



HAL
open science

Optimising Message Broadcasting in Opportunistic Networks

Leonardo Chancay-García, Enrique Hernández-Orallo, Pietro Manzoni, Anna Maria Vegni, Valeria Loscrì, Juan Carlos Cano, Carlos Tavares Calafate

► **To cite this version:**

Leonardo Chancay-García, Enrique Hernández-Orallo, Pietro Manzoni, Anna Maria Vegni, Valeria Loscrì, et al.. Optimising Message Broadcasting in Opportunistic Networks. *Computer Communications*, 2020, 157, pp.162-178. 10.1016/j.comcom.2020.04.031 . hal-02544184

HAL Id: hal-02544184

<https://hal.science/hal-02544184>

Submitted on 16 Apr 2020

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.

Optimising Message Broadcasting in Opportunistic Networks

Leonardo Chancay-García^a, Enrique Hernández-Orallo^{a,*}, Pietro Manzoni^a, Anna Maria Vegni^b,
Valeria Loscri^c, Juan Carlos Cano^a, Carlos T. Calafate^a

^a*Departamento de Informática de Sistemas y Computadores. Universitat Politècnica de València. Spain.*

^b*COMLAB Telecommunication lab, Department of Engineering, Roma Tre University, Rome, Italy.*

^c*INRIA Lille-Nord Europe, Lille, France.*

Abstract

Message Broadcasting in Opportunistic Networks is based on the opportunity of establishing contacts among mobile nodes for message exchange. Nevertheless, as the amount of information transmitted in a contact is limited by the transmission speed and the contact duration, large messages are less likely to be exchanged, and thus their diffusion is severely limited. Furthermore, these failed transmissions can also lead to an important waste of network resources, since the message transmission is aborted when the contact ends and the message needs to be transmitted again in the next contact.

Therefore, in this paper we study the impact that contact duration has on the broadcast of messages, showing that splitting a large message into smaller parts can improve its diffusion. Based on this idea, we propose an extension of the epidemic protocol called *Xpread*. The efficiency of this protocol mainly depends on how the original message is partitioned. Thus, in order to evaluate the impact and the efficiency of the partition scheme, we have developed an analytical model based on *Population Processes*, showing that a fixed size partition is the best option, while also providing a simple expression to obtain the optimal size.

The *Xpread* has been evaluated exhaustively using four different mobile traces, comprising both pedestrian and vehicular scenarios. The results show that the diffusion of large messages is improved up to four times with a slight reduction in the delivery time and overhead, minimising also message forwarding failures.

Keywords: Opportunistic networks, Epidemic diffusion, Population Processes

1. Introduction

The goal of message broadcasting or information dissemination among mobile nodes is to spread information and alerts using a communications infrastructure in a cost-effective and timely manner. In this context, we observe the steadily growing appearance of more sophisticated and information-rich applications, such as touristic information, location-based services, road congestion alert providers, etc., which require the diffusion of large messages [1].

*Corresponding author.

Email addresses: leochaga@doctor.upv.es (Leonardo Chancay-García), ehernandez@disca.upv.es (Enrique Hernández-Orallo), pmanzoni@disca.upv.es (Pietro Manzoni), annamaria.vegni@uniroma3.it (Anna Maria Vegni), valeria.loscri@inria.fr (Valeria Loscri), jucano@disca.upv.es (Juan Carlos Cano), calafate@disca.upv.es (Carlos T. Calafate)

Opportunistic Networks (OppNets) have arisen as an effective infrastructure-less communication model for message broadcasting, based on taking advantage of the dynamically created direct and localised communication links (*e.g.*, through a Wi-Fi Direct channel) to exchange messages between nearby nodes. In general, two main communication approaches for OppNets can be considered [2] *i.e.*, (i) *destination-oriented* approaches that assume the destinations are well known ahead, and (ii) *destination-less* approaches, which consider that network participants **should receive a message, for example, as being part of a group, or simply**, that the receivers are not known a priori. In this paper, we consider the latter approach, a scenario with more practical and effective applications, such as disaster scenarios, where the infrastructure has been damaged, or delay-tolerant applications, including local information dissemination and crowd sensing. All of these scenarios can be well served with flooding mechanisms, such as epidemic protocols.

The original source of the messages to be broadcast can be **either** fixed nodes, specifically devoted to this issue and spread along a specific area, or even some mobile nodes, which can obtain this information using other networks (4G or Wi-Fi). For example, considering Vehicular Ad Hoc Networks (VANETs), Vehicle-to-Vehicle (V2V) communication can be established via ad-hoc networking without relying on a fixed infrastructure, whereas in Vehicle-to-Infrastructure (V2I) communication, fixed nodes or Road Side Units (RSUs) are used to spread different kinds of centralised information among the nearby vehicles [3]. Anyway, we adopt a more general approach by integrating all types of nodes (*i.e.*, mobiles and fixed) with the aim of delivering data using the store-carry-and-forward (SCF) paradigm, and considering also pedestrian and vehicular scenarios.

In general, the message diffusion using opportunistic networks depends mainly on node mobility and the specific protocols used. The high levels of node mobility in pedestrian and vehicular scenarios are responsible for a highly dynamic network topology. In general, these mobile nodes exhibit irregular contact patterns and short contact durations, reducing the effectiveness of message diffusion.

Surprisingly, most of the evaluations of the proposed routing algorithms do not consider the overall communication time. When a contact occurs, the amount of information transmitted is bounded by (i) the connection setup time, (ii) the actual data rate for the established link, and (iii) the contact duration. The setup time necessary to establish a connection between a pair of nodes which precedes the transmission of messages has usually been dismissed, although we will show in this paper its significant impact on information diffusion.

It has been stated that contact duration follows an exponential law distribution [4, 5]. In this paper, we have studied the distribution of **the contact duration using real traces confirming its exponential distribution**, meaning that most contact durations are relatively short compared to the longer ones, so large messages have lower opportunities to be transmitted. Furthermore, the problem, in that case, is that the message transmission may have started, but as it is aborted when the contact ends, causing message forwarding to fail, and so it needs to be re-transmitted in future contacts. This results in a waste of communication link capacity and energy [6]. Therefore, the diffusion of large messages can be improved if they are divided into smaller parts or “chunks”, avoiding (or minimizing) forwarding failures. In a previous paper, we proposed the *Xpread* protocol [7], which is based on splitting a message into smaller parts (chunks) that are transmitted using an epidemic diffusion. Then, only when a node receives all parts of the message, **the message** is considered to be successfully transmitted. In this paper, we further investigate how to optimise message broadcasting using *Xpread* protocol. assessing it with more complex and diverse realistic scenarios.

The efficiency of the proposed *Xpread* protocol will depend mainly on the right selection of the message partition scheme, that is, the number and sizes of the partitions. Therefore, it is necessary to evaluate the performance of this protocol under different partitioning patterns. To this end, we developed an analytical model based on population processes in which we consider several key aspects such as contact duration, communication time, and intensity of contacts. We based our analytical models on *Population Processes*, which have been widely used to evaluate the epidemic diffusion of messages in mobile networks [8, 9, 10]. Using this model, we determined that the best partition scheme for the *Xpread* protocol is the fixed size chunk, also obtaining an expression to obtain its optimal size. With this partitioning scheme, we can obtain diffusion improvement factors up to two times when compared to the classic epidemic protocol.

We validate the efficiency of the *Xpread* protocol using four different mobility traces, comprising both pedestrian and vehicular scenarios. The experiments show that *Xpread* increases the diffusion of messages with a slight reduction of the delivery time, and a significant reduction of the overhead per message successfully transmitted, compared to the classic epidemic protocol. As expected, the broadcasting of large messages is particularly improved, reducing the number of unsuccessful transmissions. Furthermore, we state the importance of considering the communication setup time when analysing the performance of these types of protocols.

The paper is organised as follows. Section 2 provides an overview of related works. Section 3 is devoted to studying the factors affecting message diffusion, which is the underpinning for the *Xpread* protocol that is described in Section 4. Then, we propose an analytical model to evaluate and optimise this protocol in Section 5. In Section 6 we further evaluate *Xpread* using simulation and real traces. Finally, conclusions are drawn at the end of the paper.

2. Related work

The most common approach to increase the broadcasting of messages is through message replication (also known as flooding). Replication can be uncontrolled, such as the Epidemic approach [11], or controlled such as *Spray and Wait* [12], where a limit is set to the number of messages copies in the network. Nevertheless, replication schemes (flooding) can consume a lot of network resources, so several proposals, such as probabilistic, counter-based, distance-based, location-based, and cluster-based, have been designed to reduce the flooding problem [13, 14]. These approaches are particularly interesting for vehicular networks, where new routing algorithms based on vehicle trajectories [15, 16], geographical information [17, 18, 19], and social behaviour [20, 21, 22] have been proposed.

It has been stated that large message transmission depends on contact duration. Nevertheless, in most cases, these temporal aspects have been neglected, considering only small message transmission. The first study of this dependence in vehicular networks is shown in [23], experimentally evidencing that there is a maximum message size limit to be exchanged depending on the vehicle's relative speed. Therefore, if a node needs to transmit larger messages it has no other alternative than to wait for a longer contact opportunity (for example, reducing its relative speed), or to use some kind of fragmentation. Contact duration can be also considered in the design of routing protocols, such as the proposal in [24], that considers both message transmission time and contact-duration in order to select the next forwarding node. However, this approach requires an estimation of future contact durations in order to be effective.

Another approach, is message fragmentation, in order to increase message diffusion and reduce unsuccessful transmissions. For store-and-forward networks, fragmentation was already suggested

in the late 1980s [25] for overcoming limited communication and storage resources. Furthermore, an early form of message fragmentation was introduced in the DTN design [26]. Fragmentation in OppNets is proposed in [27], considering two alternatives: proactive fragmentation, where a node splits a message prior to transmitting it; and reactive fragmentation, where if a transmission is interrupted, the message is divided accordingly. Fragmentation is also proposed in [28], where the authors propose a protocol that avoids forwarding failure by dividing messages into smaller fragments. The authors present a mathematical model based on random mobility models such as RWP (Random Way Point) and Markov processes to derive the optimal fragment size that minimises message delivery delay. A similar approach is described in [29], where the authors propose a single-copy fragmented data routing strategy (FDR) that performs well with short contact durations.

Some works have also explored the use of message redundancy by splitting a message using erasure codes, in order to improve the delivery ratio. This approach was initially studied in [30], formulating the problem of improving message delivery as an optimisation problem in order to obtain the optimal number of redundant copies. A similar approach was also introduced in [31]. The authors in [32] propose using limited replication (Spray-and-wait) and redundancy (using erasure-code) to improve the diffusion of a message. The paper proposes a method for obtaining the optimal replication and redundancy values given a set of networks parameters using an RWM model of the mobility of the nodes. Finally, the application of Raptor-codes for vehicular networks is proposed in [33] as a way to improve the efficiency and reliability of information broadcasting, while in [34] the rateless property of Fountain codes is adopted for message dissemination in vehicular environments, by means of Luby Transform codes optimized for small message length.

Nevertheless, erasure codes and other forms of redundancy have several drawbacks: firstly, they increase the number of messages and the generated volume of information in the network, thus contributing to network congestion and local buffer consumption; secondly, the processes of splitting a message, and, upon reception of the required parts, the reconstruction of the original message can be computationally intensive and time-consuming tasks, increasing the local consumption of resources.

We based our performance model on well-known models for the spreading of human epidemic diseases, *e.g.* SIR (Susceptible, Infectious and Recovered) models. For example, the authors of [10] introduced a mathematical approach for the diffusion of messages in opportunistic networks using the Epidemic protocol. One of the main conclusions of their analysis (mathematical model and its respective simulation) is that SIR models are quite accurate for the average behaviour of Epidemic DTNs. In [35] the authors proposed a detailed analytical model to analyse the epidemic information dissemination in mobile social networks. It is also based on SIR models including rules that concern users' behaviour, especially when their interests change according to the information type, and it can have a considerable impact on the dissemination process. After large simulations, they have demonstrated the accuracy of their model. Differently from the aforementioned proposals, other authors had instead a contact-time perspective and relied on different mathematical methods. The authors of [36] present an analytical study describing the performance of the Epidemic protocol, arguing that intermittently-connected mobile networks can be modelled as edge-Markovian dynamic graphs. They propose a new model for epidemic propagation on such graphs and calculate a closed-form expression that links the best achievable delivery ratio to common opportunistic network parameters such as message size, maximum tolerated delay, and link lifetime. In this research, they have shown that, given a certain maximum delay and node mobility,

bundle size has a major impact on the delivery ratio. Nevertheless, these previous models do not take into account several social aspects that influence the performance of message dissemination, such as the user density, and the dynamics of users arriving and leaving a place.

