb{N}\}$  be an irreducible, aperiodic and stationary Markov chain. Consider the stochastic process  $\Psi = \{\Psi_n, n \in \mathbb{N}\}$ , where  $\Psi_n = (\Phi_n, \Phi_{n+1}, \dots, \Phi_{n+h-1})$  with  $h$  a positive integer.  $\Psi$  is also an irreducible aperiodic stationary Markov chain.

*Proof.* First of all it is evident that  $\Psi$  is also a Markov chain, whose states are possible  $h$ -uples of states of  $\Phi$ , e.g.  $(s_1, s_2, \dots, s_h)$ . The transition probabilities could be calculated starting from those of  $\Phi$ . Stationarity of  $\Psi$  easily follows from the stationarity of  $\Phi$ .  $\Psi$  is also irreducible because  $\Phi$  is irreducible. In fact, given two states  $s' = (s_1, s_2, \dots, s_h)$  and  $t' = (t_1, t_2, \dots, t_h)$ , for the irreducibility of  $\Phi$ , it exists  $n_0$ , such that the chain  $\Psi$  moves from  $s'$  to a state  $u' = (u_1, u_2, \dots, u_{h-1}, t_1)$  after  $n_0$  steps, and then it is possible to move from  $s_h$  to  $t_1$ .  $t'$  is a state of  $\Psi$  and therefore it is also a valid sequence of state transitions for  $\Phi$ , consequently in  $h-1$  time steps,  $\Phi$  can move from  $t_1$  to  $t_h$  going through  $t_2, \dots, t_{h-1}$  and  $\Psi$  can move from  $u'$  to  $t'$ . In conclusion in  $n_0 + h - 1$  steps,  $\Psi$  can move from  $s'$  to  $t'$ .

Aperiodicity requires a more detailed discussion. Given a possible state  $s' = (s_1, s_2, \dots, s_h)$ , we want to prove that the greatest common divisor of the possible time steps after which the chain  $\Psi$  can return in  $s'$  is equal to 1. Note that even if  $\Phi$  had the property that it is possible to directly move from each state to itself, for a state  $s'$  with  $s_i \neq s_1$  for some  $i = 2, \dots, h$ , at least  $h$  steps are required to return to that state. Consider the minimum number  $k_0$  of time steps after which the chain  $\Phi$  can move from  $s_h$  to  $s_1$  (again  $k_0$  exists because  $\Phi$  is irreducible). Consider then the increasing sequence of all the possible time steps  $(k_1, k_2, \dots)$  after which it is possible to return in  $s_1$ . Observe that also  $2k_i$  belongs to this sequence. It is clear that  $\Psi$  can return in  $s'$  after  $(k_0 + k_1 + h - 1, k_0 + k_1 + h - 1, \dots)$ . Let us denote  $g$  the greatest common divisor of this sequence of numbers, we have that for each  $i > 0$   $(k_0 + k_i + h - 1) \bmod g = 0$ . In particular also  $(k_0 + 2k_i + h - 1) \bmod g = 0$ , and it follows that  $(k_1) \bmod g = 0$ . This implies that  $g$  is also a divisor of the sequence  $(k_1, k_2, \dots)$ . Since  $\Phi$  is aperiodic, it follows that  $g = 1$ . This concludes the proof that  $\Psi$  is also aperiodic.  $\square$

## D Proof of Equation (8)

In this Appendix we prove equation (8), namely

$$\left| \left[ \mathbf{A}_{(s)}^{(k)} \right]_{uv} - \frac{1}{M} \right| \leq 2\tau_1 \left( \mathbf{A}_{(s)}^{(k)} \right) .$$

*Proof.* First we observe that a geometric bound holds also for the difference between any two elements on the same column and different rows. In fact:

$$\left| \left[ \mathbf{A}_{(s)}^{(k)} \right]_{u,v} - \left[ \mathbf{A}_{(s)}^{(k)} \right]_{w,v} \right| \leq 2\tau_1 \left( \mathbf{A}_{(s)}^{(k)} \right) .$$

