Modelling Data Dissemination in Opportunistic Networks *

Chiara Boldrini
IIT-CNR
Via Moruzzi, 1
56124 Pisa, Italy
c.boldrini@iit.cnr.it

Marco Conti
IIT-CNR
Via Moruzzi, 1
56124 Pisa, Italy
m.conti@iit.cnr.it

Andrea Passarella
IIT-CNR
Via Moruzzi, 1
56124 Pisa, Italy
a.passarella@iit.cnr.it

ABSTRACT

In opportunistic networks data dissemination is an important, although not widely explored, topic. Since opportunistic networks topologies are very challenged and unstable, data-centric approaches are an interesting direction to pursue. Data should be proactively and cooperatively disseminated from sources towards possibly interested receivers, as sources and receivers might not be aware of each other, and never get in touch directly. In this paper we consider a utility-based cooperative data dissemination system in which the utility of data is defined based on the social relationships between users. Specifically, we study the performance of this system through an analytical model. Our model allows us to *completely* characterise the data dissemination process, as it describes both its stationary and transient regimes. After validating the model, we study the system's behaviour with respect to key parameters such as the definition of the data utility function, the initial data allocation on nodes, the number of users in the system, and the data popularity.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Wireless Communication*

General Terms

Performance

Keywords

opportunistic networks, data dissemination, modelling

1. INTRODUCTION

Opportunistic networks [8] are challenged mobile (multi-hop) ad hoc networks characterised by prolonged disconnections, partitions, unpredictable and unstable topologies. With respect to the legacy MANET paradigm, all these features are seen as properties of the network, instead of exceptions to cope with. This results in a paradigm shift for the design of network services. Continuous multi-hop paths may seldom be available between communicating end points. Network protocols should be opportunistic, in the sense that they should opportunistically exploit any contact[1] between nodes to bring data closer to possibly interested users. Users mobility should be exploited to bridge partitions and move data. Therefore, knowledge about the social behaviour of users is a key piece of information, as it complements the unreliable knowledge about the network topology. As nodes might never be aware of each other, *data*-centric networking is an interesting direction to explore, besides conventional *topology*-centric networking.

It is clear that in such an environment data-dissemination systems are very useful. According to a data-centric paradigm, such services move (replicate, distribute) *data objects*, such as MP3 files, advertisements, . . . , within the network, trying to understand the regions where there are interested users. Clearly, as opportunistic networks are formed by resource-limited mobile devices, data dissemination should also be aware of resource consumption (e.g., memory, buffer, energy), and trade performance for resource usage.

As we discuss in Section 1.1, traditional schemes for data dissemination in mobile networks assume rather stable or predictable network topologies, and are thus unsuitable for opportunistic networks. In this paper we consider a utility-based data-dissemination scheme proposed in [3][2], and characterise its performance through an analytical model. This data dissemination scheme (summarised in Section 2) adopts a utility-based approach. The main idea is that nodes gather other users' interests (i.e., the type of data objects they are interested into) during contacts, and estimate the availability of these data objects in the network. They use this information to compute utility values for data objects they "see" (i.e., objects that are available on encountered nodes), and to decide what to fetch and store locally. This decision is also based on the cost of the data objects in terms of resource consumption.

The main goal of the model we present in Section 3 is

*This work was partially funded by the European Commission under the HAGGLE (027918) and SOCIALNETS (217141) FET Projects.

CHANTS'08, September 15, 2008, San Francisco, California, USA.

[1]By contact we mean a one-hop communication opportunity.

[2]The work in [3] is also available online as a Technical Report, at `http://bruno1.iit.cnr.it/~chiara/dataDisseminationTR.pdf`.

understanding if the data-dissemination system reaches stationary regimes, and to characterise their properties. We propose a Markovian model of the data distribution process resulting from our dissemination system. The main result of our analysis (derived in Section 3.4) is that the data distribution process always converges to one of two possible stationary regimes. Either nodes store only data objects they are interested into, or oscillate between storing only data objects they are interested into and data objects other users are interested into. The analysis also characterises the transient behaviour of the system, and thus permits to compute the probabilities of reaching either stationary regimes starting from arbitrary initial conditions. As discussed in Section 3.1, this characterisation is the building block to study typical performance figures such as hit rate, fairness, network overhead.

Finally, we validate the model through simulations, and use it to study the data distribution process with respect to several parameters, such as the definition of the data utility function, the initial data objects' allocation, the number of users in the system, and the data objects' popularity. This analysis allows us to achieve deep insights into the system's behaviour, and permits fine control on its evolution. For example, it permits to design the data utility function in order to achieve the desired stationary regime, also in relation with the expected number of users of the system and the popularity of the data objects. The results we show allow us to also understand which initial distribution of data objects on nodes is preferred to achieve a desired stationary regime, or, if the initial distribution cannot be controlled, how to set the other system's parameters to control the system's evolution.

1.1 Related work

Utility-based approaches to data distribution have been studied for legacy Internet environments [1], as well as for single-hop mobile environments [9]. This body of work is the main inspiration for our utility based framework. However, these solutions are not directly applicable to opportunistic networks. Indeed, they propose non-cooperative solutions, designed either for wired Internet or single-hop wireless environments, in which *proactive* dissemination of data objects is not necessary. Similarly, strategies for conventional MANETs [10] are not applicable, as they build upon the assumption of connected topologies with rather stable paths.

Distributed systems based on gossiping [4] are also related to our system. Indeed, most of the networking systems for opportunistic networks can be seen as gossiping schemes. Specifically, in our system the choice of which data objects to fetch from encountered nodes can be seen as a gossiping operation. The main novelty of our system is using context information describing users' social relationships to choose the set of nodes where to replicate data objects, instead of simple probabilistic policies.

Recently, data dissemination for opportunistic networks has been targeted in the PodNet project [7]. With respect to PodNet, our system provides a general utility-based framework, while the work in [7] proposes heuristics for data dissemination services. A utility-based system is also proposed in [2], which, however, considers routing protocols instead of data dissemination services. This makes the definition of utility functions and the related analysis of the system totally different. Furthermore, our system is completely structure-less, and thus differentiates itself from data dissemination schemes requiring some level of knowledge about the network structure, such as multicast trees [12], or broker overlay networks [11].

Finally, this paper differentiates from our previous work in [3], as in that paper we provide full specification of the data dissemination algorithms and a preliminary simulation evaluation, while here we develop an analytical model characterising the transient and stationary regimes of the data dissemination process.

2. OVERVIEW OF THE DATA DISSEMINATION SYSTEM

In this section we firstly describe our reference application scenario. Then, we summarise the utility-based data dissemination system we use to support it.

2.1 Application scenario for data dissemination in opportunistic networks

The application scenario we target is similar to the one used in PodNet [7], named "podcasting for ad hoc networks". As in the typical opportunistic networking paradigm, we consider a number of mobile users whose devices cannot be encompassed by a conventional MANET. Instead, communication is achieved by opportunistically exploiting pair-wise contacts between users to exchange messages, and bringing them towards eventual destinations. Sporadic contacts of users with point of access to the Internet (e.g., WiFi hotspots) are possible although not necessary. In podcasting applications, data objects (e.g., MP3 files, advertisements, software updates, ...) are organised in different *channels* to which users can subscribe. We assume that the channel(s) of a data object is decided by the source of the object at the generation time. Data objects might be generated from within the Internet, and "enter" the opportunistic network upon sporadic contacts of users with Internet Access Points. Or, data objects may be generated dynamically by the users of the opportunistic network according to the Web 2.0 model (e.g., users may wish to share pictures taken with their mobile phones). The data dissemination system summarised in Section 2.2 is responsible for managing subscriptions, and bringing data objects to subscribed users.

2.2 Utility-based data dissemination system

As discussed in Section 1.1, the network we consider does not permit to exploit structures such as multicast trees (as in [12]) or broker-based publish/subscribe overlays [11]. Instead, our data dissemination paradigm follows a gossiping-like approach.

Every data dissemination system must specify mechanisms for managing subscriptions and delivering data to subscribed users. As far as the former aspect, as will be clear in the following, our framework just requires that each node advertises the channels its user is subscribed to upon making contact with any other node. The framework does not need per-user subscription state, and thus unsubcriptions are not required.