3. Contact time characterisation

145 This section is devoted to studying one of the main factors that affect the diffusion of mes-
sages: contact time. The goal is to obtain an expression of the contact time that will be used to
determine the probability that a message can be successfully transmitted. We study two different
scenarios: a vehicular scenario, particularly the diffusion of information among vehicles in a city,
and a pedestrian scenario, where the diffusion takes place in a metro station. In both scenarios,
150 the information is initially transmitted by some fixed nodes (or RSUs in VANETs terminology)
deployed throughout the city and delivered to pedestrian and vehicles directly or indirectly. Since
the efficiency of this diffusion is mainly affected by the opportunity of contacts, we start evaluating
the contacts intensity and the distribution of their duration.

A contact is defined as an opportunity for transmission between a pair of nodes, which can
155 be used to exchange messages. Since the target scenarios have both fixed and mobile nodes, we
consider two different sets of contacts *i.e.*, (i) the contacts between fixed and mobile nodes, and
(ii) the contacts between two mobile nodes. As shown in this section, both sets of contacts have
different characteristics, which [justify](#) their differentiation.

Throughout this paper, four different mobility traces are used as detailed below:

- 160 • **ROMA**: it is a vehicular taxi trace, which contains mobility traces of approximately 320
taxi cabs in Rome, Italy [37], collected between February 1st, 2014 and March 2nd, 2014.
The trace consists of a dataset of GPS coordinates, centered near the Coliseum and covering
an area of $100km \times 100km$ of Rome. The fixed nodes have been placed on two taxis ranks
(“Roma Termini Station” and “Piazza Venezia”). Using this trace, we obtained the contacts
165 between nodes using a range of 50m (which corresponds to a typical Wi-Fi range) for one day
(February 4th, 2014). Using a one day trace allows us to evaluate the diffusion of messages
depending on the different traffic densities throughout a day.
- **STATION**: it is a pedestrian mobility trace, which was generated from the movement of
170 pedestrian in a subway station in Valencia, Spain [38]. This trace corresponds to a one hour
interval (14:00 to 15:00) taken on Thursday, December 7th, 2017. The fixed nodes are located
at the turnstiles areas, at the entries of the station. Finally, the contacts were generated using
a communication range of 8m (a typical short-range Bluetooth connection).
- **SFO**: it is also another well-known taxi trace [39]. This trace contains the GPS coordinates
of approximately 500 taxi cabs collected over 24 days in May 2008, in the San Francisco Bay
175 Area, USA. The fixed nodes were located on the taxicab company’s parking lots. As in the
ROMA trace, we obtained the contacts between nodes using a range of 50m for one day (May
18th, 2008).
- **NCCU**: it is a pedestrian trace, which comes from an experiment at the NCCU campus [40],
where GPS position data were collected during two weeks (336 h) using an Android app
180 installed in the smartphones of 115 students. As in the STATION trace, we also consider a
communication range of 8 m for obtaining the contacts, and the fixed points were placed in
two classrooms.

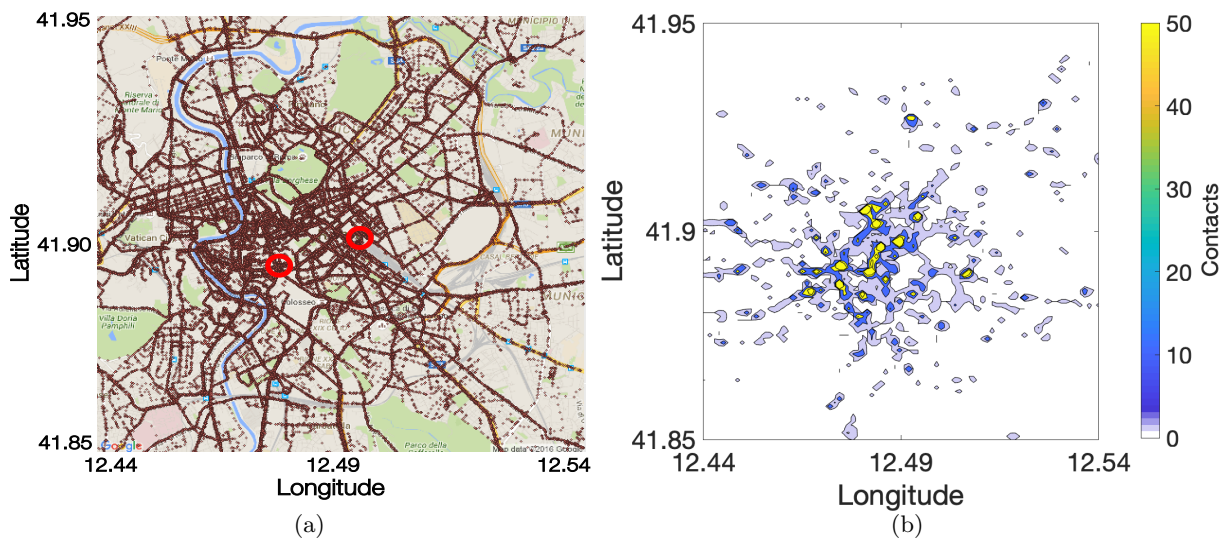


Figure 1: Spatial distribution of contacts of the ROMA scenario. (a) Map of the studied area. The red circles represent the fixed nodes and the brown points the vehicles locations (b) Contour map of the number of contacts per km^2 . Note that the upper range limit represents locations with 50 or more contacts.

The diversity of these scenarios, comprising both vehicular and pedestrian scenarios, allows us to evaluate in general the impact of contact duration on message transmission and therefore, the application of our proposed protocol.

In this section, we focus our study on the first two mobility traces. Particularly, Figure 1a shows a map of the considered area of the ROMA trace with the movement trace of the vehicles. For the STATION trace, Figure 2a shows a photo of the studied station and the considered area. In both scenarios, two fixed nodes are located in places with a high opportunity of contacts. These fixed nodes are used as the source of the information (messages). In the ROMA scenario, the two fixed nodes have been placed in the locations with a larger influx of taxis, that is, locations with a large number of contacts, as well as being distributed throughout the 24 hours (particularly at night). On the STATION scenario, the fixed nodes are located at the turnstiles areas (one in each edge of the station), that is considered the most convenient place, as all people must pass through the turnstiles. These fixed nodes are shown in Figure 1a and 2a as red circles.

Basically, the two most important aspects that impact the diffusion of information are the intensity of the contacts and their duration. These factors are mainly affected by the mobility of the nodes and their communication range. Firstly, for analytical models as the one devised in Section 5, the intensity of contacts between pairs of nodes is represented by the contact rate [41, 8, 10, 35], which can be easily obtained from traffic traces [42, 43]. Particularly, we use two contact rates *i.e.*, (i) λ that is the contact rate between pairs of mobile nodes, and (ii) Λ that is the one between mobiles nodes and fixed nodes¹. The obtained values for the evaluated traces are presented in Table 1. For both scenarios, the contact rates between fixed nodes and mobile nodes Λ are higher than the contact rates λ between only mobile nodes; in particular, for the STATION it scenario is

¹Note that both λ and Λ refer to the average contact rate between pairs of nodes, and not between all nodes, so these contact rates do not depend on the nodes density.

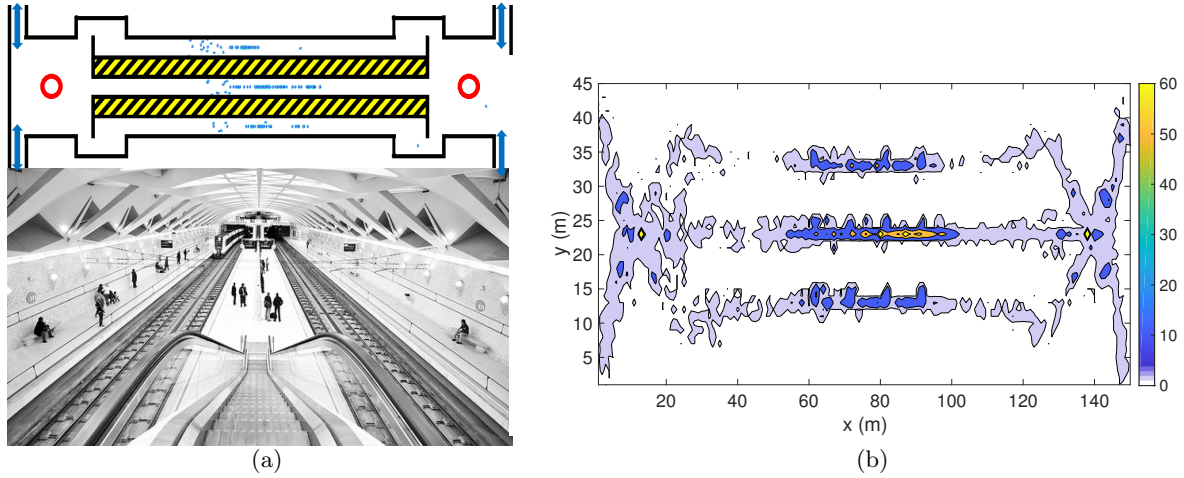


Figure 2: Spatial distribution of contacts of the STATION scenario (a) Map (on the top) and photo of the subway station. The red circles are the fixed nodes and the blue arrow the four entrance gates. The small blue points represent the pedestrians. (b) Contour map of the number of contacts per m^2 . The number of contacts has been truncated to 60, although there are some locations with higher contacts.

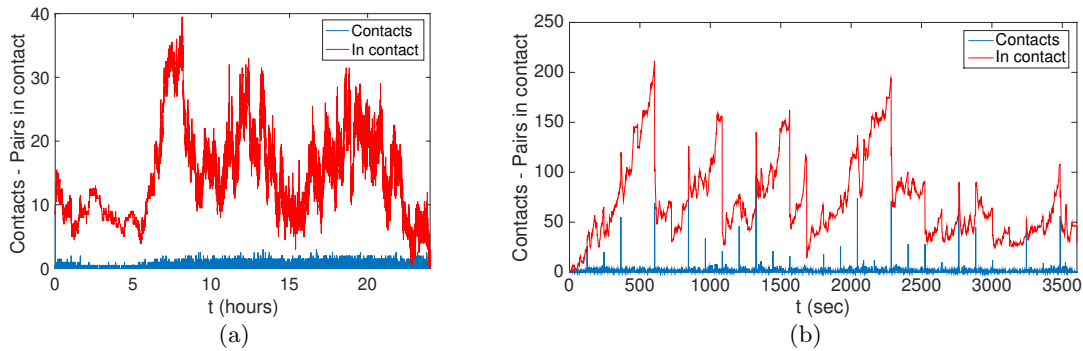


Figure 3: Number of contacts and pair of nodes in contact along the day. (a) ROMA Scenario (b) STATION Scenario.

205 tenfold higher as nearly all pedestrians have to pass through the turnstiles.

210 Firstly, we study the spatial and temporal distribution of the contacts. For the ROMA scenario, we can see in Figure 1b a contour map that represents the accumulated number of contacts per km^2 . As expected, the contacts are concentrated in the city centre where taxi circulation is greater. Regarding their temporal distribution, we can see in Figure 3a the number of newly started contacts (*blue line*) and the number of active contacts between pairs of nodes (*red line*). The temporal pattern reflects the activity in a common workday, with a low number of contacts at night, and several peaks along daytime. For the STATION scenario, we can distinguish in Figure 1b the pedestrians paths, where the number of contacts is low (and of short duration), and the high number of contacts concentrated in the station platforms, where the pedestrians stay and wait for a while waiting for the train arrivals. Furthermore, these contacts have long durations. Figure 3b plots the number of contacts depending on time, where we can clearly see the train arrivals: the number of contacts steadily increases before a train arrival as people are waiting on the platforms, and upon train arrivals, the number of contacts falls dramatically as people leave the platforms.

215

	ROMA	STATION	SFO	NCCU
Type	Vehicular	Pedestrian	Vehicular	Pedestrian
Nodes	319	263	534	115
Duration [h]	24	1	24	336
Contacts	22,853	5,618	343,485	7,062
λ	3.31×10^{-6}	2.75×10^{-6}	13.96×10^{-6}	10.15×10^{-6}
Λ	6.21×10^{-6}	36.39×10^{-6}	27.42×10^{-6}	27.20×10^{-6}
Pareto fit (mobile nodes)				
α, x_{min}	0.91, 5.88	0.78, 2.2	0.54, 3.72	0.88, 3.12
Mean [s]	108.8	40.36	201.2	128.2
Median [s]	14	5	15	17
Pareto fit (mobile-fixed nodes)				
α, x_{min}	0.88, 4.5	0.71, 2.5	0.52, 3.5	4.5, 0.75
Mean [s]	41.76	10.28	484.08	89.97
Median [s]	12	4	22	21

Table 1: Main contacts parameters of the evaluated scenarios.