We call  $\epsilon$  the right side of the above expression that we can rewrite as:

$$\left[ \mathbf{A}_{(s)}^{(k)} \right]_{u,v} - \epsilon \leq \left[ \mathbf{A}_{(s)}^{(k)} \right]_{w,v} \leq \left[ \mathbf{A}_{(s)}^{(k)} \right]_{u,v} + \epsilon ,$$

therefore

$$\begin{aligned} & \sum_{w=1}^M \left[ \mathbf{A}_{k+1} \right]_{z,w} \left( \left[ \mathbf{A}_{(s)}^{(k)} \right]_{u,v} - \epsilon \right) \leq \\ & \leq \sum_{w=1}^M \left[ \mathbf{A}_{k+1} \right]_{z,w} \left[ \mathbf{A}_{(s)}^{(k)} \right]_{w,v} \leq \\ & \leq \sum_{w=1}^M \left[ \mathbf{A}_{k+1} \right]_{z,w} \left( \left[ \mathbf{A}_{(s)}^{(k)} \right]_{u,v} + \epsilon \right) , \end{aligned}$$

which is equal to

$$\left[ \mathbf{A}_{(s)}^{(k)} \right]_{u,v} - \epsilon \leq \left[ \mathbf{A}_{(s)}^{(k+1)} \right]_{z,v} \leq \left[ \mathbf{A}_{(s)}^{(k+1)} \right]_{u,v} + \epsilon .$$

By induction:

$$\left[ \mathbf{A}_{(s)}^{(k)} \right]_{u,v} - \epsilon \leq \left[ \mathbf{A}_{(s)}^{(k+r)} \right]_{z,v} \leq \left[ \mathbf{A}_{(s)}^{(k+1)} \right]_{u,v} + \epsilon ,$$

and letting  $r$  go to infinity

$$\left[ \mathbf{A}_{(s)}^{(k)} \right]_{u,v} - \epsilon \leq \frac{1}{M} \leq \left[ \mathbf{A}_{(s)}^{(k+1)} \right]_{u,v} + \epsilon ,$$

i.e.,

$$\left| \left[ \mathbf{A}_{(s)}^{(k)} \right]_{u,v} - \frac{1}{M} \right| \leq \epsilon ,$$

and being that this inequality is true for all  $u$  and all  $v$ , a geometric bound can be derived also for  $\left\| \mathbf{A}_{(s)}^{(k)} - \mathbf{J} \right\|_{\max}$ .  $\square$

## E Proof of Proposition 4

In this Appendix we prove proposition 4, reported in the following for reader convenience. Further we also prove a key result that allows us to write equation (10). Since we proved proposition 3, the results here presented can be viewed as easily extensions to the case of ergodic and stationary process of Theorem 1 in [3]. Let us first recalling the following lemma. From [16]:

**Lemma 2.** *Let  $0 < \beta < 1$  and let  $\{\alpha(k)\}_{k \geq 0}$  be a positive scalar sequence. Assume that  $\lim_{k \rightarrow +\infty} \alpha(k) = 0$ . Then*

$$\lim_{k \rightarrow +\infty} \sum_{l=0}^k \beta^{k-l} \alpha(l) = 0 .$$

*In addition, if  $\sum_{k=1}^{\infty} \alpha(k) < \infty$ , then*

$$\sum_{k=1}^{\infty} \sum_{l=0}^k \beta^{k-l} \alpha(l) < \infty .$$

Using lemma 2 we can prove that

**Proposition** (Convergence of Agent Estimates). *Let assumption 1 holds,  $\mathbf{x}^i(k)$  be generated according to (1) for all  $i \in \{1, \dots, M\}$ ,  $k \geq 1$  and  $\mathbf{y}(k)$  be generated according to (4) for all  $k \geq 1$ . Then*

$$\lim_{k \rightarrow +\infty} \|\mathbf{y}(k) - \mathbf{x}^i(k)\|_2 = 0 \quad \text{a.s., for all } i .$$