Implementing data delivery requires several mechanisms, as follows. First of all, we assume that nodes that generate data objects persistently store them locally, so that, even if not replicated at all, they never disappear from the network. If data objects are generated in the Internet, they are stored at the Access Points that provide Internet access to

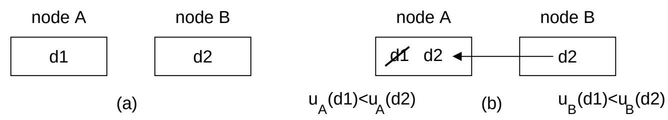

Figure 1: Example of replication.

the opportunistic network. In addition to that, nodes contribute a limited-size shared buffer (hereafter called cache) for data dissemination purposes. Following a gossiping-like approach, whenever two nodes make contact (i.e., "meet") they decide which data objects to fetch from the peer's cache, if any, and store fetched objects in the local cache. As will be clear in the following, this may result in replacing data objects stored locally. Note that, as decisions of the two peers are independent of each other, the objects fetched by one peer are not, in general, evicted from the other peer's cache, and are thus replicated. The policy used to select the data objects to fetch is hereafter referred to as *replication* policy, and is the core of our system. Users receive data objects of channels they are subscribed to when they meet another user storing such data objects in their cache.

As in the majority of gossiping protocols, the main challenge of our system is defining a replication policy that does not rely on precise global knowledge about the network state, but that nevertheless achieves a global performance target. Our framework can be customised towards maximising the hit rate, the per-user fairness, or minimise the network overhead, for example. Our system uses a utility-based framework to define the replication policy. When a node meets a peer, it computes the utility of each data object either stored locally or on the peer's cache, and identifies the set of data objects that can be stored in its cache (i.e., fit in the cache's size) and maximise the total utility of the cache. Figure 1 shows a trivial example in which the caches of nodes A and B can store just one data object. The utility of the object carried by B is the highest one for both nodes, thus node A replaces the object it is currently storing with the object available in B's cache (note that the utility of the same data object computed on different nodes can be different). More in general, the utility-based framework permits to consider several *resource constraints* at the same time (besides the cache size) such as, for example, the energy and bandwidth available for the data dissemination operations. According to the framework, upon a contact, each node solves a multi-constrained knapsack problem (MKP) to identify the optimal set of data objects that do not violate any constraint, and fetches those objects in the set that are on the peer's cache. Formally, the MKP has the following form:

$$\begin{cases} \max & \sum_k U_k x_k & \\ \text{s.t.} & \sum_k c_{jk} x_k \leq 1 & j = 1, \ldots, m \\ & x_k \in \{0,1\} & \forall k \end{cases}, \tag{1}$$

where k denotes the k-th object that the node could select (either stored locally or on the peer's cache), U_k its utility, c_{jk} the consumption of resource j related to fetching and storing object k, normalised to the maximum allowed consumption of that resource (i.e., $0 \leq c_{jk} \leq 1$ holds true), m the number of constrained resources, and $x_k \in \{0,1\}$ the MKP's variables. As long as the number of resources (m) is not too high (which is quite reasonable), the solution of the above MKP can be well approximated by fast on-line algorithms [5], making the framework suitable for mobile environments also from a computational standpoint.

The ability of such a data-dissemination system to achieve the global performance target depends on the definition of the function used to compute utility values. In our framework we define the utility function inspired by the literature on utility-based Web caching [1], which has been also exploited to define caching policies in infrastructure-based single-hop wireless systems [9]. The typical form of the utility function is the product of the access probability to the data object (p_{ac}) by a measure of the retrieval cost (c), normalised by the object's size (s). The rationale of this definition is that the utility of an object should be high if it is very popular and costly to be retrieved. Normalising by the size is usually just a way to have a simple, but accurate, approximate solution of the resulting knapsack problem. In our framework we use exactly the same kind of utility function, by defining the cost c as a strictly monotonically decreasing function (denoted as $f_c()$) of the object's availability in the network (hereafter referred to as p_{av}). Specifically, p_{av} is defined as the probability of finding the object in the cache of any node. Clearly, the higher p_{av}, the lower the cost to retrieve the object, the lesser its value. Different types of functions can be used for $f_c()$. For example, in [3] we have considered an exponential function $e^{-\lambda p_{av}}$, $\lambda > 0$.

In the Web caching literature, p_{ac} and c are computed with respect to the set of Web users accessing the cache. In our opportunistic networking scenario, the users of any node cache are i) the local user, and ii) the users of the other encountered nodes. The general form of the utility function takes this into consideration by defining multiple components, one for the local user and one for any social community the local user is in contact with. In this paper, we consider a simplified, yet significant, definition, by which the utility function is made up of two components only, one related to the local user ($u^{(l)}$), the other (the *social* component, $u^{(s)}$) aggregating the utility for all the other users:

$$U = u^{(l)} + u^{(s)} = p_{ac}^{(l)} f_c(p_{av}^{(l)}) + p_{ac}^{(s)} f_c(p_{av}^{(s)}). \tag{2}$$

In Equation 2: i) $p_{ac}^{(l)}$ represents the probability that the local user is interested into the data object; ii) $p_{ac}^{(s)}$ represents the probability that any encountered node is interested into the data object; iii) $p_{av}^{(l)}$ represents the probability that the local node "sees" the data object in the caches of any encountered node; and iv) $p_{av}^{(s)}$ represents the average probability (over encountered nodes), that those nodes "see" the data object in the caches of any node *they* encounter. Note that, according to this utility function, nodes do not necessarily store data objects the local user is interested into. As will be clear in the following, this might happen if such data objects are so spread in the network that it is extremely easy to find them on encountered nodes' caches. In this case, it is *more useful* to store less spread data objects, because this improves the overall utility of the system.

In our system, nodes do not need any global knowledge to estimate the p_{ac} and p_{av} parameters. Specifically, each node must only advertise, upon each contact, the set of channels the local user is subscribed to, and an index of the current content of the cache. As described in detail in [3], this information is sufficient to estimate p_{ac} and p_{av} related to both the local user (local component of U), and the other encountered users (social component of U).

Finally, we highlight two modifications of the above general framework introduced in [3] to reduce the per-node state associated with the data dissemination operations. Firstly, we assume that a user subscribed to a channel is interested into *all* the data objects of the channel. Therefore, the p_{ac} indexes (both local and social) of Equation 2, computed by a given node, are the same for all the data objects of the same channel. Secondly, the availability indexes are aggregated over all objects of the same channel. p_{av} thus represents the probability of finding data objects of a given channel on any node's cache, for the local user ($p_{av}^{(l)}$), or the encountered users ($p_{av}^{(s)}$). From these assumptions it follows that the utility of all the data objects of the same channel, computed by a given nod, is the same, i.e., the terms U_k in Equation 1 computed by a particular node are the same for all the data objects of the same channel. Thus, in the following, we talk about *channel* utility. Although these modifications result in approximate indices for the access probability and availability of individual data objects, this does not severely impact on the performance of our system, as shown in [3].

A side effect of these modifications (confirmed by simulation results in [3]) is the fact that nodes tend to store data objects of the most useful channel only, because, as soon as it becomes the most useful, nodes fetch data objects of that channel which replace objects of any other channel. We will exploit this property in Section 3.2.

3. ANALYSIS OF UTILITY-BASED DATA DISSEMINATION

3.1 Analysis target and results preview

The goal of the model we present hereafter is studying the process of distribution of data objects in the network when our data dissemination system is used. Specifically, we want to understand if the process reaches a stationary regime, and characterise its properties. This allows us to describe how data objects distribute on nodes in the stationary regime, and is therefore the key building block to determine the performance of the system with respect to a number of figures, such as the hit rate, the fairness, the network overhead, etc. Under the assumptions discussed in Section 3.2, it is sufficient to use a Markovian representation of the data distribution process, that describes how many nodes store data objects of any particular channel during the process evolution.

Clearly, the process evolution depends on the utility function. Therefore, in Section 3.3 we analyse its behaviour, depending on the status of the Markov chain. This is the basis for the core of our analysis, presented in Section 3.4, and specifically for Theorem 1, which is the main analytical result. Theorem 1 states that the data distribution process always reaches one among two alternative stationary regimes (corresponding to two stationary distributions of the Markov chain). In one of the regimes, each node stores only the data objects it is interested into (which corresponds to a greedy behaviour). In the other regime, each node oscillates between storing only data objects it is interested into, and storing only data objects it is not interested into. The Theorem also provides the conditions under which either configuration is reached, starting from arbitrary initial conditions. Thus, our analysis is also able to completely describe the transient regime of the process and, therefore, completely characterises the data dissemination process. These results are derived for generic utility functions in the form of Equation 2. We complete the analysis in Section 3.5, by specialising the results of Theorem 1 to the case of utilities with exponential and linear cost functions.