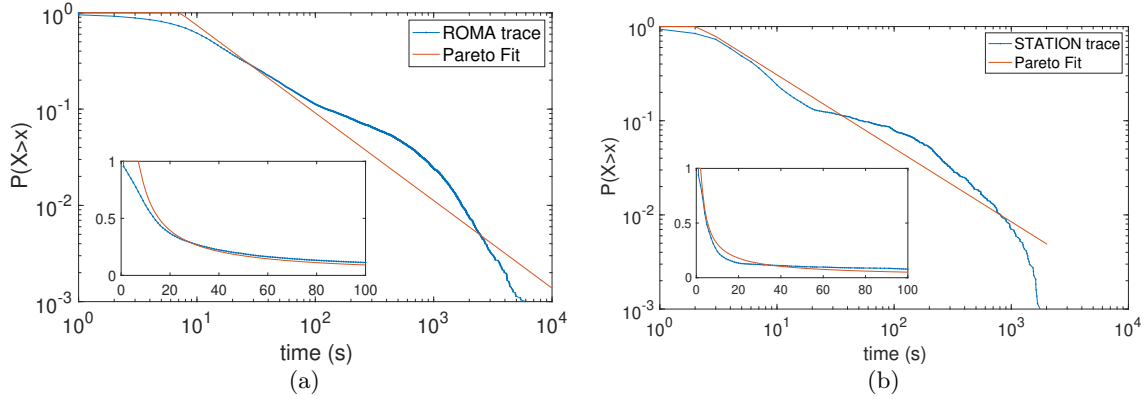


Figure 4: cCDF of the contact durations among mobile nodes for the ROMA scenario (a), and for the STATION scenario (b). The external plot is a log-log scale, while the internal plot is linear from 0 to 100 s.

Secondly, regarding the duration of the contacts, it has been established that it follows a power law distribution [4, 5], so the probability for a contact of duration x can be expressed as $p(x) = Cx^{-\alpha}$ for $x \geq x_{min}$. We confirmed this fact by analysing the contact durations using the above ROMA and STATION traces. The results for the contact between mobile nodes are shown in Figure 4 as a complementary Cumulative Distribution Function (cCDF), which is the probability that a contact duration X is greater than a given time x , *i.e.*, $P(X > x)$. Through the use of a cCDF plot with a log-log scale, it becomes easy to identify the power law nature of contact durations. Moreover, we can observe a significant proportion of large contact durations, which we identified as the contacts that some taxis have when waiting at taxi ranks in the ROMA scenario, or when pedestrians are waiting on the station platforms.

We can use a Pareto distribution with shape parameter α and scale parameter x_{min} for fitting these contact durations. Its cCDF is defined as:

$$\bar{F}_X(x) = P(X > x) = \begin{cases} \left(\frac{x_{min}}{x}\right)^\alpha & x \geq x_{min} \\ 1 & x < x_{min} \end{cases} \quad (1)$$

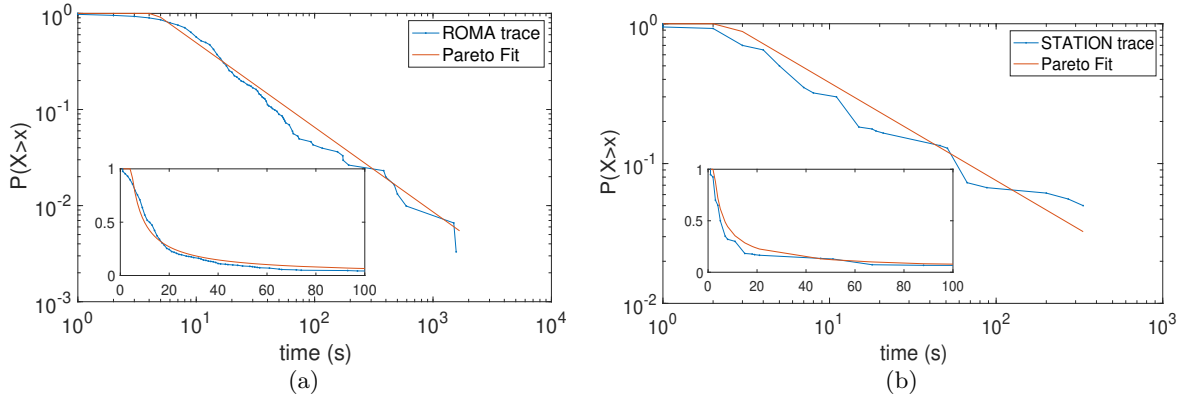


Figure 5: cCDF of the contact durations between fixed and mobile nodes for the ROMA scenario (a), and for the STATION scenario (b).

The results of fitting the contact duration in both scenarios are in Table 1, which are also plotted in Figure 4. The plots show that the Pareto Distribution is fairly fitted for both small and long durations. Comparing between scenarios, we can see that contact durations in the STATION scenario are shorter than in the ROMA scenario. Finally, since a Pareto distribution has only a well-defined mean when $\alpha > 1$, instead, we will use the median to obtain a better estimator for a typical contact duration. These values are also shown in Table 1.

Additionally, we also studied the distribution of contact durations between mobile nodes and fixed nodes. The results are shown in Figure 5, where we can see, as expected, that the duration distribution is different from the previous ones, and in general, contact durations have shorter durations, further limiting the transmission of larger messages. Regarding the other two traces, SFO and NCCU, the results obtained are very similar to the ones described in this section, confirming their power law behaviour (see the Pareto fitting parameters in Table 1).

In conclusion, we have shown that most of the contacts have short durations, and so the opportunity to successfully transmit large messages is reduced. In addition, the proposed Pareto distribution fitting will be used to obtain the probability of successfully transmitting a message in the proposed model.

4. Improving message diffusion: the Xpread protocol

This section studies how the diffusion of large messages can be improved. In the previous section, we have seen that most contact durations are short, reducing the successful transmission of these large messages. For example, in the ROMA scenario, the probability for transmitting a message that requires a transmission time of at least 14 s (the median) will be 50% for mobile-to-mobile contacts and approximately 42% for fixed-to-mobiles contacts. In the STATION contacts, these probabilities are reduced to approximately 17% and 15%, respectively.

The transmission time of messages depends mainly on the message size but also on the available bandwidth. Formally, the communication time required for transmitting a set of n messages during the same contact can be expressed as follows:

$$t_c = t_s + \frac{1}{Bw} \sum_{i=1}^n m_i \quad (2)$$

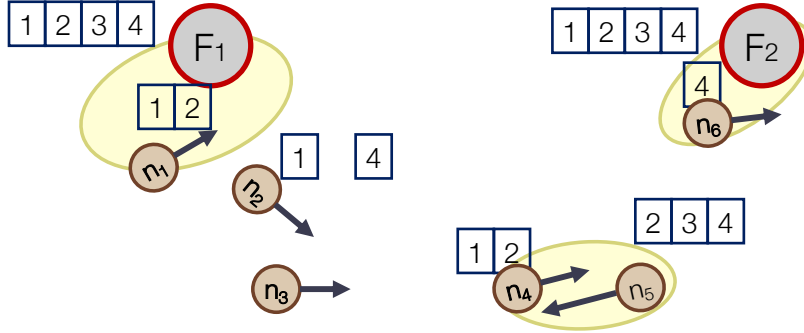


Figure 6: Example of how *Xpread* works, when the original message is divided into four chunks. The node's boxes represent the chunks that the node has received. The arrows represent the movement direction of the nodes.

This expression comprises two components *i.e.*, (i) a fixed setup time t_s for two nodes to establish a connection, and (ii) the sum of the transmission time of every message ($t_t = m/Bw$), which depends on the message size (m [bit]) and the available bandwidth (Bw [bit/s]). Therefore, given
 255 a communication time t_c , we can obtain from the contact duration cCDF the likelihood that a message can be successfully transmitted as $P(X > t_c)$. As expected, bigger messages will have far lower chances to be transmitted since the contact duration follows a power law distribution.

Consequently, to increase the likelihood of the diffusion of these large messages, we propose the *Xpread* protocol as an improvement of the classic *Epidemic* protocol, which is based on fragmenting
 260 the original message into smaller parts in order to have more opportunities to be transmitted.

Basically, *Xpread* works as follows. In Figure 6 the sender initially divides the original message into several smaller parts (from now on, *chunks*), with a convenient identification. These chunks are redirected to the fixed nodes, assuring that the chunks are always transmitted in a different
 265 order to assure fairness in the diffusion of all message parts. Then, each mobile node is responsible for receiving, storing and relaying the chunks, as if it was the standard *Epidemic* protocol. Nevertheless, only when all chunks are received, it will be considered that the message was successfully received. For example, in Figure 6 the message is divided into four chunks. A contact between fixed node one (F_1) and node one (n_1) ends with the successful transmission of chunks #1 and #2. On the other side, F_2 only has enough time to transmits the chunk #4 to n_6 . Then, from the state
 270 represented in the figure, a future contact between nodes 4 and 5 could transmit chunk #1 from n_4 to n_5 , and chunks #3 and #4 from n_5 to n_4 , so both nodes would have all the chunks. Then, the original message could be considered successfully delivered. This way, dividing the original message into smaller parts will allow more successfully chunk transmissions, thereby increasing the delivery rate as it will be shown in the evaluation section.

The message partitioning scheme depends on several factors that are evaluated in the following
 275 sections. We consider two different partition schemes *i.e.*, (i) *fixed-size* chunks, in which all messages are divided into equally sized chunks², and (ii) *fixed-number* of chunks, in which a message is divided into a fixed number of chunks. For example, considering a 10 MB-message, for a 1 MB *fixed-size* chunk, it will be divided into 10 chunks, and, for a 5 *fixed-number* of chunks, it will be divided into 5 chunks of 2MB each one. Note that, in the *fixed-number* scheme, small messages
 280 are divided into 5 chunks of 2MB each one.

²If the message size or the last chunk size is smaller than the fixed size, the chunk size is reduced, thus avoiding internal fragmentation

are also divided, but this is not the case for the *fixed-size* scheme, in which messages lower than the defined size will not be divided. Additionally, for each scheme, we must select the number of chunks or the chunk size that optimises the diffusion, which will depend mainly on the distribution of the contact durations, as detailed in the following sections.

285 The *Xpread* protocol implementation does not require any modification to the transport or network layers. Note that, in these layers, a large message is already fragmented into small packets to fit into the network MTU (Maximum Transmission Unit) to improve the transmission performance. Nevertheless, the reception of the message is not successful until all the fragments are received.

5. Analytical modelling and evaluation

290 This section introduces the analytical model used to evaluate the dissemination of a message considering key aspects such as message size, communication time and contact duration, which are required to determine the best partitioning scheme for our *Xpread* protocol. We consider a scenario with F fixed nodes, and N mobile nodes (*i.e.*, pedestrians or vehicles). The model is based on *population processes or models*, which is a mathematical tool used in population dynamics
295 to model the spread of parasites, viruses, and diseases [44]. Specifically, our model is based on *biological* epidemic models, where individuals can be infected by other nodes when a contact occurs. In our model, an infected node refers to a node that has the message, and an infection refers to the process whereby a node that has the message transmits it to another node.

The most basic epidemic model (see [8]) considered N nodes moving freely within a closed area
300 with a given effective contact rate between pairs λ . In this basic model, nodes are classified into two classes: the susceptible nodes S (nodes that do not have the message), and the infected nodes \mathcal{I} (nodes with the message). The dynamics of this system can be expressed using a deterministic model based on ODEs (Ordinary Differential Equations). Nevertheless, this model does not take into account the transmission time nor the probability of successful transmission.

305 Therefore, this simpler model should be extended to evaluate the diffusion of a message when it is divided into two chunks, and taking into account the transmission time and contact duration. This new model is based on some of the previous models introduced in [45, 46].

5.1. *Xpread* Diffusion Model

In this subsection, we propose a new model for evaluating the diffusion of information taking into
310 account the partitioning of messages with the aim of evaluating the *Xpread* protocol. Specifically, we firstly devise a model where the message is divided into two halves. Then, this model can be extended to consider the division of a message into a variable number of parts, as shown in the Appendix.

The area is populated with N mobile nodes and F fixed nodes with two defined contact rates: λ
315 and Λ . Considering communication time, when an infected/fixed node contacts a susceptible node, both nodes establish a connection and a message can start its transmission. If communication time is lower than the contact duration t_c , then the message is successfully transmitted; otherwise, the transmission is aborted when the contact time ends. Furthermore, during the communication time t_c (or when t_c is higher than the contact duration, during the median contact duration), the nodes
320 involved cannot infect other nodes, so several new classes of nodes must be created.