*Proof.* Iterating equations 1 and 4 we can write

$$\begin{aligned}
& \|\mathbf{y}(k) - \mathbf{x}^i(k)\|_2 = \\
& = \left\| \frac{1}{M} \sum_{j=1}^M \mathbf{x}^j(1) - \sum_{j=1}^M \left[ \mathbf{A}_{(1)}^{(k-1)} \right]_{ij} \mathbf{x}^j(1) + \right. \\
& - \frac{1}{M} \sum_{s=1}^{k-1} \sum_{j=1}^M \gamma_j(s) \mathbf{d}^j(s) + \\
& \left. + \sum_{s=1}^{k-2} \sum_{j=1}^M \left[ \mathbf{A}_{(s+1)}^{(k-1)} \right]_{ij} \gamma_j(s) \mathbf{d}^j(s) + \gamma_i(k-1) \mathbf{d}^i(k-1) \right\|_2 \leq \\
& \leq \sum_{j=1}^M \|\mathbf{x}^j(1)\|_2 \left\| \frac{1}{M} - \left[ \mathbf{A}_{(1)}^{(k-1)} \right]_{ij} \right\|_2 + \\
& + \sum_{s=1}^{k-2} \sum_{j=1}^M \|\gamma_j(s)\|_2 \left\| \frac{1}{M} - \left[ \mathbf{A}_{(s+1)}^{(k-1)} \right]_{ij} \right\|_2 \|\mathbf{d}^j(s)\|_2 + \\
& + \frac{1}{M} \sum_{j=1}^M \|\gamma_j(k-1)\|_2 \|\mathbf{d}^j(k-1)\|_2 + \\
& + \|\gamma_i(k-1)\|_2 \|\mathbf{d}^i(k-1)\|_2 \leq \\
& \leq \sum_{j=1}^M \|\mathbf{x}^j(1)\|_2 b(k-1, 1) + L \sum_{j=1}^M \sum_{s=1}^{k-2} \gamma_j(s) b(k-1, s+1) + \\
& + L \left( \frac{1}{M} \sum_{j=1}^M \gamma_j(k-1) + \gamma_i(k-1) \right), \tag{15}
\end{aligned}$$

where the last inequality follows from the bounded sub-gradient assumption and defining the quantity

$$b(k, s) \stackrel{\text{def}}{=} \max_{i,j} \left| \left[ \mathbf{A}_{(s)}^{(k)} \right]_{ij} - \frac{1}{M} \right| \quad \text{for all } k \geq s .$$

Immediately we note that the last term in the right hand side (rhs) of equation (15) goes to zero as  $k$  goes to infinity, since by assumption  $\lim_{k \rightarrow \infty} \gamma_j(k) = 0$  for all  $j$ . From the proof of Proposition 3 we have that for almost all the sequences  $\{\mathbf{A}(k)\}_{k \geq 1}$

$$b(k, s) \leq C\beta^{k-s} \quad \text{for all } k \geq s, \tag{16}$$

where  $C > 0$  and  $0 < \beta < 1$  (with  $C$  in general depending of the considered sequence). Therefore for almost all the sequences also the first term in the rhs of (15) goes to zero increasing  $k$ . Finally, we if (16) holds, Lemma 2 applies and  $\lim_{k \rightarrow +\infty} \sum_{s=1}^{k-2} \gamma_j(s) b(k-1, s+1) = 0$  for all  $j$ . Thus, for almost all the sequences, for all  $i$  we have that

$$0 \leq \lim_{k \rightarrow +\infty} \|\mathbf{y}(k) - \mathbf{x}^i(k)\|_2 \leq 0,$$

proving the desired result.  $\square$

Lemma 2 is also a key ingredient to prove the following

**Proposition** (Convergence of Agent Estimates). *Let assumption 1 and ?? hold,  $\mathbf{x}^i(k)$  be generated according to (1) for all  $i \in \{1, \dots, M\}$ ,  $k \geq 1$  and  $\mathbf{y}(k)$  be generated according to (4) for all  $k \geq 1$ . Furthermore we assume that all the updates are synchronous, i.e.,  $\gamma_i(s) = \gamma(s)$  for all  $i$ ,  $s \geq 1$ . Then*

$$\sum_{k=1}^{+\infty} \gamma(k) \|\mathbf{y}(k) - \mathbf{x}^i(k)\|_2 < \infty \quad \text{a.s., for all } i .$$