Theorem 1 has several implications. First of all, it permits to achieve full control on the evolution of the data dissemination system. Since it provides the parameters' values under which either stationary regime is reached, it enables tuning the system to reach any feasible target behaviour. Furthermore, it provides additional insights with respect to simulation results presented in [3], for example on the difference between pure greedy and cooperative (social) utility functions. In that paper we have shown that a social-oriented utility function achieves higher per-user fairness (in terms of hit rate) with respect to a greedy policy, at the cost of slightly increased traffic overhead. The reason of this lies in the fact that the social-oriented policy function can lead the system into the oscillating stationary regime (while the greedy policy does not). By making nodes alternatively storing data objects they are interested to and they are not, the system increases the availability of all data objects, thus improving fairness. On the other hand, the oscillations are also the reason of the additional traffic overhead.

3.2 Assumptions and Markov representation

We assume that time is slotted, and nodes compute utilities at the beginning of each time slot. Within one time slot, each node fills its cache (by exchanging data objects with encountered peers) with data objects of the channel identified as the most useful at the beginning of the slot. Based on the last property discussed in Section 2.2, we assume that, at any point in time, each node stores only data objects of the channel it considers as the most useful one (this assumption is backed up by simulation results in [3]). Finally, we assume that the mobility model is uniform (i.e., all nodes move according to the same statistical process), that the size of data objects is the same, and the size of the caches is the same across all nodes. Under these assumptions, we can model the evolution of the data distribution process with a Markov chain whose status is the vector $\mathbf{n} = (n_1, \ldots, n_C)$, where C is the number of channels, and n_i the number of nodes considering channel i as the most useful (and thus storing data objects of channel i). This Markov chain completely describes the data distribution process in the network. Specifically, since the chain is finite, stationary distributions always exist. Note that the stationary distribution of $\mathbf{n}$ is the key building block, as far as the data dissemination system is concerned, for performance indices such as the hit rate, fairness, etc. Also note that, as it focuses on the figure $\mathbf{n}$, the model does not depend on the nodes' cache size, and is valid for any cache size as long as each node can find enough data objects of the channel identified as the most useful to fill its cache.

In general, this Markovian model is rather complex to describe and solve due to the number of possible states, and the form of the transition probabilities. It is possible to simplify the analysis, by assuming that all the nodes have an exact view of the utility function's parameters p_{ac} and p_{av} (both for the local and for the social components). Note that this does not necessarily require global knowledge, but that each node estimates these parameters after meeting a set of peers whose caches and interests correctly reproduce the current status of the network. The exact achievement of this condition might not be granted in practice, and its approximation heavily depends on the underlying mobility process. In the following, we assume a mobility model that meets this requirement. Generalising our model to any mobility model is the main subject of future work.

Under these assumptions, we are able to completely describe both the transient and the stationary regimes of the process. Hereafter, we present the analysis in the case of 2 channels. The same methodology permits to deal with more general cases, as well. However, we consider the special case of 2 channels as the analysis is quite simple, and the results are intuitive. Even this rather simple case shows the deep level of understanding about the system's behaviour that can be achieved, and motivates us to use this modelling approach to more complex cases.

3.3 Analysis of the utility function

The state transitions of the Markov chain are clearly determined by the values of the utility function. Under our assumptions, the function's parameters used by each node can be described as follows. For the sake of explanation, let us focus on a user subscribed to channel j, and let us evaluate its utility parameters with respect to channel i. The local access probability to is $p_{ac}^{(l)} = \mathbf{1}_{\{j\}}(i)$ where $\mathbf{1}_{\{\cdot\}}(\cdot)$ is the standard indicator function. The social access probability is the probability of meeting a node subscribed to channel i. Under our assumptions, this is the probability that any given node subscribes to channel i (throughout referred to as z_i)[3]. Since we assume that nodes can compute exact p_{av} parameters, $p_{av}^{(l)}$ and $p_{av}^{(s)}$ are both equal to n_i/M, where M is the number of nodes in the system. Therefore, the utility of channel i computed by any node subscribed to j is

$$U_{ij} = (\mathbf{1}_{\{j\}}(i) + z_i) \cdot f_c\left(\frac{n_i}{M}\right). \quad (3)$$

The properties of the Markov chain can be analysed by exploiting the fundamental observation that all nodes subscribed to j store data objects of channel $\hat{\imath}$ such that:

$$\hat{\imath} = \arg\max_i \{U_{ij}\}. \quad (4)$$

In the 2-channel case, if we assume that users subscribe to a single channel only, nodes subscribed to channel 1 store channel 1 iff $U_{11} = (1+z_1)f_c(n_1/M) \geq U_{21} = z_2 f_c(n_2/M)$, while nodes subscribed to channel 2 store channel 2 iff $U_{22} = (1+z_2)f_c(n_2/M) \geq U_{12} = z_1 f_c(n_1/M)$. Based on this observation, we can identify 3 regions, depending on the value taken by n_1, as in Figure 2. The boundaries of these regions (besides the trivial values 0 and M), are defined by the points where U_{11} is equal to U_{21} (point H_1^+ in Figure 2), and where U_{22} is equal to U_{12} (point H_1^- in Figure 2). For values $n_1 < H_1^-$ channel 1 is so poorly distributed, that *all* nodes consider it as the most useful one, and thus store it. For values $n_1 > H_1^+$ channel 1 is so widely distributed that *no* node considers it as the most useful, and thus no one stores it. Between H_1^- and H_1^+ nodes subscribed to channel 1 store channel 1, and nodes subscribed to channel 2 store channel 2. Note that, the same line of reasoning holds also with respect to n_2. Specifically, it can be shown that $H_2^+ = M - H_1^-$ and $H_2^- = M - H_1^+$.

[3]Note that we are assuming that the probability distribution of subscribing to channels is the same for all nodes.

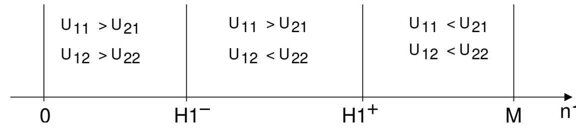

Figure 2: Utility regions.

The type and parameters of the cost function $f_c()$ play a fundamental role with respect to these three regions: they determine the values of H_1^- and H_1^+, and thus, the form of the regions. We are in the position of showing that, in turns, *the shape of the regions fully determines the transient and stationary regimes of the Markov chain*. This is shown in the following section (3.4) with respect to strictly monotonically decreasing cost functions $f_c()$. Note that the following derivations assume that $f_c()$ is such that $H_1^+ \geq H_1^-$ holds true. Similar results can be obtained also in the complementary case (not shown here for the sake of space).

3.4 Analysis of the data dissemination process with generic cost functions

The core of our analysis is the result in Theorem 1, which provides the possible stationary regimes of the distribution process, as well as the conditions under which each regime is reached. Note that we express the conditions with respect to H_1^- and H_1^+. It is trivial to express the same conditions in terms of H_2^- and H_2^+.

THEOREM 1. *The utility-based data dissemination process converges to one of the following configurations:*

- *nodes store the data objects of the channel they are subscribed to iff $H_1^+ \geq M$ or $H_1^- \leq 0$;*
- *nodes either store data objects of the channel they are subscribed to, or oscillate between synchronously storing data objects of channel 1 and data objects of channel 2, iff $0 < H_1^- \leq Mz_1$ and $Mz_1 \leq H_1^+ < M$;*
- *nodes oscillate between synchronously storing data objects of channel 1 and channel 2 iff $H_1^- > Mz_1$ or $H_1^+ < Mz_1$.*

As anticipated in Section 3.1, Theorem 1 states that the data dissemination process always reaches one of two possible stationary regimes. The first one corresponds to a greedy behaviour, in which nodes store only data objects of the channel they are subscribed to. The second one is an oscillating regime, in which either all nodes store data objects of channel 1, or store data objects of channel 2. Note that, as data objects are always available at nodes that generate them in the first place (or at Access Points), they never completely disappear from the network. Theorem 1 also provides the parameters' ranges for which just one of these regimes can be reached, or both are feasible. As we will show in Section 3.4.1 (Lemma 4), this immediately permits to identify the set of initial conditions leading to each regime. Therefore, Theorem 1 describes both the stationary and the transient regimes of the data distribution process, thus providing its full characterisation.