The classes of the epidemic model are detailed in Figure 7. Besides the common susceptible \mathcal{S} , and infected \mathcal{I} classes, two new classes of communicating nodes are created *i.e.*, (i) the \mathcal{C}_f class comprises all fixed and susceptible node pairs that are communicating, and (ii) the \mathcal{C}_{n1} and \mathcal{C}_{n2}

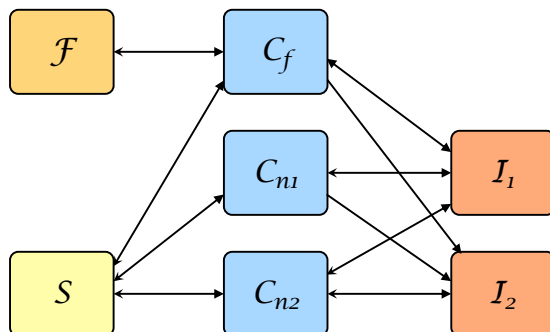


Figure 7: Graphical representation of the transitions between classes for the division into two-parts model.

325 classes include the infected and susceptible node pairs communicating. Additionally, since infected
 330 **mobile** nodes can have one or two of the chunks of the message, we further classify the infected
 nodes into two classes *i.e.*, \mathcal{I}_1 and \mathcal{I}_2 , namely, the nodes that have one or two chunks, respectively.
 Note that only the nodes that are in class \mathcal{I}_2 are considered to have received the whole message.
 Nodes that are in class \mathcal{I}_1 have one chunk and need to contact again to get the other chunk. With
 regard to the communication classes, the \mathcal{C}_n class is divided into two newer classes, \mathcal{C}_{n1} and \mathcal{C}_{n2} , to
 335 take into account the nodes that are communicating from classes \mathcal{I}_1 and \mathcal{I}_2 , respectively. Finally,
 the \mathcal{F} class includes the available fixed nodes that are not communicating. Note that, while \mathcal{C}_f ,
 \mathcal{C}_{n1} , and \mathcal{C}_{n2} classes include those nodes that are not available for contacting with other nodes, the
 \mathcal{I}_1 , \mathcal{I}_2 , \mathcal{S} , and \mathcal{F} classes include the nodes that are available for starting a new transmission, with
 their associated contact rates.

335 The time that nodes stay in the communication classes \mathcal{C}_f , \mathcal{C}_{n1} , and \mathcal{C}_{n2} depends on whether the
 message can be successfully transmitted or not, **which are finally a function of the contact duration
 and the message size**. In order to model this communication time, we consider the probability of
 transmitting a message with a given communication time using the fitted contact-duration Pareto
 distributions, as detailed in Subsection 3. **In particular**, we used two cCDFs *i.e.*, (i) $\bar{F}_n(x)$, for
 340 contact durations between mobile nodes, and (ii) $\bar{F}_f(x)$, for contact durations between fixed and
 mobile nodes.

Now, **having** defined the probabilities of transmitting successfully a message, we must obtain
 the **different** communication times required to transmit the whole message or its chunks: t_{cw} stands
 for the communication time **needed for** the whole message, and t_{c1} refers to the communication
 345 time **for one chunk**. If t_w is the whole message transmission time, and considering the setup time
 t_s , then $t_{cw} = t_s + t_w$ and $t_{c1} = t_s + t_w/2$. These two communication times also imply two different
 values for the ratios of successful message transmission, which will depend on the cCDFs of the
 contact durations. For the contacts among mobile nodes, $p_{nw} = \bar{F}_n(t_{cw})$ and $p_{n1} = \bar{F}_n(t_{c1})$ stand
 for the ratio of successful transmission regarding the whole message itself and for one of its chunks,
 350 respectively; contacts between fixed and mobile nodes are obtained in a similar way, but using the
 other distribution *i.e.*, $p_{fw} = \bar{F}_f(t_{cw})$ and $p_{f1} = \bar{F}_f(t_{c1})$, respectively.

From the transmission probabilities we can obtain the ratio of successful transmissions, which
 will be $p_{nw}\lambda$ and $p_{n1}\lambda$ for the contact between mobiles nodes, implying that both nodes stay in
 classes \mathcal{C}_{n1} and \mathcal{C}_{n2} for times t_{cw} and t_{c1} . On the other hand, $(1 - p_{nw})\lambda$ and $(1 - p_{n1})\lambda$ will be
 355 the ratio of unsuccessful transmissions, with a waiting time of t_n , which is the median and can be

considered the typical contact duration³. Accordingly, in the case of contacts between fixed and susceptible nodes, we assume the same communication times (t_{cw} and t_{c1}) defining the successful transmission ratio as $p_{fw}\Lambda$ for the whole message and $p_{f1}\Lambda$ for one of its chunks. In both cases the waiting time for unsuccessful message transmissions is t_f , which is the median of the transmission
360 time between mobile and fixed nodes.

Summing up, to design our model, we take into account the following transitions (see Figure 7):

- $\mathcal{F} \rightarrow \mathcal{C}_f$: fixed nodes that contact with susceptible nodes to start a new communication.
- $\mathcal{S} \rightarrow \mathcal{C}_f$: susceptible nodes that contact with fixed nodes to start a new communication.
- $\mathcal{S} \rightarrow \mathcal{C}_{n1}$: susceptible nodes that contact with infected nodes that have only one part of the
365 message (class \mathcal{I}_1) to start a new communication.
- $\mathcal{S} \rightarrow \mathcal{C}_{n2}$: susceptible nodes that contact with infected nodes that have both chunks (class \mathcal{I}_2) to start a new communication.
- $\mathcal{I}_1 \rightarrow \mathcal{C}_f$: infected nodes with one chunk that contact with fixed nodes to start a new communication to obtain the second chunk.
- $\mathcal{I}_1 \rightarrow \mathcal{C}_{n1}$: infected nodes with one chunk that contact with susceptible nodes to start a new
370 communication.
- $\mathcal{I}_1 \rightarrow \mathcal{C}_{n2}$: infected nodes with one chunk that contact with nodes that have the two chunks (\mathcal{I}_2) to start a new communication to obtain the second chunk.
- $\mathcal{I}_2 \rightarrow \mathcal{C}_{n2}$: infected nodes with two chunks that contact with susceptible nodes or nodes with
375 one chunk \mathcal{I}_1 to start a new communication.
- $\mathcal{C}_f \rightarrow \mathcal{F}$: fixed nodes ending a communication.
- $\mathcal{C}_f \rightarrow \mathcal{S}$: susceptible nodes that do not get infected since the contact time with the fixed node is less than the communication time required to transmit at least one of the chunks
- $\mathcal{C}_{n1} \rightarrow \mathcal{S}$: the same as $\mathcal{C}_f \rightarrow \mathcal{S}$, but for contacts with infected nodes with one chunk.
- $\mathcal{C}_{n2} \rightarrow \mathcal{S}$: the same as $\mathcal{C}_f \rightarrow \mathcal{S}$, but for contacts with infected nodes with two chunks.
- $\mathcal{C}_f \rightarrow \mathcal{I}_1$: susceptible nodes that receive only one part of the message from a contact with a fixed node.
- $\mathcal{C}_f \rightarrow \mathcal{I}_2$: susceptible nodes that receive both parts of the message from a contact with a fixed node.
- $\mathcal{C}_{n1} \rightarrow \mathcal{I}_1$: infected nodes **with only one chunk that end** their transmission, and newly infected nodes that successfully get only one part of the message.
- $\mathcal{C}_{n1} \rightarrow \mathcal{I}_2$: infected nodes **with only one chunk that end** their transmission, and newly infected nodes that successfully get both **parts** of the message.

³In case $t_n > t_c$, the waiting time will also be t_c .

$$\begin{aligned}
\mathcal{F}' &= -\Lambda\mathcal{F}\mathcal{S} & +p_{fw}\Lambda\mathcal{S}\mathcal{F}(t_{cw}) & & +(1-p_{fw})p_{f1}\Lambda\mathcal{S}\mathcal{F}(t_{c1}) & & +(1-p_{fw})(1-p_{f1})\Lambda\mathcal{S}\mathcal{F}(t_f) \\
& -\Lambda\mathcal{F}\mathcal{I}_1 & & & +p_{f1}\Lambda\mathcal{I}_1\mathcal{F}(t_{c1}) & & +(1-p_{f1})\Lambda\mathcal{I}_1\mathcal{F}(t_f) \\
\mathcal{S}' &= -\Lambda\mathcal{S}\mathcal{F} & & & & & +(1-p_{fw})(1-p_{f1})\Lambda\mathcal{S}\mathcal{F}(t_f) \\
& -\lambda\mathcal{S}\mathcal{I}_1 & & & & & +(1-p_{n1})\lambda\mathcal{S}\mathcal{I}_1(t_n) \\
& -\lambda\mathcal{S}\mathcal{I}_2 & & & & & +(1-p_{nw})(1-p_{n1})\lambda\mathcal{S}\mathcal{I}_2(t_n) \\
\mathcal{I}'_1 &= & & & +(1-p_{fw})p_{f1}\Lambda\mathcal{S}\mathcal{F}(t_{c1}) & & \\
& & & & +(1-p_{nw})p_{n1}\lambda\mathcal{S}\mathcal{I}_2(t_{c1}) & & \\
& -\Lambda\mathcal{I}_1\mathcal{F} & & & & & +(1-p_{f1})\Lambda\mathcal{I}_1\mathcal{F}(t_f) \\
& -\lambda\mathcal{I}_1\mathcal{S} & & & +2p_{n1}\lambda\mathcal{S}\mathcal{I}_1(t_{c1}) & & +(1-p_{n1})\lambda\mathcal{S}\mathcal{I}_1(t_n) \\
& -\lambda\mathcal{I}_1\mathcal{I}_2 & & & & & +(1-p_{n1})\lambda\mathcal{I}_1\mathcal{I}_2(t_n) \\
\mathcal{I}'_2 &= & +p_{fw}\Lambda\mathcal{S}\mathcal{F}(t_{cw}) & & & & \\
& & & & +p_{f1}\Lambda\mathcal{I}_1\mathcal{F}(t_{c1}) & & \\
& -\lambda\mathcal{I}_2\mathcal{S} & +2p_{nw}\lambda\mathcal{S}\mathcal{I}_2(t_{cw}) & & +(1-p_{nw})p_{n1}\lambda\mathcal{S}\mathcal{I}_2(t_{c1}) & & +(1-p_{nw})(1-p_{n1})\lambda\mathcal{S}\mathcal{I}_2(t_n) \\
& -\lambda\mathcal{I}_2\mathcal{I}_1 & & & +2p_{n1}\lambda\mathcal{I}_1\mathcal{I}_2(t_{c1}) & & +(1-p_{n1})\lambda\mathcal{I}_1\mathcal{I}_2(t_n) \\
\mathcal{C}'_f &= +2\Lambda\mathcal{F}\mathcal{S} & -2p_{fw}\Lambda\mathcal{S}\mathcal{F}(t_{cw}) & & -2(1-p_{fw})p_{f1}\Lambda\mathcal{S}\mathcal{F}(t_{c1}) & & -2(1-p_{fw})(1-p_{f1})\Lambda\mathcal{S}\mathcal{F}(t_f) \\
& +2\Lambda\mathcal{F}\mathcal{I}_1 & & & -2p_{f1}\Lambda\mathcal{I}_1\mathcal{F}(t_{c1}) & & -2(1-p_{f1})\Lambda\mathcal{I}_1\mathcal{F}(t_f) \\
\mathcal{C}'_{n1} &= +2\lambda\mathcal{S}\mathcal{I}_1 & & & -2p_{n1}\lambda\mathcal{S}\mathcal{I}_1(t_{c1}) & & -2(1-p_{n1})\lambda\mathcal{S}\mathcal{I}_1(t_n) \\
\mathcal{C}'_{n2} &= +2\lambda\mathcal{S}\mathcal{I}_2 & -2p_{nw}\lambda\mathcal{S}\mathcal{I}_2(t_{cw}) & & -2(1-p_{nw})p_{n1}\lambda\mathcal{S}\mathcal{I}_2(t_{c1}) & & -2(1-p_{nw})(1-p_{n1})\lambda\mathcal{S}\mathcal{I}_2(t_n) \\
& +2\lambda\mathcal{I}_1\mathcal{I}_2 & & & -2p_{n1}\lambda\mathcal{I}_1\mathcal{I}_2(t_{c1}) & & -2(1-p_{n1})\lambda\mathcal{I}_1\mathcal{I}_2(t_n)
\end{aligned} \tag{3}$$

$$\begin{aligned}
\mathcal{F}[i+1] &= \mathcal{F}[i] + h \cdot (& -\Lambda\mathcal{F}[i]\mathcal{S}[i] & +p_{fw}\Lambda\mathcal{S}[i-icw]\mathcal{F}[i-icw] & +(1-p_{fw})p_{f1}\Lambda\mathcal{S}[i-ic1]\mathcal{F}[i-ic1] \\
& & +(1-p_{fw})(1-p_{f1})\Lambda\mathcal{S}[i-if]\mathcal{F}[i-if] & -\Lambda\mathcal{F}[i]\mathcal{I}_1[i] & +p_{f1}\Lambda\mathcal{I}_1[i-ic1]\mathcal{F}[i-ic1] \\
& & +(1-p_{f1})\Lambda\mathcal{I}_1[i-if]\mathcal{F}[i-if] & & &)
\end{aligned} \tag{4}$$

- $\mathcal{C}_{n2} \rightarrow \mathcal{I}_1$: the same as $\mathcal{C}_{n1} \rightarrow \mathcal{I}_1$, but with nodes that contact with class \mathcal{I}_2 .
- $\mathcal{C}_{n2} \rightarrow \mathcal{I}_2$: the same as $\mathcal{C}_{n2} \rightarrow \mathcal{I}_1$, but with nodes that contact with class \mathcal{I}_2 .