*Proof.* From equation (15) in the above proof and using  $\gamma_i(s) = \gamma(s)$  for all  $i$ , we have that

$$\begin{aligned} \|\mathbf{y}(k) - \mathbf{x}^i(k)\|_2 &\leq \sum_{j=1}^M \|\mathbf{x}^j(1)\|_2 b(k-1, 1) + \\ &+ LM \sum_{s=1}^{k-2} \gamma(s) b(k-1, s+1) + 2L\gamma(k-1) , \end{aligned}$$

that, defining  $C_1 = \max \left\{ \sum_{j=1}^M \|\mathbf{x}^j(1)\|_2, ML \right\}$  and  $\gamma(0) = 1$ , can be rewritten as

$$\|\mathbf{y}(k) - \mathbf{x}^i(k)\|_2 \leq C_1 \sum_{s=0}^{k-2} \gamma(s) b(k-1, s+1) + 2L\gamma(k-1) ,$$

multiply at both sides for  $\gamma(k)$  we have

$$\begin{aligned} \gamma(k) \|\mathbf{y}(k) - \mathbf{x}^i(k)\|_2 &\leq C_1 \sum_{s=0}^{k-2} \gamma(k) \gamma(s) b(k-1, s+1) + \\ &+ 2L\gamma(k) \gamma(k-1) . \end{aligned}$$

Recalling the result in theorem 3 we have that for almost all the sequences  $\{\mathbf{A}(k)\}_{k \geq 1}$ ,  $b(k, s) \leq C_2 \beta^{k-s}$  for all  $k \geq s$ , where  $C_2 > 0$  and  $0 < \beta < 1$  (with  $C_2$  in general depending of the considered sequence). Calling  $C$  the product  $C_1 C_2$ , we have for almost all the sequences

$$\begin{aligned} \gamma(k) \|\mathbf{y}(k) - \mathbf{x}^i(k)\|_2 &\leq C \sum_{s=0}^{k-2} \gamma(k) \gamma(s) \beta^{k-s-2} + \\ &+ 2L\gamma(k) \gamma(k-1) . \end{aligned}$$

Noting that  $\gamma(k) \gamma(s) \leq \gamma(k)^2 + \gamma(s)^2$  and  $2\gamma(k) \gamma(k-1) \leq \gamma(k)^2 + \gamma(k-1)^2$  we obtain for almost all the sequences

$$\begin{aligned} \gamma(k) \|\mathbf{y}(k) - \mathbf{x}^i(k)\|_2 &\leq C \gamma(k)^2 \sum_{s=0}^{k-2} \beta^{k-s-2} + \\ &+ C \sum_{s=0}^{k-2} \gamma(s)^2 \beta^{k-s-2} + L\gamma(k)^2 + L\gamma(k-1)^2 \leq \\ &\frac{C \gamma(k)^2}{1-\beta} + C \sum_{s=0}^{k-2} \gamma(s)^2 \beta^{k-s-2} + L\gamma(k)^2 + L\gamma(k-1)^2 , \end{aligned}$$

and summing this last inequality over all the  $k$

$$\begin{aligned}
\sum_{k=1}^{+\infty} \gamma(k) \|\mathbf{y}(k) - \mathbf{x}^i(k)\|_2 &\leq \frac{C}{1-\beta} \sum_{k=1}^{+\infty} \gamma(k)^2 + \\
C \sum_{k=1}^{+\infty} \sum_{s=0}^{k-2} \gamma(s)^2 \beta^{k-s-2} &+ L \sum_{k=1}^{+\infty} \gamma(k)^2 + \\
+ L \sum_{k=1}^{+\infty} \gamma(k-1)^2 &\quad \text{a. s. .} \tag{17}
\end{aligned}$$