The proof of Theorem 1 follows immediately from the following Lemmas. See Appendix A for the proofs of the Lemmas.

LEMMA 1. *In the Markov chain represented by* $\mathbf{n} = (n_1, n_2)$ *the state* (Mz_1, Mz_2) *is absorbing, and all the other states are transient iff* $H_1^+ \geq M$ *or* $H_1^- \leq 0$.

LEMMA 2. *In the Markov chain represented by* $\mathbf{n} = (n_1, n_2)$ *the state* (Mz_1, Mz_2) *is absorbing, the class* $\{(0, M), (M, 0)\}$ *is recurrent and periodic with period 2, and all the other states are transient iff* $0 < H_1^- \leq Mz_1$ *and* $Mz_1 \leq H_1^+ < M$.

LEMMA 3. *In the Markov chain represented by* $\mathbf{n} = (n_1, n_2)$ *the class* $\{(0, M), (M, 0)\}$ *is recurrent and periodic with period 2, and all the other states are transient iff* $H_1^- > Mz_1$ *or* $H_1^+ < Mz_1$.

3.4.1 Probability of reaching the absorbing state and the recurrent class

Theorem 1 identifies one recurrent periodic class and one absorbing state. Trivially, when only the absorbing state or the unique periodic recurrent class is feasible, that is reached with probability 1. However, under the conditions of Lemma 2, both the absorbing state (Mz_1, Mz_2) and the recurrent class $\{(0, M), (M, 0)\}$ are feasible. In this case it is useful to compute the probability of reaching either, as a function of the initial allocation of data objects in the network. This is the final step to characterise also the transient regime of the process.

To this end, the following lemma describes the evolution of the Markov chain from any possible initial state (see Appendix A for the proof).

LEMMA 4. *Under the conditions of Lemma 2, the absorbing state* (Mz_1, Mz_2) *is reached in at most one step from any initial state* (n_1^0, n_2^0) *such that* $H_1^- \leq n_1^0 \leq H_1^+$. *The periodic recurrent class* $\{(0, M), (M, 0)\}$ *is reached in at most one step from all the other initial states.*

Based on Lemma 4, the probability of reaching the absorbing state (Mz_1, Mz_2) is the probability that the number of nodes choosing channel 1 as the most useful at the system's start up lies within $[H_1^-, H_1^+]$. We assume that, at the system's startup, each node independently selects channel 1 with probability p_1, and channel 2 with probability $p_2 = 1 - p_1$ (according to the previous hypotheses, we also assume that each node immediately stores data objects of the channel selected as the most useful). Therefore, n_1^0 is a random variable with binomial distribution. Thus, the probability of reaching the absorbing state (Mz_1, Mz_2) is equal to:

$$P_{abs} = \sum_{k=\lceil H_1^- \rceil}^{\lfloor H_1^+ \rfloor} \binom{M}{k} p_1^k (1-p_1)^{M-k}, \quad (5)$$

and the probability of reaching the periodic recurrent class $\{(0, M), (M, 0)\}$ is $1 - P_{abs}$.

Clearly, P_{abs} is a key performance parameter, as it indicates the probability that the system oscillates or does not oscillate in the stationary regime.

3.5 Results for specific cost functions

When $f_c()$ is precisely instantiated, the conditions of Theorem 1 translate into conditions on the cost function parameters. Hereafter we explicitly derive these conditions for exponential and linear cost functions, that are the types of functions that we consider in the performance analysis. These are two of the simplest functions that meet the general requirements of $f_c()$. The proofs follow from basic algebraic manipulations from the utility definition (Equation 3), the conditions defining H_1^- and H_1^+, and the conditions of Theorem 1 (see Appendix A for the details).

COROLLARY 1. *When* $f_c(n_i) = e^{-\lambda \frac{n_i}{M}}$ *then:*

- $H_1^- = \frac{M}{2}\left(1 - \frac{1}{\lambda}\ln\frac{1+z_2}{z_1}\right)$;
- $H_1^+ = \frac{M}{2}\left(1 + \frac{1}{\lambda}\ln\frac{1+z_1}{z_2}\right)$.

Furthermore, the data dissemination process converges to one of the following stationary regimes:

- *nodes store data objects of the channel they are subscribed to iff* $\lambda \leq \max\left\{\ln\frac{1+z_1}{z_2}, \ln\frac{1+z_2}{z_1}\right\}$;
- *nodes either store data objects of the channel they are subscribed to, or oscillate between synchronously storing data objects of channel 1 and data objects of channel 2, iff* $\max\left\{\ln\frac{1+z_1}{z_2}, \ln\frac{1+z_2}{z_1}\right\} < \lambda \leq \max\left\{\frac{\ln\frac{1+z_1}{z_2}}{2z_1-1}, \frac{\ln\frac{1+z_2}{z_1}}{2z_2-1}\right\}$
- *nodes oscillate between synchronously storing data objects of channel 1 and data objects of channel 2 iff* $\lambda > \max\left\{\frac{\ln\frac{1+z_1}{z_2}}{2z_1-1}, \frac{\ln\frac{1+z_2}{z_1}}{2z_2-1}\right\}$.

COROLLARY 2. *When* $f_c(n_i) = 1 - \lambda\frac{n_i}{M}$ *then:*

- $H_1^- = \frac{M}{2}\left(1 - \frac{2-\lambda}{\lambda}z_2\right)$;
- $H_1^+ = \frac{M}{2}\left(1 + \frac{2-\lambda}{\lambda}z_1\right)$.

Furthermore, the data dissemination process converges to one of the following stationary regimes:

- *nodes store data objects of the channel they are subscribed to iff* $\lambda \leq \frac{2\max\{z_1,z_2\}}{1+\max\{z_1,z_2\}}$;
- *nodes either store data objects of the channel they are subscribed to, or oscillate between synchronously storing data objects of channel 1 and data objects of channel 2, iff* $\frac{2\max\{z_1,z_2\}}{1+\max\{z_1,z_2\}} < \lambda \leq \frac{2\max\{z_1,z_2\}}{3\max\{z_1,z_2\}-1}$
- *nodes oscillate between synchronously storing data objects of channel 1 and data objects of channel 2 iff* $\lambda > \frac{2\max\{z_1,z_2\}}{3\max\{z_1,z_2\}-1}$.

4. PERFORMANCE EVALUATION

The main focus of our evaluation is studying the impact of the key system and environment parameters on the resulting stationary regime. This is indeed the main focus for tuning data dissemination system at design time. Therefore, we investigate the behaviour of P_{abs}, defined in Equation 5, and the transition points between the different regions in which different stationary regimes can be achieved, derived in Corollaries 1 and 2.

First of all, we validate our analytical model through simulations, and investigate the impact of different initial data objects allocations (Section 4.1). As the model shows to be accurate, we perform a sensitiveness analysis of P_{abs} with respect to two key system's parameters, i.e., the number of users and the popularity of the channels (Sections 4.2 and 4.3, respectively).

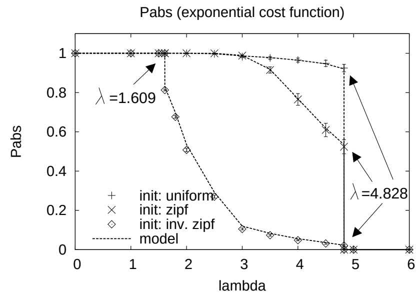

Figure 3: Validation with exponential cost function.

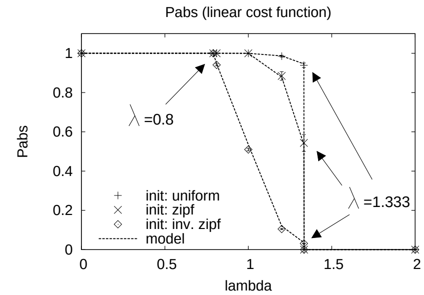

Figure 4: Validation with linear cost function.

4.1 Model validation

In this section we validate the analytical expressions of P_{abs} (Equation 5) and of the transition points in Corollary 1 and 2, i.e., the points in which the type of stationary regime changes. We consider different distributions for the initial allocation of data objects at the system startup. We also consider both exponential and linear cost functions.