From these transitions, we can model the dynamics of this system using a set of Delay Differential Equations (DDEs), as defined in the set equations of Eq. (3). For a more compact representation of these equations we omit the time *by* using the following notation for the multiplication of two classes \mathcal{A}, \mathcal{B} , with no shift, and with time shift t_x :

$$\begin{aligned}
\mathcal{A}\mathcal{B} &= \mathcal{A}(t)\mathcal{B}(t) \\
\mathcal{A}\mathcal{B}(t_x) &= \mathcal{A}(t-t_x)\mathcal{B}(t-t_x)
\end{aligned} \tag{5}$$

and finally, $\mathcal{A}' = \mathcal{A}'(t)$. Using this notation, for example $\mathcal{S}\mathcal{F}$ stands for $\mathcal{S}(t)\mathcal{F}(t)$, and $\mathcal{S}\mathcal{F}(t_{cw})$ stands for $\mathcal{S}(t-t_{cw})\mathcal{F}(t-t_{cw})$. Note also that $\mathcal{A}\mathcal{B} = \mathcal{B}\mathcal{A}$. The equations are arranged into five columns, where the third, fourth and fifth columns contain the transitions concerning transmissions of the whole message (two chunks), one new chunk, and unsuccessfully transmission, respectively.

As an example of transitions, a contact between fixed and susceptible nodes starts with nodes from classes \mathcal{F} and \mathcal{S} moving to the \mathcal{C}_f class with the rate $\Lambda\mathcal{S}\mathcal{F}$. Then, three possible transitions can occur depending on the ratio of successful transmissions *i.e.*, (i) p_{fw} is the ratio of the successful transmission of the whole message, requiring to stay in this class for time t_{cw} before moving to

classes \mathcal{F} and \mathcal{I}_2 ; (ii) $(1 - p_{fw})p_{f1}$ is the ratio for transmitting one chunk with stay time t_{c1} , and then moving to classes \mathcal{F} and \mathcal{I}_1 ; (iii) $(1 - p_{fw})(1 - p_{f1})$ is the ratio for no message transmission with stay time t_f , and then moving to classes \mathcal{F} and \mathcal{S} . The rest of the transitions are very similar to the one explained.

Finally, the system defined by Eq. (3) can be solved using a set of discrete equations with a step h , and then applying, for example, the Euler's method. The t_{cw} , t_{c1} , t_n and t_f delays can be expressed with a shift on the indexes ($i - icw$, $i - ic1$, $i - in$ and $i - if$), where $icw = \lfloor t_{cw}/h \rfloor$, $ic1 = \lfloor t_{c1}/h \rfloor$, $in = \lfloor t_n/h \rfloor$ and $if = \lfloor t_f/h \rfloor$. For example, the first equation \mathcal{F} in Eq. (3) can be easily transformed to Eq. (4), and so can the other equations, obtaining a set of seven discrete equations.

With the initial values of $\mathcal{S}_0 = N$, $\mathcal{F}_0 = F$ and zero for the other classes, we can obtain the evolution of the system iterating over i . With regard to the dynamics of this system, it is easy to observe that all nodes will get infected (i.e., will get a copy of the message) when t is long enough. Therefore, the final state will be $\mathcal{S}(t) = 0$, $\mathcal{I}_2(t) = N$, $\mathcal{F}(t) = F$, and the rest of classes 0.

Alternatively, if we want to consider the *TTL* (Time-To-Live) of a message, we need only to evaluate the diffusion up to time $t = TTL$, and obtain the diffusion results (i.e., the values of the classes at this time). Note that, after that time, since chunks are removed from the local buffers, the diffusion stops.

The two following metrics are used. The first one is the *diffusion ratio*, that is the proportion of infected nodes depending on time. It is important to note that, if we want to obtain the real number of infected nodes, we must consider, not only the nodes in class \mathcal{I}_2 , but also the infected nodes that are communicating, which are $\mathcal{C}_{n2}(t)/2$. Thus, the diffusion ratio R_d is:

$$R_d(t) = \frac{\mathcal{I}_2(t) + \frac{1}{2}\mathcal{C}_{n2}(t)}{N}. \quad (6)$$

The second metric is the *diffusion time* D_t , which is the time a message reaches all the nodes, and it can be obtained as the time t when $R_d(t) = 1$.

This model has been further extended to consider the division into a higher number of chunks. Following the previous pattern for generating classes and transitions, we created a generic model where we can evaluate the diffusion of a message that is divided into w chunks, using the Pareto cCDF for obtaining the ratio of successful transmissions, and the communication time equation Eq. (2) for obtaining the different communication times for transmitting from 1 to w chunks of the message. The extended model is detailed in the Appendix.

5.2. Diffusion Dynamics and Optimal Partitioning

The goal of this subsection is to evaluate the two main aspects required for the optimisation of the *Xpread* protocol: the partition scheme and the impact of the message size and setup time in the efficiency of the diffusion⁴. The evaluation is performed using the extended model described in the Appendix, which considers a variable number of partitions. In all the cases, the system is solved using the iterative Euler's method. In the following evaluations, we consider a network with $N = 300$ and $F = 2$ using the contact distributions and contact rates obtained from the ROMA trace (considering a 50 m range, and an average bandwidth of $Bw = 10$ Mbps, resembling a Wi-Fi transmission), as detailed in Section 3.

⁴The impact of factors such as the number of nodes, contact rate, etc. has been widely studied in previous works [8, 35, 46] and is not the object of this evaluation.

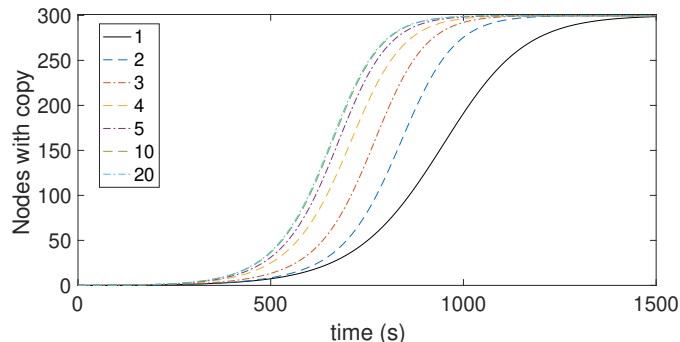


Figure 8: Diffusion of a 10 MB-message when there is no division (curve '1'), and when it is divided into 2, 3, 4, 5, 10 and 20 chunks (size of the chunks from 5 MB to 0.5 MB) for a setup time of 5 s.

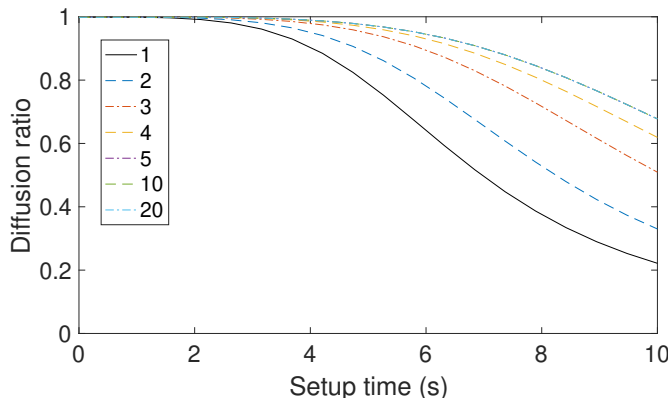


Figure 9: Ratio of diffusion of a 10 MB-message when there is no division (curve '1'), and when it is divided into 2, 3, 4, 5, 10 and 20 chunks (size of the chunks ranging from 5 MB to 0.5 MB) for $t = 1000$ s.

435 Firstly, we show in Figure 8 the dynamic of the diffusion of a message of 10 MB depending on
time (that is, the infected nodes) using different message partitions ranging from 1 (no division) to
20 (that is, the message divided into 20 chunks of 0.5 MB each), and considering a setup time of
 $t_s = 5$ s. In general, we can observe the typical epidemic diffusion, in which, after a critical point,
the message is spread very fast. As for the partitioning, there are significant improvements when
the message is divided up to 10 chunks, being the diffusion for 10 and 20 chunks mostly the same.
440 That is, chunks with a size below 1MB do not increase the diffusion effectiveness.

Secondly, we evaluate the impact of the setup time on the message diffusion. Figure 9 shows
the diffusion ratio of a 10 MB-message at time $t = 1000$ s for setup times from 0 to 10 s. We
clearly show the impact of the setup time on the diffusion ratio since higher setup times reduce
the available communication time for transmitting the message. As in the previous experiment,
445 increasing the division from 10 to 20 chunks did not improve the diffusion ratio.

Now, we focus our experiments on determining the best partitioning scheme. Thus, we per-
formed the following experiments: using the previous scenario, we obtained, for message
sizes ranging from 1 to 100 MB, and setup times of 1, 3, 5 and 10 s, the number of divisions that max-
imise the message diffusion (specifically, the diffusion ratio). The results are shown in Figure 10,
450 where we plot the obtained message size instead of the number of partitions in order to show how

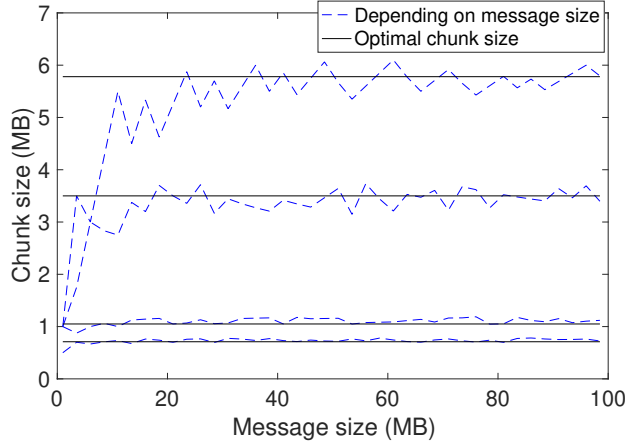


Figure 10: Chunk sizes that optimise the diffusion ratio depending on message size (dashed blue line), for setup times 1, 3, 5 and 10 s (from top to bottom). The black continuous line is the average chunk size of the different setup times.

the message size actually determines the improvement. We can see that, for very small message sizes, no partition is optimal, and so the chunk size equals to the message size. Nevertheless, for greater messages, the chunk size converges to a fixed size (the black lines) that depends solely on the setup time. For example, for a message of 20 MB, the best results were obtained for 4, 6, 19 and 29 chunks, that corresponds to 5 MB, 3.33 MB, 1.06 MB and 0.69 MB. Thus, the obtained average chunk sizes are 5.78 MB, 3.52 MB, 1.05 MB and 0.71 MB, which corresponds to a communication time of 5.84, 5.95, 5.88 and 10.58 s. We can see that, for setup times from 1 to 5 s, the communication time is around the scale parameter x_{min} of the contact duration Pareto fitting (for mobile nodes). This makes sense as reducing the contact durations below the x_{min} value for the Pareto distribution does not increase its probability of transmission.

We repeated this experiment with other fitting values from the traces of the other scenarios described in Section 3 and Table 1 obtaining similar findings. From all these experiments, we empirically found that in case the setup time is smaller than x_{min} , we can set a chunk size of $(x_{min} - t_s) \cdot Bw$ up to a minimum chunk size of approximately $0.05 \cdot Bw$; smaller chunk sizes show no significant improvements. Working out, t_s must be less than $x_{min} - 0.05$. In case the setup time is higher or equal to $x_{min} - 0.05$, the experiments showed that the chunk size can be set to $(0.05 \cdot t_s) \cdot Bw$. In short, the optimal chunk size can be obtained empirically as:

$$CS \approx \begin{cases} (x_{min} - t_s) \cdot Bw & \text{if } t_s < (x_{min} - 0.05) \\ (0.05 \cdot t_s) \cdot Bw & \text{if } t_s \geq (x_{min} - 0.05) \end{cases} \quad (7)$$

Thus, from the previous results, we can conclude that the best partitioning scheme is the fixed-size one, which can be obtained using the empirical Eq. (7). On the contrary, using a fixed-number of chunks will be inefficient for small size messages as no partition is optimal for these sizes. On the other hand, the (optimal) chunk size converges to a fixed size, so it would also be complicated to obtain the (fixed) number of divisions that makes the diffusion optimal.