The desired result follows straightforwardly since: 1) the first, third and fourth term at the right hand side of equation (17) are summable by assumption on the chosen step-sizes, and 2) the second term of (17) is summable because lemma [16] applies.  $\square$

## F Derivation of the Utility Function in Equation (14)

In this Appendix we briefly review the mathematical steps of the analysis carried on in [5], under the assumptions done in Section 6 and with the formalism introduced therein. The overall objective of this Appendix is to formally derive equation (14).

For a given time  $\bar{t}$  and node  $i \in \mathcal{N}$  we define the process  $B_i(\bar{t}, t) \subseteq \mathcal{N}$  as the set containing all the nodes  $j$  such that, if a message is given to them at time  $\bar{t} - t$ , it can reach user  $i$  in two hops by time  $\bar{t}$ . For each node pair  $(i, j)$ ,  $j \neq i, j, i \in \mathcal{N}$ , we define also  $s_{ij}(\bar{t})$  as

$$s_{ij}(\bar{t}) \stackrel{\text{def}}{=} \inf_{t \geq 0} \{t : j \in B_i(\bar{t}, t)\} .$$

$\bar{t} - s_{ij}(\bar{t})$  indicates the minimum amount of time required to transfer information from node  $j$  to node  $i$  at time  $\bar{t}$  through file sharing. If we assume that the intermeeting process time between the node pair  $(i, j)$  is exponentially distributed with parameter  $\lambda_{ij}$  (i.e., Poisson meeting process) we have that in the case of two-hops protocol also  $s_{ij}(\bar{t})$  is exponentially distributed with parameter  $\lambda_{ij}$ .

Let then  $Y_i^{SP}(\bar{t})$  be, at time  $\bar{t}$ , the elapsed time since user  $i$  downloaded content directly from the SP, i.e.,  $Y_i^{SP}(\bar{t}) = \bar{t} - t_i^B S(\bar{t})$ . Note that the random variable  $Y_i^{SP}(\bar{t})$  for all  $i \in \mathcal{N}$  is exponentially distributed with parameter  $x_i$ , since the SP transfer updates directly to node  $i$  with rate  $x_i$  and in stationary conditions the forward process and the backward one have the same statistic, see e.g., [17].

Lemma 1 of [5] states that

$$Y_i(\bar{t}) = \min_{j \in \mathcal{N}} \{s_{ij}(\bar{t}) + Y_j^{SP}(\bar{t} - s_{ij}(\bar{t}))\} , \quad (18)$$

therefore, in our case,  $Y_i(\bar{t})$  is the minimum over  $M$  independent random variables. One of this random variables is exponentially distributed with parameter  $x_i$  (i.e.,  $s_{ij}(\bar{t}) + Y_j^{SP}(\bar{t} - s_{ij}(\bar{t}))$  with  $j = i$ ) and takes in account of the directed updates of  $i$  from the PS. The remaining  $M - 1$  random variables are in turn sum of two independent exponential random variables: one distributed with parameter  $\lambda_{ij}$  and the second with parameter  $x_j$ . Each of these latter  $M - 1$  variables takes in account the update of  $i$  in two hops from the PS through a given relay node  $j \neq i$ . For the independence of  $s_{ij}$  and  $Y_j^{SP}$ , and assuming in general  $\lambda_{ij} \neq x_j$ , we have that

$$p_{s_{ij}+Y_j^{SP}}(y) = \lambda_{ij} x_j \left[ \frac{e^{-\lambda_{ij}y} - e^{-x_j y}}{x_j - \lambda_{ij}} \right] \quad y > 0 ,$$

whence

$$P[s_{ij} + Y_j^{SP} > y] = \frac{x_j e^{-\lambda_{ij}y} - \lambda_{ij} e^{-x_j y}}{x_j - \lambda_{ij}} ,$$

therefore

$$P[Y_i > y] = \left[ \prod_{j \sim i} \frac{x_j e^{-\lambda_{ij}y} - \lambda_{ij} e^{-x_j y}}{x_j - \lambda_{ij}} \right] e^{-x_i y} .$$