We use a custom simulator written in C++. The simulator shares the model's assumption about the fact that nodes can exactly estimate the parameters of the utility function. The simulator also assumes that nodes can find enough data objects to fill their caches upon selecting the most useful channel. Note that these assumptions make the simulation model independent of the users' mobility process, provided it meets the requirements set in Section 3.2. The simulator parameters are thus the number of nodes M, the popularity of channels (z_1, z_2), the type of cost function $f_c()$ (and its parameter λ), the initial data allocation distribution (p_1 and p_2 being the probability of each node to store data objects of channel 1 or 2 at the system's startup, respectively). We have explored three different distributions, i.e., uniform ($p_1 = p_2 = 0.5$), zipf with parameter 1 ($p_1 = 2/3, p_2 = 1/3$), "inverse" zipf with parameter 1 ($p_1 = 1/3, p_2 = 2/3$). Note that we have always set the channel popularity distribution (z_1, z_2) according to a zipf law with parameter 1. The experiments with initial zipf data allocation model a system in which the initial allocation of data objects follows the same distribution of users' interests. The experiments with the inverse zipf initial data allocation model a system in which the initial allocation of data objects follows an "opposite" distribution with respect to users' interests (i.e., the data objects that are most interesting for users are the least spread and vice versa). For all the initial data allocation distributions, we have varied the cost function parameter λ so as to explore all the cases predicted by Theorem 1. Finally, the number of nodes M has been set to 99 to have an integer number of nodes subscribed to each channel. For each initial data allocation, cost function and value of λ we have performed at least 5000 i.i.d. simulation runs. At the beginning of each run, we allocate channels according to the initial distribution, and let the system evolve until it reaches the stationary regime. Each run provides a sample Z equal to 1 if the system does not oscillates, and equal to 0 if it oscillates in the stationary regime. We use standard analysis techniques of the simulation output (see, e.g., [6]) to estimate the average value of P_{abs} and compute confidence intervals with a 99% confidence level.

Figures 3 and 4 show P_{abs} as a function of λ for exponential and linear cost functions, respectively. The comparison between the analytical and simulation curves assesses the model's accuracy. Note that, according to the model, the transition between the region where the system always converges to the absorbing state (Mz_1, Mz_2) (in which $P_{abs} = 1$) and the region where the system either converges to the absorbing state or oscillates occurs for $\lambda = 1.609$ for the exponential cost function, and $\lambda = 0.8$ for the linear cost function. Then, the transition to the region where the system always oscillates (in which $P_{abs} = 0$) occurs at $\lambda = 4.828$ and $\lambda = 1.333$, respectively. Figures 3 and 4 show that also these transition points are correctly predicted by the model. It is also interesting to note that these transition points may mark a sort of discontinuous phase changes in the system. For example, in the exponential case, P_{abs} abruptly drops from 1 to about 0.8 at the first transition point, when the initial data allocation is inverse zipf. Then, P_{abs} drops from about 0.9 and 0.5 to 0 at the second transition point, for the uniform and zipf initialisation, respectively. Also note that the initial data allocation has a large impact on P_{abs}. This can be intuitively understood by looking again at Figure 2. For example, let us compare the curves related to an initial zipf and inverse zipf data allocation, in the regions under the hypothesis of Lemma 2 ($1.609 < \lambda \leq 4.828$ for the exponential case). In the former case, the distribution of the number of nodes initially storing channel 1 tends to concentrate around Mz_1. In the latter case, around Mz_2. It is easy to show that, as $z_1 > z_2$, $H_1^+ - Mz_1$ is greater than $Mz_2 - H_1^-$. Therefore, the probability of having initial conditions (n_1^0, n_2^0) leading to an oscillating regime when the inverse zipf initial data allocation is used (i.e., having $n_1^0 < H_1^-$) is greater than when a zipf initial data allocation (i.e., having $n_1^0 > H_1^+$) is used. Similar remarks can be derived for the linear cost function, as well.

4.2 Sensitiveness to the user population

In this section we analyse, by exploiting the analytical model, the sensitiveness of P_{abs} to the number of users in the system, M. Figures 5 and 6 consider the cases of exponential and linear cost functions. We mainly wish to highlight the different behaviour with different initialisations, thus we plot together curves obtained with initial zipf and inverse zipf data allocations. The behaviour with uniform initialisation is similar to that with zipf initialisation. Since the users' interests distribution (z_1, z_2) is always zipf with parameter 1 (thus, $z_1 = 2/3, z_2 = 1/3$), we consider values of M which are integer multiples of 3, so as to have an integer number

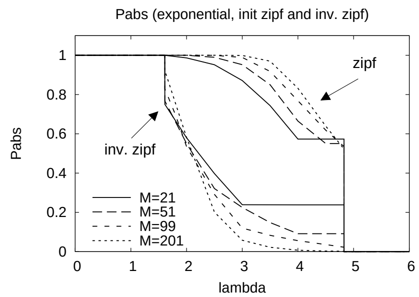

Figure 5: $P_{abs}(M)$ with exponential cost function.

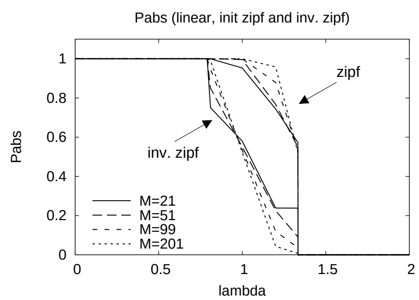

Figure 6: $P_{abs}(M)$ with linear cost function.

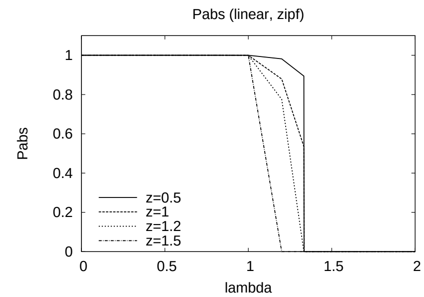

Figure 7: $P_{abs}(z)$ with zipf initialisation.

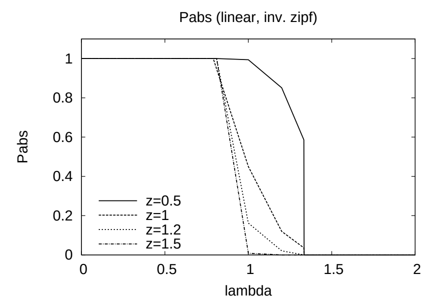

Figure 8: $P_{abs}(z)$ with inverse zipf initialisation.

of nodes in the absorbing state (Mz_1, Mz_2). Furthermore, we increase M according to a logarithmic scale.

First of all, note that the transition points between the regions with different stationary regimes do not change. This is because the transition points are functions of λ, z_1, and z_2 only (see Section 3.5). It is interesting to note that for zipf initialisation P_{abs} constantly increases with M, while with the inverse zipf initialisation it increases for low values of λ and decreases afterwards. In the zipf initialisation, the initial distribution of n_1^0 concentrates around Mz_1. It is easy to show that the differences $H_1^+ - Mz_1$ and $Mz_1 - H_1^-$ (i.e., the distance between Mz_1 and the transition points) can be expressed as $\alpha(\lambda, z_1, z_2)M, \alpha > 0\ \forall\lambda$. Even though the variance of the n_1^0 distribution increases with M, this is more than compensated by the linear increase of those differences with M. This means that, as M increases, the probability of starting in a state of the chain that reaches the absorbing state (i.e., the probability that $n_1^0 < H_1^+$, see Section 3.4.1) increases. On the other hand, in the case of the inverse zipf initialisation, n_1^0 is distributed around Mz_2. It can be shown that $Mz_2 - H_1^- = \alpha(\lambda, z_1, z_2)M$ too. However, in this case, α is positive just below a certain value of λ. Thus, beyond this threshold, increasing M results in *lower* probabilities of reaching the absorbing state (i.e., of having $n_1^0 > H_1^-$). The same holds true both for exponential and linear cost functions.