Finally, we evaluated the impact of message size on diffusion effectiveness. Figure 11a shows the ratio of diffusion depending on message size, which ranges from 1 to 100 MB for different chunk sizes, and a 5 s setup time. Additionally, in order to identify the improvement, we also plotted in Figure 11b the improvement factor of partitioning the message against no partition (epidemic

diffusion). Firstly, it is clearly shown that the diffusion is improved when the chunk size is reduced up to 1 MB (that is, the optimal chunk size for this setup time). With regard to the message size, we can see that, for message sizes smaller than 30 MB, no improvements are obtained since epidemic diffusion reaches 100% of the nodes. From this value (*i.e.*, 30 MB), the improvement is increased by up to a factor of 1.8 for message sizes of about 100 MB, and a chunk size of 1 MB. The primary reason for this behaviour is the difference between the probability of transmission of the whole message and its chunks, which is maximized for values close to this message size. As for the diffusion time, we can see in Figure 11c a significant reduction when the chunk size is smaller. Note that, in this figure, the range of the message size is reduced to the values that allow us to obtain a full diffusion ratio.

To summarise, in this section, we have developed an analytical model that allows studying the dynamics of the message diffusion for the *Xpread* protocol. Using this model we have determined that the best partitioning scheme for the *Xpread* protocol is the fixed-size, also obtaining a simple expression to determine its optimal size. With this partitioning scheme, we can obtain diffusion improvements factors up to two, mainly when the epidemic protocol is not so efficient. These results have been also confirmed using the other mobility traces.

6. Performance evaluation

In the previous section, we evaluated analytically the dynamics and efficiency of the *Xpread* protocol finding also an optimal partition scheme. The great advantage of our model is that it allows evaluating in a fast and direct way the dynamics of the diffusion process. Nevertheless, due to some of the assumptions and simplifications performed, some of the results concerning real transmission diffusion cannot be evaluated. Therefore, the aim of this section is to evaluate experimentally the *Xpread* protocol. In Opportunistic Networks, this is usually performed by using simulators combined with real mobility traces [47]. Basically, this type of simulations are contact-based, that is, the simulator reads a trace with the positions of the nodes, detects the nodes that are in contact (that is, within a given range), and performs the exchange of messages between the contacting nodes. Finally, some metrics are obtained. This is the approach of well known opportunistic networks simulators such as the ONE [48] and OMNeT++ [49].

Based on these ideas, we have implemented a custom contact-based simulator with the aim to evaluate the diffusion of our proposed partition scheme. This simulator⁵, which can use the same mobility traces as the ONE simulator, implements the diffusion protocol, also considering the setup time and the diffusion time. In order to validate our simulator, we checked the results of transmitting one message using several scenarios and compared the obtained results with the ones obtained using the ONE simulator (using the epidemic simulation), achieving the same results.

6.1. Detailed experiments using ROMA and STATION scenarios

Firstly, we start with the ROMA traces, considering only the hours from 7 a.m. to 12 a.m., which can be considered the rush hours, and using a setup time of 5 s and a bandwidth of 10 Mbps. The results are shown in Figure 12. As in the analytical evaluation, we obtained the diffusion ratio and improvement factor (Figure 12a and 12b), confirming the improvement in terms of diffusion

⁵The code of this simulator will be freely available for the research community on the GitHub site of our research group when the paper will be published.

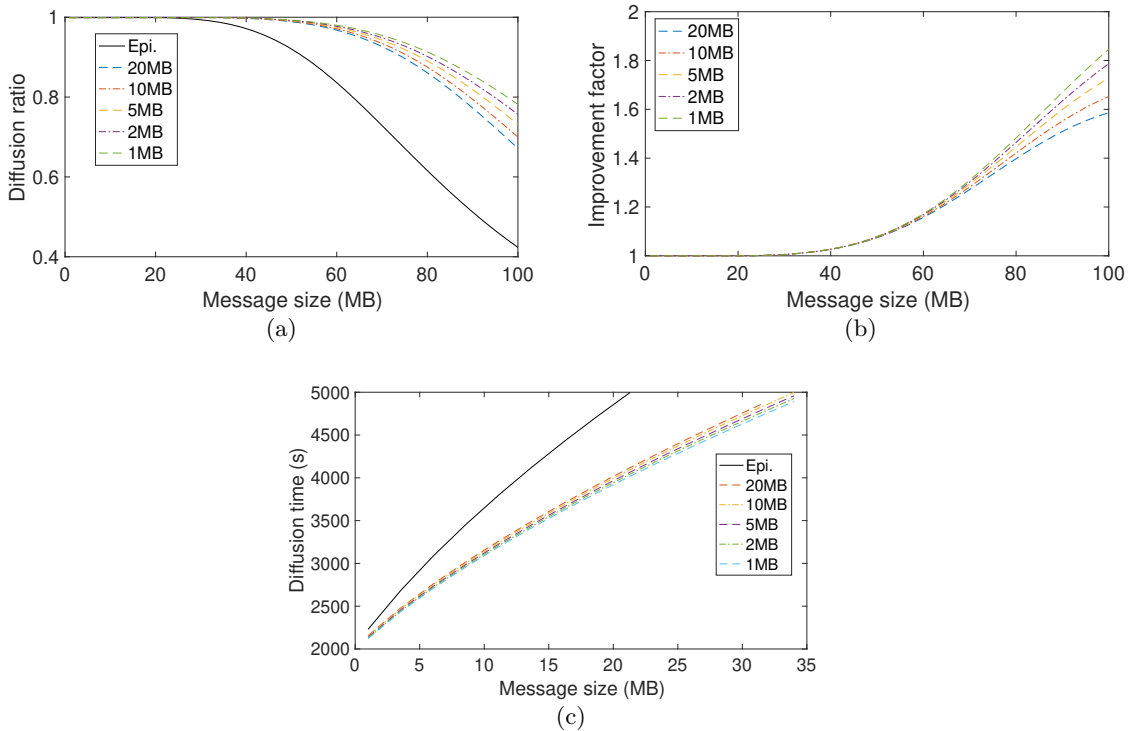


Figure 11: Diffusion ratio and diffusion time results for the ROMA trace depending on message size for a setup time of 5 s. (a) Diffusion ratio. (b) Improvement factor of the diffusion ratio with respect to the epidemic protocol. (c) Diffusion time (only when the diffusion ratio is 1).

effectiveness of the *Xpread* protocol, and that the best results are obtained for a 1 MB chunk size. Figure 12c shows the average delivery time, that is obtained as the mean of the time of all nodes that have received the message (note that it is not the same as the previous diffusion time), and where we can see a slight reduction in the delivery time for smaller chunk sizes, particularly for message sizes greater than 20 MB.

Now, we focus our evaluation on the overhead. Figure 12d shows the number of chunks successfully transmitted, evidencing, as expected, that for smaller chunk sizes, its number increases substantially for larger messages. Nevertheless, in our measures, we noticed that the increment on the number of unsuccessfully transmitted messages is not as substantial when compared to the Epidemic, which has a lot of started and not finished transmissions (that is, failed transmissions). We also obtained the whole number of bytes transmitted (T_B) in both successfully and unsuccessfully transmitted messages, which is a good measure of the generated overhead. Figure 12e shows that, for message sizes greater than 20 MB, the *Xpread* protocol reduces the number of bytes transmitted, particularly for small chunk sizes. The main cause is that the Epidemic protocol diffusion ratio falls since it attempts to transmit large messages and wastefully consumes transmission resources. Finally, normalising the bytes transmitted with the number of successfully transmitted messages (*i.e.*, $N \cdot R_d$) and message size (*i.e.*, m), we obtain the overhead as $O = T_B/m/(N \cdot R_d)$, which is a relative measure of the overhead per successfully message transmitted, being that one represents no overhead. The results are in Figure 12f showing that the incurred overhead for the *Xpread* protocol is smaller than for Epidemic, being minimal for small chunks sizes.

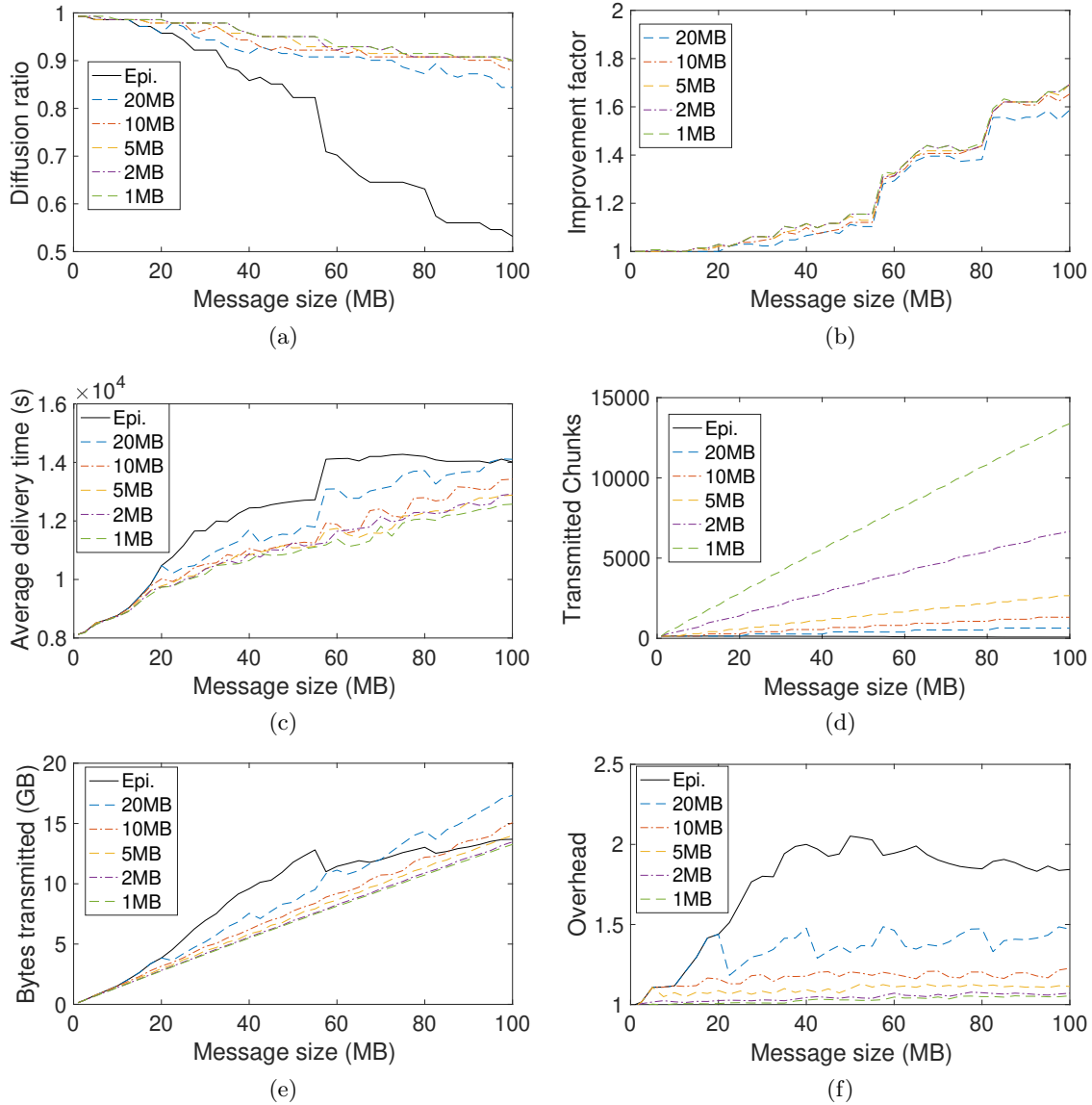


Figure 12: Results of simulating the diffusion of a message using the ROMA trace for a setup time 5 s when using the epidemic protocol (the “Epi.” line) and the *Xpread* protocol, with different chunk sizes. (a) Diffusion ratio. (b) Improvement factor of the diffusion ratio against the epidemic protocol. (c) Average delivery time. (d) Number of successfully transmitted messages. (e) Whole number of bytes transmitted (in gigabytes). (f) Overhead per message successfully transmitted.

We repeated the previous experiments using the STATION scenario. Note that this scenario
 530 consists of pedestrians using mobile devices for exchanging messages with Bluetooth technology,
 considering a setup time of 5 s and a bandwidth of 2 Mbps. Furthermore, this scenario has a
 high renovation of nodes, that is, people are entering and leaving the station continuously. This
 high node renewal has a huge impact on the broadcast of messages as nodes remain in the station
 only for a short time, reducing their opportunity to exchange the message. Besides, the imposed
 535 bandwidth increases the transmission time so large messages are very unlikely to be transmitted.