In the case of utility  $u(Y_i) = \chi\{Y_i \leq \tau\}$  we have that

$$\begin{aligned} f_i(\mathbf{x}) &= E_{\mathbf{x}}[u_i(Y_i)] = P[Y_i \leq \tau] = \\ &= 1 - \left[ \prod_{j \sim i} \frac{x_j e^{-\lambda_{ij}\tau} - \lambda_{ij} e^{-x_j\tau}}{x_j - \lambda_{ij}} \right] e^{-x_i\tau}, \end{aligned}$$

thus

$$\begin{aligned} f(\mathbf{x}) &= \sum_{i=1}^M f_i(\mathbf{x}) = \\ &= M - \sum_{i=1}^M \left[ \prod_{j \sim i} \frac{x_j e^{-\lambda_{ij}\tau} - \lambda_{ij} e^{-x_j\tau}}{x_j - \lambda_{ij}} \right] e^{-x_i\tau}. \end{aligned}$$

From equation (14) is straightforward to compute the components of the gradient vector we need to implement the distributed sub-gradient method, for all  $i$ :

$$\frac{\partial f_i(\mathbf{x})}{\partial x_i} = \tau \left[ \prod_{j \sim i} \frac{x_j e^{-\lambda_{ij}\tau} - \lambda_{ij} e^{-x_j\tau}}{x_j - \lambda_{ij}} \right] e^{-x_i\tau},$$

and, for  $z \neq i$

$$\begin{aligned} \frac{\partial f_i(\mathbf{x})}{\partial x_z} &= \left[ \prod_{j \sim i, z} \frac{x_j e^{-\lambda_{ij}\tau} - \lambda_{ij} e^{-x_j\tau}}{x_j - \lambda_{ij}} \right] e^{-x_i\tau} \cdot \\ &\quad \frac{\lambda_{iz} \{e^{-\lambda_{iz}\tau} + [\tau(\lambda_{iz} - x_z) - 1] e^{-x_z\tau}\}}{(x_z - \lambda_{iz})^2}. \end{aligned}$$