4.3 Sensitiveness to the data popularity

Finally, we discuss the behaviour of P_{abs} as a function of the parameter (denoted as z) of the zipf distribution used to model users' interests (and thus defining z_1 and z_2). The higher z, the more channel 1 is popular with respect to channel 2. Figures 7 and 8 show $P_{abs}(z)$ for linear cost functions in the cases of zipf and inverse zipf initialisations, respectively. We choose these particular configurations to show a general behaviour we have found (see Appendix B for more details). Two remarks can be done. First of all, the transition points between the different stationary regimes move with z. Specifically, as can be shown by the Equations derived in Section 3.5, the value of λ for which the first transition occurs increases with z, while the value for which the second transition occurs decreases. Furthermore, plots highlight the same kind of behaviour of P_{abs} highlighted in the previous section, as far as the initial distribution of channels is concerned. In the case of a zipf initialisation, P_{abs} always decreases when z increases, while for an inverse zipf initialisation there is a threshold value for λ: before the threshold, increasing z results in higher values of P_{abs}, beyond the threshold, P_{abs} decreases when z increases.

5. CONCLUSIONS AND FUTURE WORK

In this paper we have proposed an analytical model to study utility-based cooperative (social) data dissemination systems for opportunistic networks. Although bound to some simplifying assumptions, the modelling approach shows to be very promising, as the resulting analysis permits to achieve complete insights (and, thus, full control) on the transient and stationary regimes of the data distribution process. System designers can exploit such a model to tune the system parameters (such as the utility function) or, if possible, environment parameters (such as the initial allocation of data objects on nodes), so as to achieve a target distribution of data objects in the system. As the main subject of future work we consider the extension of the detailed results we have provided in this paper in more general cases, by releasing some of the assumptions discussed in Section 3.2.

6. REFERENCES

[1] A. Balamash and M. Krunz. An overview of web caching replacement algorithms. *IEEE Communications Surveys*, 6(2):44–56, 2004.

[2] A. Balasubramanian, B. N. Levine, and A. Venkataramani. Dtn routing as a resource allocation problem. *Proc. of ACM SIGCOMM*, 2007.

[3] C. Boldrini, M. Conti, and A. Passarella. Context and resource awareness in opportunistic network data dissemination. In *Proc. of IEEE AOC*, 2008.

[4] R. Friedman, D. Gavidia, L. Rodrigues, A. C. Viana, and S. Voulgaris. Gossiping on MANETs: the beauty and the beast. *SIGOPS Oper. Syst. Rev.*, 41(5):67–74, 2007.

[5] H. Kellerer, U. Pferschy, and D. Pisinger. *Knapsack Problems*. Springer, 2004.

[6] M. Law and D. Kelton. *Simulation Modeling and Analysis*. McGraw-Hill, second edition, 1991.

[7] V. Lenders, G. Karlsson, and M. May. Wireless Ad hoc Podcasting. In *Proc. of IEEE SECON*, 2007.

[8] L. Pelusi, A. Passarella, and M. Conti. Opportunistic Networking: data forwarding in disconnected mobile ad hoc networks. *IEEE Communications Magazine*, 44(11), Nov. 2006.

[9] H. Shen, M. Kumar, S. K. Das, and Z. Wang. Energy-efficient data caching and prefetching for mobile devices based on utility. *Mob. Netw. Appl.*, 10(4):475–486, 2005.

[10] L. Yin and G. Cao. Supporting cooperative caching in ad hoc networks. *IEEE Transactions on Mobile Computing*, 5(1):77–89, 2006.

[11] E. Yoneki, P. Hui, S. Chan, and J. Crowcroft. A socio-aware overlay for publish/subscribe communication in delay tolerant networks. In *Proc. of ACM MSWiM '07*, pages 225–234, 2007.

[12] W. Zhao, M. Ammar, and E. Zegura. Multicasting in delay tolerant networks: semantic models and routing algorithms. In *Proc. of the ACM WDTN*, 2005.

APPENDIX

A. PROOFS OF THE LEMMAS, THEOREM, AND COROLLARIES

This Appendix contains the demonstrations of Lemmas 1÷3 of Section 3.4, of Lemma 4 of Section 3.4.1, and of Corollaries 1 and 2 in Section 3.5. Note that we make explicit the assumption underlying Figure 2, i.e., $H_1^+ \geq H_1^-$ and $H_2^+ \geq H_2^-$.

LEMMA 1. *In the Markov chain represented by* $\mathbf{n} = (n_1, n_2)$ *the state* (Mz_1, Mz_2) *is absorbing, and all the other states are transient iff* $H_1^+ \geq M$ *or* $H_1^- \leq 0$.

PROOF. We firstly prove that the condition $H_1^+ \geq M \vee H_1^- \leq 0$ is a sufficient condition for (Mz_1, Mz_2) being the unique absorbing state of the chain. Let us consider the case $H_1^+ \geq M$. In this case, starting from any state, after one step nodes subscribed to channel 1 always store channel 1, because U_{11} is always greater than U_{21}. That is, starting from any state, $n_1 \geq Mz_1$ holds true after one step. Therefore, also $n_2 \leq Mz_2$ holds true after one step. We can prove that if $H_1^+ \geq M$, then $Mz_2 \leq H_2^+$. This means that, from any initial condition, after one step nodes subscribed to channel 2 always see channel 2 as the most useful, and thus store channel 2. This proves that (Mz_1, Mz_2) is an absorbing state, and all other states are transient.

To prove that $Mz_2 \leq H_2^+$ we firstly observe that, in general, both H_1^+ and H_2^+ are greater than $M/2$. By definition, in H_2^+ the following equation holds $(1+z_2)f_c(H_2^+) = z_1 f_c(M-H_2^+)$. Therefore, we can write $1 \leq \frac{1+z_2}{z_1} = \frac{f_c(M-H_2^+)}{f_c(H_2^+)}$. It can be shown that when $H_2^+ > H_2^-$ then both $f_c(H_2^+)$ and $f_c(M - H_2^+)$ are positive. Since $f_c()$ is monotonically decreasing, we obtain $M-H_2^+ \leq H_2^+ \Rightarrow H_2^+ \geq M/2$. If $z_2 \leq z_1$ then we obtain $Mz_2 \leq M/2 \leq H_2^+$. If $z_2 > z1$ we prove that $H_2^+ > H_1^+ \geq M$. Thus, in this case $Mz_2 < H_2^+$. In H_1^+ the following equation holds $(1+z_1)f_c(H_1^+) = z_2 f_c(M - H_1^+)$, or $\frac{1+z_1}{z_2} = \frac{f_c(M-H_1^+)}{f_c(H_1^+)}$. It can be shown that when $H_1^+ > H_1^-$ then both $f_c(H_1^+)$ and $f_c(M - H_1^+)$ are positive.By recalling the similar equation holding in H_2^+, and the fact that $z2 > z1$, we obtain $\frac{f_c(M-H_2^+)}{f_c(H_2^+)} > \frac{f_c(M-H_1^+)}{f_c(H_1^+)}$. As $f_c()$ is monotonically decreasing, $H_2^+ > H_1^+$ follows immediately.

The case when $H_1^- < 0$ can be proved by exactly the same methodology by just swapping channels 1 and 2, just after showing that $H_1^- \leq 0 \Rightarrow H_2^+ \geq M$. To show this, we firstly proove that $H_2^+ = M - H_1^-$ (and, by the way, that $H_2^- = M - H_1^+$). By definition (see Figure 2), in H_2^+, $U_{22} = U_{12}$, thus the following equation holds: $(1+z_2)f_c(H_2^+) = z_1 f_c(M - H_2^+)$. At the same time, also in H_1^- the equation $U_{22} = U_{12}$ holds true, thus $(1+z_2)f_c(M-H_1^-) = z_1 f_c(H_1^-)$. Thus, we obtain $\frac{f_c(M-H_2^+)}{f_c(H_2^+)} = \frac{f_c(H_1^-)}{f_c(M-H_1^-)}$. Since $f_c()$ is monotonically decreasing, this is possible iff $H_2^+ = M - H_1^-$. Following the same line of reasoning, it is straightforward to proove also that $H_2^- = M - H_1^+$.

To conclude the proof we show that $H_1^+ \geq M \vee H_1^- \leq 0$ is a necessary condition for (Mz_1, Mz_2) being an absorbing state, and all the other states being transient. This is easy to prove by showing that if $H_1^+ < M \wedge H_1^- > 0$ then the class $\{(0, M), (M, 0)\}$ is a recurrent class. Starting from state $(0, M)$, all nodes consider channel 1 as the most useful (see Figure 2). Thus, the chain reaches state $(M, 0)$ at the next step. Then, in this state all nodes consider channel 2 as the most useful, and thus the chain reaches back state $(0, M)$. $\square$

We now provide the proof of Lemma 3, as the proof of Lemma 2 can be based on the results of Lemmas 1 and 3.