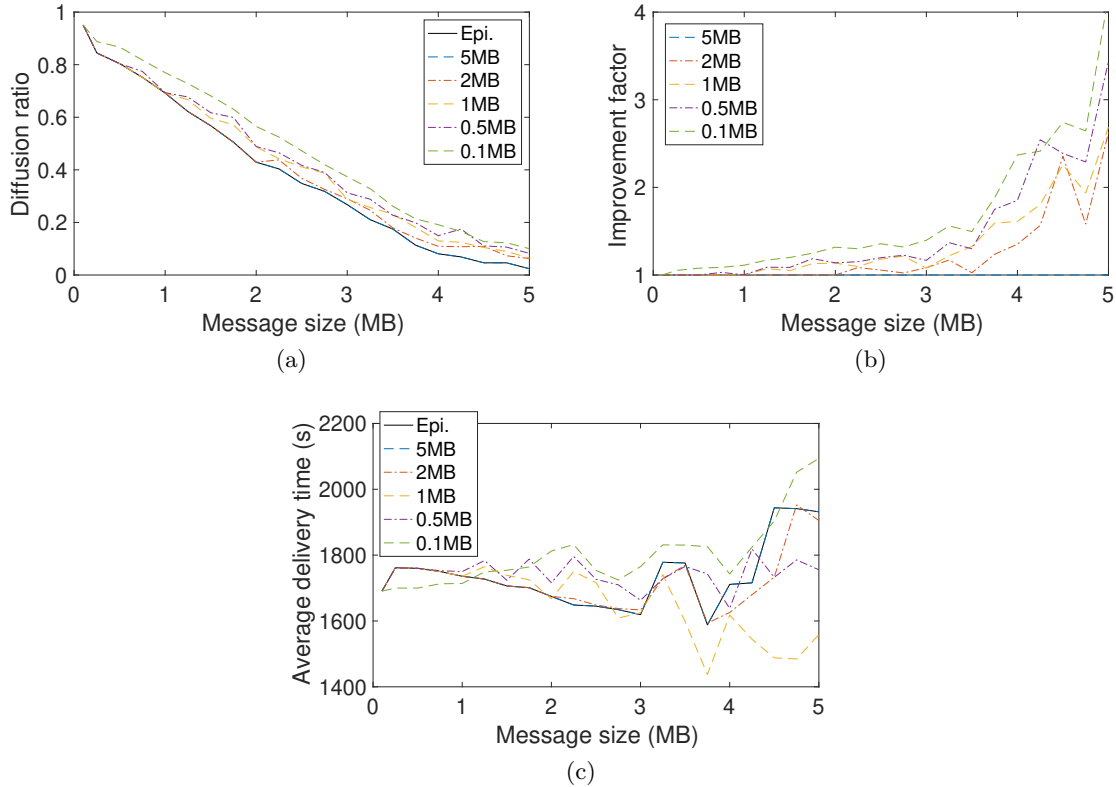


Figure 13: Results of simulating the diffusion of a message using the STATION trace for a setup time 5 s when using the epidemic protocol (the “Epi.” line) and the *Xpread* protocol, with different chunk sizes. (a) Diffusion ratio. (b) Improvement factor of the diffusion ratio against the epidemic protocol. (c) Average delivery time.

Thus, the evaluated messages sizes range from 0.1 to 5 MB.

The results are shown in Figure 13, and reflect the main temporal characteristics of this pedestrian scenario and its high node renewal. Firstly, Figure 13a shows the reduced diffusion ratio for message sizes greater than 2 MB, and, in general, a high improvement factor is obtained for these messages sizes as shown in Figure 13b. Secondly, Figure 13c shows that the delivery time is shorter than in the ROMA scenario, due to the limited time that pedestrians stay in the station. Regarding the bytes transmitted and overhead, the results of the experiments were very similar to the ones in Figure 12, that is, the number of bytes transmitted increases with the size of the message being the *Xpread* protocol the one with the lower overhead. Therefore, these plots are not included in this paper.

6.2. Extended experiments

We repeated the experiments described in Subsection 6.1 using the SFO and NCCU traces. The obtained results are similar to the ones presented above obtaining significant improvement diffusions for the *Xpread* protocol when compared to Epidemic. Remarkably, for the SFO scenario, the overhead is higher than in the ROMA scenario, due to the higher node density detected.

In Table 2, we summarise the main results depending on the setup time. For all the setup time values we obtained average values of the diffusion ratio (*i.e.*, \bar{R}_d), and the whole number of bytes

t_s	R_d		CS		O		R_d		CS		O	
	\mathcal{E}	\mathcal{X}			\mathcal{E}	\mathcal{X}	\mathcal{E}	\mathcal{X}			\mathcal{E}	\mathcal{X}
	ROMA						SFO					
0	0.88	0.98	7.2	1.92	1.14		0.78	0.98	4.4	2.91	1.48	
2	0.80	0.96	4.9	1.81	1.27		0.74	0.97	2.1	2.60	1.66	
4	0.77	0.95	2.2	1.76	1.33		0.71	0.97	0.2	2.45	1.75	
6	0.78	0.94	0.4	1.70	1.30		0.70	0.96	0.4	2.32	1.83	
8	0.74	0.94	0.5	1.65	1.36		0.67	0.95	0.5	2.19	1.95	
10	0.74	0.92	0.6	1.58	1.35		0.66	0.94	0.6	2.08	1.99	
	STATION						NCCU					
0	0.82	0.97	0.52	5.99	2.86		0.83	0.97	0.74	2.65	1.33	
2	0.57	0.77	0.04	4.86	5.11		0.82	0.96	0.26	2.53	1.34	
4	0.45	0.60	0.05	2.99	2.71		0.80	0.92	0.05	2.47	1.35	
6	0.35	0.49	0.07	3.19	4.04		0.77	0.88	0.07	2.40	1.34	
8	0.26	0.38	0.10	5.87	6.02		0.75	0.86	0.10	2.35	1.34	
10	0.19	0.34	0.12	7.88	8.91		0.71	0.82	0.12	2.33	1.40	

Table 2: Main performance metrics depending on the setup time (t_s) for **all** traces. Column names: \mathcal{E} corresponds to the Epidemic protocol, and \mathcal{X} to the Xpread protocol. R_d stands for the average diffusion ratio, “CS” for the optimal chunk size (in MB), and \bar{O} for the overhead per **message successfully transmitted**.

transmitted (*i.e.*, \bar{T}_B), for message sizes ranging from 1 to 100 MB, with a 5 MB step (except for the STATION scenario that ranges from 0.1 to 5 MB). We include the values for the Epidemic
555 protocol and for the optimal configuration of the *Xpread* protocol (that is, using the optimal chunk size obtained using Eq. (7), which is shown in column “CS”). Firstly, the obtained chunk sizes confirm the optimisation proposed in the previous section. Secondly, regarding the impact of the diffusion ratio, we can see that the diffusion ratio is reduced, and, consequently, the overhead is increased when the setup time is higher. Particularly, in the STATION scenario, the impact of
560 the setup time is very significant, due to its shorter contact times. Finally, the obtained results globally confirm the efficiency and low overhead of our *Xpread* protocol.

We repeated the previous experiments using other **scenarios** (*i.e.*, **days and time ranges of the ROMA and SFO traces different from the ones described in section 3**) obtaining similar results. Additionally, we evaluated the increase on the number of fixed nodes, which supposes a slight
565 increase of the diffusion ratio, and a reduction on the delivery time; however as for the *Xpread* protocol, its improvement factor is similar. Regarding bandwidth, the effect is a reduction in the size of the transmitted messages, being the curve similar to the presented ones. For example, if the bandwidth is reduced from 10 Mbps to 2 Mbps, the results of Figure 12 will be the same, but for a message size ranging from 0 to 20 MB.

570 6.3. Impact of the Message Forwarding policy on the diffusion

In Opportunistic Networks, nodes store messages in local buffers waiting for an opportunity to be transmitted when a contact occurs. The decision of which messages are transmitted and when is taken by the node’s buffer management message-forwarding policy. Several policies have been proposed: *Random*, FIFO, *Oldest* (send lowest TTL message first) and *Smallest* (send small messages
575 first). These policies can affect the efficiency of message diffusion, as shown in [50]. Therefore, as the message is partitioned into chunks, the local forwarding policy can have a significant impact on the diffusion as we show in this subsection.

The previous experiments were performed using a *Time-Based* order scheduling, with the aim that each node had a different and predefined order depending on time. This way, the implemented policy can assure fairness on the transmission of all the chunks. This order is based on the time t when a node starts to transmit its chunks. Specifically, if a message is divided into w chunks, at time t , a node will start the transmission with the $\text{mod}(t, w)$ chunk, being mod the modulus or remainder. Note that this only defines where to start, so if the node does not have this chunk, or if it is not necessary to transmit it, it will move to the next one, and so on. If the node reaches chunk w , it will move to the first one (*i.e.* it is a circular buffer).

Thus, in order to evaluate other message forwarding policies, we implemented in our simulator the following policies: (i) *Ordered*, where the chunks are numbered and always transmitted in this order, (ii) *Random*, where the next chunk to transmit is selected randomly among the chunks that are stored in the buffer, and (iii) FIFO, that considers the chunks are dispatched in the arrival order (First-In, First-Out). Other policies, such as *Oldest* and *Smallest*, do not make sense in this evaluation since all chunks have the same size and TTL. Using the simulator, we evaluated the diffusion ratio depending on the number of partitions for the four policies (including also the *Time-based* one) in the ROMA and STATION scenario. The network configuration was the same as in the previous subsection. Furthermore, the randomness nature of the *Random* policy has the effect of generating slightly different results for each performed simulation. Consequently, for this policy, we repeated each simulation 20 times in order to obtain the mean and 95% confidence intervals.⁶

The results are shown in Figure 14. In both scenarios, we can see that the forwarding policy has an impact on the diffusion ratio. The difference between the different policies is more significant in the STATION scenario than in the ROMA scenario, mainly due to the short duration of the contacts, which reduces the diffusion. In general, as expected, the worst policy is the *Ordered* one, since some chunks (particularly the last ones) are less likely to be transmitted, reducing their diffusion, and so some nodes cannot complete the whole message. On the other hand, the best policies are the ones that assure fairness. Beside the *Time-Based* policy, the *Random* policy obtains similar average results. This result is important as the Random policy is usually implemented in most Opportunistic Networks, so the *Xpread* protocol would not require modifications to its forwarding policy in order to obtain good diffusion results.

6.4. Evaluation of the diffusion of multiple simultaneous messages

The analytical model and the previously performed simulations have evaluated the performance of a single message diffusion. The results have shown great improvement in message diffusion when using our *Xpread* protocol. Nevertheless, in real applications, it is common to broadcast several messages simultaneously. This implies that the network load will be increased and, in general, the diffusion ratio will be reduced, and the delivery time will increase.

The goal of the following experiment is to evaluate this issue: the diffusion of several simultaneous messages. Specifically, we repeated the experiments of Subsection 6.1 for the ROMA scenario sending four different messages, all with the same size, and using two fixed nodes (*i.e.*, $F = 2$). Besides the Epidemic protocol, in these new experiments using the *Xpread* protocol we used only one chunk size, which was set to its optimal size of 1 MB. The results are shown Figure 15 (curves

⁶Note that the other policies are completely deterministic, so being the node's movement and contacts defined by the scenarios, the results of the simulation are always the same.

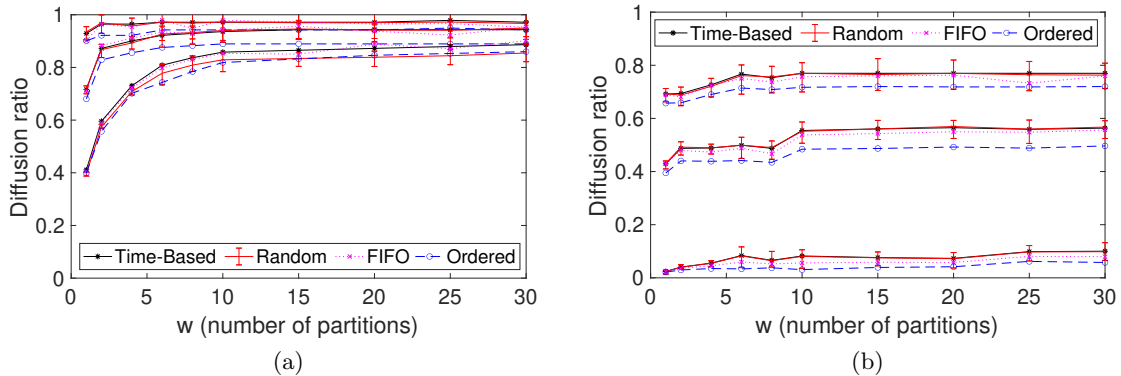


Figure 14: Buffer management impact on the diffusion ratio, depending on the number of partitions using messages with different sizes. (a) ROMA scenario: from top to bottom the message sizes are 50 MB, 100 MB, 150 MB; (b) STATION scenario: from top to bottom the message sizes are 1 MB, 2 MB and 5 MB.

with $F = 2$), where we focus our evaluation on the average diffusion ratio and delivery time. We can see that, when compared with the results of sending one message of Figure 12, the diffusion ratio is reduced notably, and the delivery time is increased slightly. This is the effect of the increased network load. Anyhow, *Xpread* provides a significant diffusion improvement over the *Epidemic*.