## References

- [1] A. Nedic and A. Ozdaglar, “Distributed Subgradient Methods for Multi-Agent Optimization,” *Automatic Control, IEEE Transactions on*, vol. 54, no. 1, pp. 48–61, Jan. 2009.
- [2] J. Tsitsiklis, D. Bertsekas, and M. Athans, “Distributed Asynchronous Deterministic and Stochastic Gradient Optimization Algorithms,” *Automatic Control, IEEE Transactions on*, vol. 31, no. 9, pp. 803–812, Sep. 1986.
- [3] I. Lobel and A. Ozdaglar, “Distributed Subgradient Methods for Convex Optimization over Random Networks,” *Automatic Control, IEEE Transactions on - Accepted for publication*, May 2010.
- [4] A. Tahbaz-Salehi and A. Jadbabaie, “Consensus Over Ergodic Stationary Graph Processes,” *Automatic Control, IEEE Transactions on*, vol. 55, no. 1, pp. 225 – 230, Jan. 2010.
- [5] S. Ioannidis, A. Chaintreau, and L. Massoulie, “Optimal and Scalable Distribution of Content Updates over a Mobile Social Network,” in *IEEE INFOCOM 2009*, Rio de Janeiro, Brazil, Apr. 2009, pp. 1422–1430.
- [6] S. Burleigh, A. Hooke, L. Torgerson, K. Fall, V. Cerf, B. Durst, K. Scott, and H. Weiss, “Delay-tolerant Networking: an Approach to Interplanetary Internet,” *Communications Magazine, IEEE*, vol. 41, no. 6, pp. 128–136, Jun. 2003.
- [7] L. Pelusi, A. Passarella, and M. Conti, “Opportunistic Networking: Data Forwarding in Disconnected Mobile Ad Hoc Networks,” *Communications Magazine, IEEE*, vol. 44, no. 11, pp. 134–141, Nov. 2006.
- [8] G. Neglia, G. Reina, and S. Alouf, “Distributed gradient optimization for epidemic routing: A preliminary evaluation,” in *Wireless Days (WD), 2009 2nd IFIP*, Paris, FR, Dec. 2009.
- [9] A. N. Shiryaev, *Probability (2nd ed.)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 1995, translator-Boas, R. P.
- [10] R. C. Bradley, “Basic Properties of Strong Mixing Conditions. A Survey and Some Open Questions,” *Probability Surveys*, 2005.
- [11] E. Seneta, *Non-negative Matrices and Markov Chains*. Springer, New York 2<sup>nd</sup> edition, 1981.
- [12] J. C. Oxtoby, “Ergodic sets,” *Bulletin of the American Mathematical Society*, vol. 58, pp. 116–136, 1952.
- [13] P. Denantes, “Performance of averaging algorithms in time-varying networks,” Ph.D. dissertation, Master’s thesis, TUM/EPFL, 2007.
- [14] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.
- [15] R. Groenevelt, P. Nain, and G. Koole, “Message Delay in Mobile Ad Hoc Networks,” in *Proc. of PERFORMANCE EVALUATION*, Juan-les-Pins, France, Oct. 2005.

- [16] A. Nedic, A. Ozdaglar, and P. Parrilo, “Constrained Consensus and Optimization in Multi-Agent Networks,” *Automatic Control, IEEE Transactions on*, Feb. 2010, accepted for publication.
- [17] S.M. Ross, *Introduction to Probability Models*. Elsevier, Academic Press, 2007.

## Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Distributed Sub-gradient Method's Overview</b>	<b>4</b>
<b>3</b>	<b>Application to Optimization in DTNs: Motivation</b>	<b>7</b>
<b>4</b>	<b>Extension to more General Mobility Models</b>	<b>11</b>
<b>5</b>	<b>Asynchronous Updates</b>	<b>15</b>
<b>6</b>	<b>Application to Optimization in DTNs: a Case Study</b>	<b>17</b>
<b>7</b>	<b>Acknowledgements</b>	<b>20</b>
<b>8</b>	<b>Conclusion</b>	<b>21</b>
<b>A</b>	<b>Derivation of Equation (3)</b>	<b>22</b>
<b>B</b>	<b>Stationarity and Ergodicity: Concepts</b>	<b>24</b>
<b>C</b>	<b>Proof of Lemma 1</b>	<b>25</b>
<b>D</b>	<b>Proof of Equation (8)</b>	<b>26</b>
<b>E</b>	<b>Proof of Proposition 4</b>	<b>27</b>
<b>F</b>	<b>Derivation of the Utility Function in Equation (14)</b>	<b>31</b>



---

Centre de recherche INRIA Sophia Antipolis – Méditerranée  
2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Centre de recherche INRIA Bordeaux – Sud Ouest : Domaine Universitaire - 351, cours de la Libération - 33405 Talence Cedex  
Centre de recherche INRIA Grenoble – Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier  
Centre de recherche INRIA Lille – Nord Europe : Parc Scientifique de la Haute Borne - 40, avenue Halley - 59650 Villeneuve d'Ascq  
Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique  
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex  
Centre de recherche INRIA Paris – Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex  
Centre de recherche INRIA Rennes – Bretagne Atlantique : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex  
Centre de recherche INRIA Saclay – Île-de-France : Parc Orsay Université - ZAC des Vignes : 4, rue Jacques Monod - 91893 Orsay Cedex

---

Éditeur  
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)  
<http://www.inria.fr>  
ISSN 0249-6399