LEMMA 3. *In the Markov chain represented by* $\mathbf{n} = (n_1, n_2)$ *the class* $\{(0, M), (M, 0)\}$ *is recurrent and periodic with period 2, and all the other states are transient iff* $H_1^- > Mz_1$ *or* $H_1^+ < Mz_1$.

PROOF. We firstly proof that if $H_1^- > Mz_1 \vee H_1^+ < Mz_1$ then the class $\{(0, M), (M, 0)\}$ is recurrent and periodic with period 2, and all the other states are transient.

Let us firstly focus on the case $H_1^+ < Mz_1$. We show that from any initial state the process ends up oscillating with period 2 between $(0, M)$ and $(M, 0)$.

Let us consider an initial condition such that $H_1^- \le n_1^0 \le H_1^+$. Since, as shown in the proof of Lemma 1, $H_2^- = M - H_1^+$ and $H_2^+ = M - H_1^-$, then we can write $M - H_1^- \ge M - n_1^0 \ge M - H_1^+$, thus $H_2^- \le n_2^0 \le H_2^+$. Therefore, in the initial condition (n_1^0, n_2^0) nodes subscribed to channel 1 (2) consider channel 1 (2) as the most useful, and the process reaches state (Mz_1, Mz_2) after one step. In this state *all* nodes see channel 2 as the most useful, as $Mz_1 > H_1^+$ and thus $Mz_2 = M - Mz_1 < M - H_1^+ = H_2^-$. Therefore, the process reaches state $(0, M)$. We show that in this state *all* nodes consider channel 1 as the most useful. To this end, we preliminary show that if $H_1^+ < M$ and $z_1 > z_2$ then $H_1^- > 0$. From the proof of Lemma 1 it is straightforward to show that if $z_1 > z_2$ then $H_1^+ > H_2^+$. Thus, if $z_1 > z_2$ and $H_1^+ < M$ we obtain $H_1^- = M - H_2^+ > M - H_1^+ > 0$. In our hypothesis $H_1^+ < Mz_1 \le M$, thus these properties hold true. Therefore, in the state $(0, M)$ all nodes consider channel 1 as the most useful as $H_1^- > 0$ and $H_2^+ < M$. The chain thus reaches state $(M, 0)$. Again, since $H_1^+ < M$ and thus $H_2^- > 0$ and $H_1^+ < M$, the state reaches $(0, M)$. Thus, the class $(0, M), (M, 0)$ is recurrent and periodic with period 2. Let us now consider the case $n_1^0 > H_1^+$. As this implies that $n_2^0 < H_2^-$, all nodes see channel 2 as the most useful, and the process reaches the state $(0, M)$ in one step. As shown above, from this state the process reaches $(M, 0)$ and then $(0, M)$ again. Finally, let us consider the case $n_1^0 < H_1^-$. This implies $n_2^0 > H_2^+$ and thus all nodes see channel 1 as the most useful. Thus, the process reaches the state $(M, 0)$ and thus the periodic recurrent class $(0, M), (M, 0)$.

The case when $H_1^- > Mz_1$ can be analysed by swapping channel 1 and 2 in the above line of reasoning, because $H_1^- > Mz_1 \Rightarrow H_2^+ < Mz_2$.

To conclude the proof, we show that if the class $\{(0, M), (M, 0)\}$ is recurrent and periodic with period 2, and all the other states are transient then $H_1^- > Mz_1$ or $H_1^+ < Mz_1$. To this end, it is sufficient to observe that if $H_1^- \le Mz_1 \le H_1^+$ then the state (Mz_1, Mz_2) is absorbing. $\square$

We are now in the position of proving Lemma 2 as well.

LEMMA 2. *In the Markov chain represented by* $\mathbf{n} = (n_1, n_2)$ *the state* (Mz_1, Mz_2) *is absorbing, the class* $\{(0, M), (M, 0)\}$ *is recurrent and periodic with period 2, and all the other states are transient iff* $0 < H_1^- \le Mz_1$ *and* $Mz_1 \le H_1^+ < M$.

PROOF. We first proove that if $0 < H_1^- \le Mz_1$ and $Mz_1 \le H_1^+ < M$ then the state (Mz_1, Mz_2) is absorbing, the class $\{(0, M), (M, 0)\}$ is recurrent and periodic with period 2, and all the other states are transient. This can be shown by considering the possible evolutions of the process starting from any initial state (n_1^0, n_2^0).

Let us consider the case in which $n_1^0 < H_1^-$, which implies $n_2^0 > H_2^+$. All nodes consider channel 1 as the most useful, and thus the process reaches the state $(M, 0)$. Since $H_1^+ < M$ and $H_1^- > 0$ then all nodes consider channel 2 as the most useful, and the process reaches the state $(0, M)$. From the thypothesis it is straightforward that $0 < H_2^- \le Mz_2 \le H_2^+ < M$ holds true. Therefore, from state $(0, M)$ the process gets back to $(M, 0)$. This shows that $(0, M), (M, 0)$ is a recurrent class with period 2, and all states (n_1, n_2) such that $n_1 < H_1^-$ are transient. The same conclusion can be drawn in the case $n_1 > H_1^+$, by swapping channels 1 and 2 in the above line of reasoning, and considering that $n_1 > H_1^+ \Rightarrow n_2 < H_2^-$. Finally, let us consider the case in which $H_1^- \le n_1^0 \le H_1^+$. Note that this implies $H_2^- \le n_2^0 \le H_2^+$ as well. This means that in state (n_1^0, n_2^0) nodes consider the channel they are subscribed to as the most useful. Therefore, the process reaches state (Mz_1, Mz_2) in one step. As $H_1^- \le Mz_1 \le H_1^+$ and $H_2^- \le Mz_2 \le H_2^+$, the state (Mz_1, Mz_2) is absorbing, and thus all states (n_1, n_2) such that $H_1^- \le n_1 \le H_1^+$ are transient.

To conclude the proof, we have to show that if the state (Mz_1, Mz_2) is absorbing, the class $\{(0, M), (M, 0)\}$ is recurrent and periodic with period 2, and all the other states are transient, then $0 < H_1^- \le Mz_1$ and $Mz_1 \le H_1^+ < M$. This can be easily shown by recalling Lemmas 1 and 3. Specifically, if $H_1^- \le 0$ or $H_1^+ \ge M$ then Lemma 1 shows that the states $(0, M)$ and $(M, 0)$ are both transient. Furthermore, if $0 \le Mz_1 < H_1^- \le H_1^+ < M$, or $0 < H_1^- \le H_1^+ < Mz_1 < M$ hold true, then Lemma 3 shows that the state (Mz_1, Mz_2) is transient. $\square$

We are also in the position to proove Lemma 4.

LEMMA 4. *Under the conditions of Lemma 2, the absorbing state* (Mz_1, Mz_2) *is reached in at most in one step from any initial state* (n_1^0, n_2^0) *such that* $H_1^- \le n_1^0 \le H_1^+$. *The periodic recurrent class* $\{(0, M), (M, 0)\}$ *is reached in at most one step from all the other initial states.*

PROOF. This has been already shown by prooving Lemma 2. $\square$

Finally, we derive the conditions on the parameter (λ) of both exponential and linear cost functions, under the hypotheses of Lemmas 1, 2, and 3, respectively.

COROLLARY 1. *When* $f_c(n_i) = e^{-\lambda \frac{n_i}{M}}$ *then:*

- $H_1^- = \frac{M}{2}\left(1 - \frac{1}{\lambda} \ln \frac{1+z_2}{z_1}\right)$;
- $H_1^+ = \frac{M}{2}\left(1 + \frac{1}{\lambda} \ln \frac{1+z_1}{z_2}\right)$.

Furthermore, the data dissemination process converges to one of the following stationary regimes:

- *nodes store the channel they are subscribed to iff* $\lambda \le \max\left\{\ln \frac{1+z_1}{z_2}, \ln \frac{1+z_2}{z_1}\right\}$;
- *nodes either store the channel they are subscribed to, or oscillate between storing all channel 1 and channel 2, iff* $\max\left\{\ln \frac{1+z_1}{z_2}, \ln \frac{1+z_2}{z_1}\right\} < \lambda \le \max\left\{\frac{\ln \frac{1+z_1}{z_2}}{2z_1-1}, \frac{\ln \frac{1+z_2}{z_1}}{2z_2-1}\right\}$

- *nodes oscillate between storing all channel 1 and channel 2 iff* $\lambda > \max\left\{\frac{\ln\frac{1+z_1}{z_2}}{2z_1-1}, \frac{\ln\frac{1+z_2}{z_1}}{2z_2-1}\right\}$.

PROOF. We just provide the line of reasoning of the proof, as most of it then consists in straightforward algebraic manipulations.

The values of H_1^+ can be found by recalling that in H_1^+ $U_{11} = U_{21}$, which results in the equation $(1+z_1)e^{-\lambda\frac{H_1^+}{M}} = z_2 e^{-\lambda\frac{M-H_1^+}{M}}$. Solving this equation for H_1^+ yields the expression in the corollary. The expression of H_1^- can be similarly found by recalling that in H_1^- $U_{22} = U_{12}$.

The three conditions on λ to reach the alternative stationary regimes are derived by substituting the expressions of H_1^+ and H_1^- in the constraints of Lemmas 1, 2 and 3. Specifically, the constraints of Lemma 1 are $H_1^- \le 0 \vee H_1^+ \ge M$. The first constraint yields the condition $\lambda \le \ln\frac{1+z_2}{z_1}$. The second constraint yields the condition $\lambda \le \ln\frac{1+z_1}{z_2}$. Since the overall condition of the Lemma results by the logical or of the two conditions, the Lemma is satisfied iff $\lambda \le \max\left\{\ln\frac{1+z_1}{z_2}, \ln\frac{1+z_2}{z_1}\right\}$.

The overall constraint of Lemma 2 can be written as $0 < H_1^- \le Mz_1 \le H_1^+ < M$, which can be broken down in two sub-constraints that have to be met. The first sub-constraint is $0 < H_1^+ \le H_1^+ < M$, which clearly yields the sub-condition $\lambda > \max\left\{\ln\frac{1+z_1}{z_2}, \ln\frac{1+z_2}{z_1}\right\}$. The second sub-constraint is $H_1^- \le Mz_1 \le H_1^+$. Recall from the proof of Lemma 1 that $H_1^- \le M/2$ and $H_1^+ \ge M/2$ always hold true. Therefore, if $z_1 \ge 1/2$ the part of the sub-constraint $H_1^- \le Mz_1$ is always verfied. In this case, by substituting the expression of H_1^+, the sub-constraint yields the sub-condition $\lambda \le \frac{\ln\frac{1+z_1}{z_2}}{2z_1-1}$. On the other hand, if $z_1 < 1/2$ the part of the sub-constraint $Mz_1 \le H_1^+$ is always verified, and the sub-constraint yields the sub-condition $\lambda \le \frac{\ln\frac{1+z_2}{z_1}}{2z_2-1}$. Considering the unique condition $\lambda \le \max\left\{\frac{\ln\frac{1+z_1}{z_2}}{2z_1-1}, \frac{\ln\frac{1+z_2}{z_1}}{2z_2-1}\right\}$ the dependence on the value of z_1 can be dropped. Finally, the condition shown in Corollary 1 corresponding to the constraints of Lemma 2 can be derived by taking the logical and of the sub-conditions on λ.

Finally, the overall constraint of Lemma 3 is the logical or of the sub-constraints $H_1^- > Mz_1$ and $H_1^+ < Mz_1$. The first one is never verified when $z_1 > 1/2$, and yilds the condition $\lambda > \frac{\ln\frac{1+z_2}{z_1}}{2z_2-1}$ when $z_1 < 1/2$. The second sub-constraint is never verified for $z_1 < 1/2$ and yields the sub-condition $\lambda > \frac{\ln\frac{1+z_1}{z_2}}{2z_1-1}$ for $z_1 > 1/2$. Taking the logical or of the sub-condition yields the condition $\lambda > \max\left\{\frac{\ln\frac{1+z_1}{z_2}}{2z_1-1}, \frac{\ln\frac{1+z_2}{z_1}}{2z_2-1}\right\}$. □

COROLLARY 2. *When* $f_c(n_i) = 1 - \lambda\frac{n_i}{M}$ *then:*

- $H_1^- = \frac{M}{2}\left(1 - \frac{2-\lambda}{\lambda}z_2\right)$;
- $H_1^+ = \frac{M}{2}\left(1 + \frac{2-\lambda}{\lambda}z_1\right)$.

Furthermore, the data dissemination process converges to one of the following stationary regimes:

- *nodes store the channel they are subscribed to iff* $\lambda \le \frac{2\max\{z_1,z_2\}}{1+\max\{z_1,z_2\}}$;
- *nodes either store the channel they are subscribed to, or oscillate between storing all channel 1 and channel 2, iff* $\frac{2\max\{z_1,z_2\}}{1+\max\{z_1,z_2\}} < \lambda \le \frac{2\max\{z_1,z_2\}}{3\max\{z_1,z_2\}-1}$
- *nodes oscillate between storing all channel 1 and channel 2 iff* $\lambda > \frac{2\max\{z_1,z_2\}}{3\max\{z_1,z_2\}-1}$.

PROOF. As in Corollary 1, the expressions of H_1^+ and H_1^- are straightforward by recalling the properties of the utility functions in these points. Furthermore, the line of reasoning to obtain conditions on λ starting from the constraints of Lemmas 1, 2 and 3 are exactly the same described in the proof of Corollary 2. The precise expressions of these conditions follow from simple algebraic manipulations. □

B. SENSITIVENESS TO THE DATA POPULARITY: GENERAL FINDINGS

In this Appendix we discuss a general behaviour of the system with respect to the data popularity. Specifically, we focus on the case of linear and exponential cost functions, and investigate how the values of λ in the transition points between the different stationary regimes depend on the data popularity. Also in this case we are assuming that $H_1^+ > H_1^-$ and $H_2^+ > H_2^-$ hold true.

This can be easily performed by analysing the functions $\lambda(z_1, z_2)$ in the transition points between the different stationary regimes, shown in Section 3.5. Let us focus on the linear case, and on the transition point between the conditions of Lemma 1 and Lemma 2 (hereafter, first transition point). It is easy to show that, when channel 1 is the most popular ($z_1 > z_2 \Rightarrow z_1 > 1/2$), then the first transition point is defined by $\lambda(z_1) = \frac{2z_1}{1+z_1}$. Increasing the data popularity (i.e., the z parameter of the Zipf distribution) results in increasing z_1. As $\lambda(z_1)$ is always increasing with $z1$, the value of λ in the first transition point always increases with the data popularity. The same remark holds true also when channel 2 is the most popular, as in this case i) $z_2 > z_1$ and $z_2 > 1/2$; ii) increasing z results in increasing z_2, and iii) the transition point is defined by $\lambda(z_2) = \frac{2z_2}{1+z_2}$. The transition point between the conditions of Lemma 2 and Lemma 3 (hereafter, second transition point) can be similarly analysed. When channel 1 is the most popluar the transition is defined by $\lambda(z_1) = \frac{2z_1}{3z_1-1}$, while when channel 2 is the most popular it is defined by $\lambda(z_2) = \frac{2z_2}{3z_2-1}$. Both functions are decreasing with z. This shows that, in the linear case, the value of λ in the first transition point always increases with the data popluarity, and always decreases in the second transition point.

The analysis when the cost function is exponential can be carried out exactly with the same methodology. The first transition point is defined by $\lambda(z_1) = \ln\frac{1+z_1}{1-z_1}$ when channel 1 is the most popular, and by $\lambda(z_2) = \ln\frac{1+z_2}{1-z_2}$ when channel 2 is the most popular. In both cases, the value of λ in the first transition point always increases with the data popularity. However, the value of λ in the second transition point is not monotonic with z. Specifically, the transition point is defined either by $\lambda(z_1) = \frac{\ln\frac{1+z_1}{1-z_1}}{2z_1-1}$ or by $\lambda(z_2) = \frac{\ln\frac{1+z_2}{1-z_2}}{2z_2-1}$. In both cases, λ is initially decreasing and the increasing. The threshold between the two behaviours is provided by the equations $\ln\frac{1+z_1}{1-z_1} = \frac{2z_1-1}{1-z_1^2}$ and $\ln\frac{1+z_2}{1-z_2} = \frac{2z_2-1}{1-z_2^2}$. This shows that in the exponential case the value of λ in the first transition point alwasy increases, but in the second transition point decreases up to a certain point, and increases afterwards.