A closer study of the previous simulations showed that a critical bottleneck for the diffusion of the messages was the number of fixed nodes. Although the used message forwarding policy assures fairness on the selection of the chunk, the problem is that, as the source of these messages are these fixed nodes, they require more time to spread all the chunks among the mobile nodes when they get in contact with the fixed nodes. This limits the spreading of the chunks by the mobile nodes on the network, which could be faster if the fixed nodes could initially transmit more chunks. A solution is to introduce more fixed nodes as a way to accelerate the transmission of the chunk. Consequently, we repeated the previous experiments using four fixed nodes (*i.e.*, $F = 4$). The results are shown in Figure 15 (curves with $F = 4$), where we can see a significant improvement on the average diffusion ratio, and also a remarkable reduction on the delivery time, confirming the speed up of the message diffusion. Further experiments with more fixed nodes showed a slight speed up, but not as significant as when the fixed nodes were increased from 2 to 4. Nevertheless, in the real world, this comes at a cost: we need to deploy these fixed nodes, which cannot always be feasible. And paradoxically, this is one of the main reasons for using Opportunistic Networks, to avoid the use of fixed infrastructure. Finally, we also did similar simulations in the STATION scenario, obtaining similar results.

Now, we evaluate the effect of reducing the message TTL (Time-To-Live) on the diffusion. Note that, in real applications, messages can have a temporal life (for example, up-to-date traffic or weather information), so it is necessary to evaluate this issue. In the previous experiments, we set the TTL to the simulation running time; so, in this new experiment, we reduce it to 1 hour. The results are shown in Figure 16, where we can see that diffusion is reduced when compared to previous results with longer TTL (Figure 12), particularly for large messages. As expected, the main reason is that the *Xpread* protocol does not have enough time to transmit all the chunks. On the other hand, *Epidemic* diffusion is very poor for messages greater than 40 MB, being even sometimes unable to reach more than two or three nodes. In these cases, *Xpread* can provide a

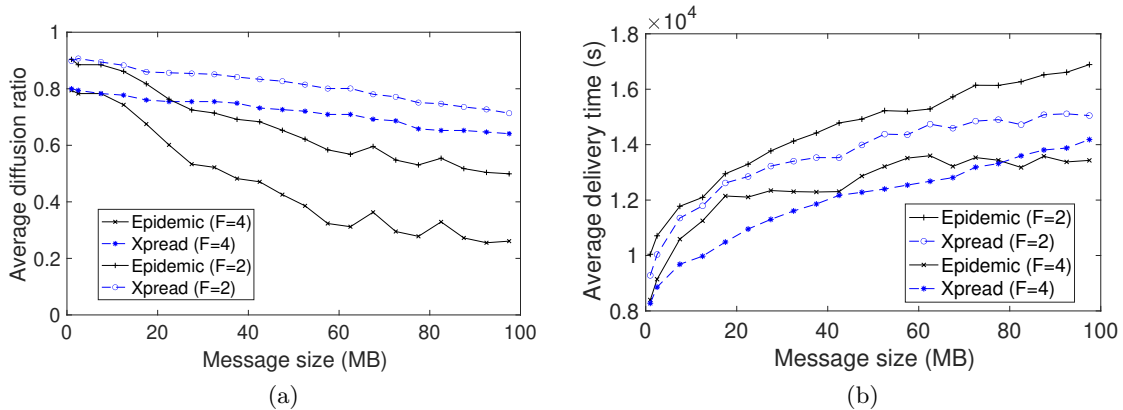


Figure 15: Results of simulating the diffusion of 4 simultaneous messages using the ROMA trace for a setup time 5 s, and with two fixed nodes and four fixed nodes. (a) Average diffusion ratio. (b) Average delivery time.

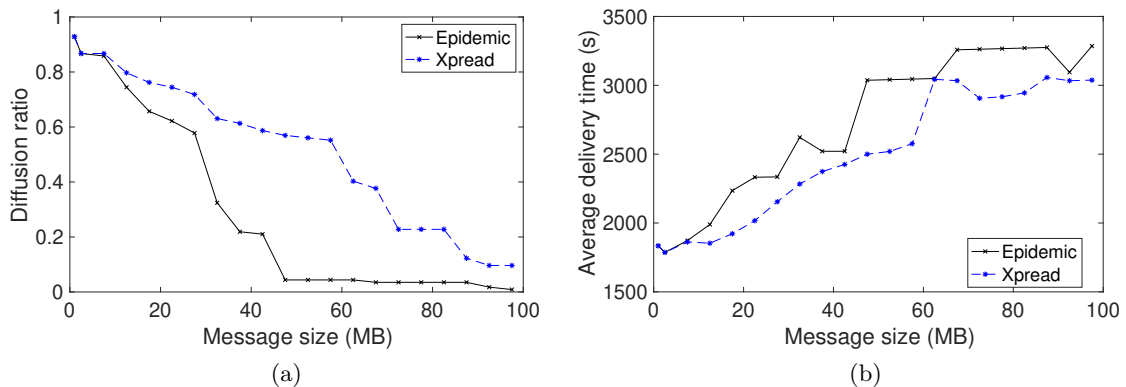


Figure 16: Results of simulating the diffusion of one message with a TTL of 1 hour. (a) Average diffusion ratio. (b) Average delivery time.

diffusion rate of up to 0.6, making it an ideal option for transmitting messages with shorter live times.

650 Summing up, the experimental results have confirmed that our *Xpread* protocol substantially improves the diffusion of messages with a slight reduction of the delivery time, and reducing the overhead per message successfully transmitted.

7. Conclusions

655 In this paper we focused on the impact of contact duration on message broadcasting in Opportunistic Networks, showing that splitting a message into smaller parts can improve its diffusion. Moreover, we have proposed an extension to the epidemic protocol called *Xpread*, showing that messages diffusion can be improved, particularly for large messages.

660 Since the efficiency of our protocol mainly depends on how the original message is partitioned, we proposed an analytical model to obtain the optimal partitioning scheme. Using this model we determined that the best partitioning scheme is the fixed-size, also obtaining a simple expression

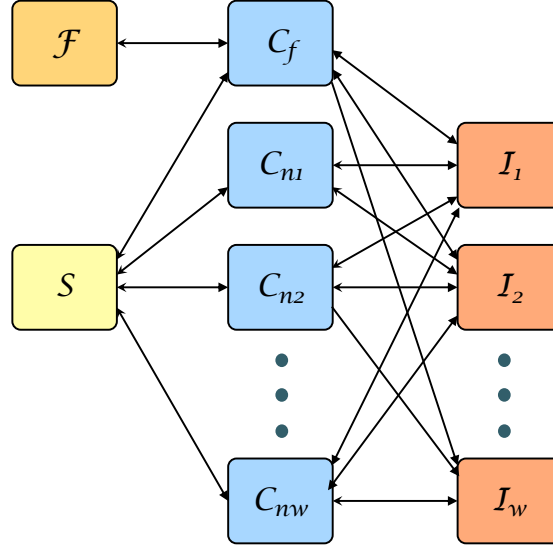


Figure A.1: Graphical representation of the transitions between classes for the division into w -parts model.

to determine the optimal chunk size. With this partitioning scheme, we can obtain diffusion improvements factors up to two, mainly when the epidemic protocol is not so efficient.

Finally, through simulation, we showed that X_{pread} increases the diffusion of messages with a slight reduction of the delivery time and a significant reduction of the overhead. Moreover, in our evaluation, we evidenced the importance of considering the communication setup time when analysing the performance of these types of protocols. Regarding the message forwarding policy, we have shown that using a simple *Random* policy can assure the required fairness for the transmission of the chunks.

Acknowledgments

This work was partially supported by the *Ministerio de Ciencia, Innovación y Universidades*, Spain, under Grant RTI2018-096384-B-I00; and the Secretaría Nacional de Educación Superior, Ciencia, Tecnología e Innovación del Ecuador (SENESCYT), Ecuador.

Appendix: Model extension to a variable number of chunks

In this appendix, we show how to extend the model considering the partition of a message into w chunks extending the equations defined in Eq. 3. The idea is first to describe the mathematical model and then to depict how it can be solved through numerical methods. This numerical solution is the one used in Section 5.

As in the model described in 5, we consider an area populated with N mobile nodes and F fixed nodes with two defined contact rates *i.e.*, λ and Λ . From this basic model (that is, when $w = 2$), we need to extend the number of infected and communicating classes considering w subclasses: \mathcal{I}_i ($i = 1 \dots w$) correspond to the classes with nodes that have i chunks, and \mathcal{C}_{ni} ($i = 1 \dots w$) are the classes that receive the nodes that are communicating from classes \mathcal{I}_i . The rest of the classes (*i.e.*, \mathcal{S} , \mathcal{F} and \mathcal{C}_f) do not change.

	+w chunks	+(w-1) chunks	...	+2 chunks	+1 chunk	NO chunk
$\mathcal{F}' =$	$-\Lambda\mathcal{F}\mathcal{S}$	$+\overline{\mathbf{P}\mathbf{f}}_w p_{fw-1} \Lambda\mathcal{S}\mathcal{F}(t_{cw-1})$...	$+\overline{\mathbf{P}\mathbf{f}}_3 p_{f2} \Lambda\mathcal{S}\mathcal{F}(t_{c2})$	$+\overline{\mathbf{P}\mathbf{f}}_2 p_{f1} \Lambda\mathcal{S}\mathcal{F}(t_{c1})$	$+\overline{\mathbf{P}\mathbf{f}}_1^w \Lambda\mathcal{S}\mathcal{F}(t_f)$
	$-\Lambda\mathcal{F}\mathcal{I}_1$	$+p_{fw-1} \Lambda\mathcal{I}_1\mathcal{F}(t_{cw-1})$...	$+\overline{\mathbf{P}\mathbf{f}}_3^{w-1} p_{f2} \Lambda\mathcal{I}_1\mathcal{F}(t_{c2})$	$+\overline{\mathbf{P}\mathbf{f}}_2^{w-1} p_{f1} \Lambda\mathcal{I}_1\mathcal{F}(t_{c1})$	$+\overline{\mathbf{P}\mathbf{f}}_1^{w-1} \Lambda\mathcal{I}_1\mathcal{F}(t_f)$
	$-\Lambda\mathcal{F}\mathcal{I}_2$	$+\overline{\mathbf{P}\mathbf{f}}_3^{w-2} p_{f2} \Lambda\mathcal{I}_2\mathcal{F}(t_{c2})$	$+\overline{\mathbf{P}\mathbf{f}}_2^{w-2} p_{f1} \Lambda\mathcal{I}_2\mathcal{F}(t_{c1})$	$+\overline{\mathbf{P}\mathbf{f}}_1^{w-2} \Lambda\mathcal{I}_2\mathcal{F}(t_f)$

	$-\Lambda\mathcal{F}\mathcal{I}_{w-1}$	$+p_{f1} \Lambda\mathcal{I}_{w-1}\mathcal{F}(t_{c1})$	$+\overline{\mathbf{P}\mathbf{f}}_1^1 \Lambda\mathcal{I}_{w-1}\mathcal{F}(t_f)$
$\mathcal{S}' =$	$-\Lambda\mathcal{S}\mathcal{F}$	$+\overline{\mathbf{P}\mathbf{f}}_1^w \Lambda\mathcal{S}\mathcal{F}(t_f)$
	$-\lambda\mathcal{S}\mathcal{I}_1$	$+\overline{\mathbf{P}\mathbf{n}}_1^1 \lambda\mathcal{S}\mathcal{I}_1(t_n)$
	$-\lambda\mathcal{S}\mathcal{I}_2$	$+\overline{\mathbf{P}\mathbf{n}}_2^1 \lambda\mathcal{S}\mathcal{I}_2(t_n)$

	$-\lambda\mathcal{S}\mathcal{I}_w$	$+\overline{\mathbf{P}\mathbf{n}}_1^w \lambda\mathcal{S}\mathcal{I}_w(t_n)$
$\mathcal{I}'_1 =$	$+\overline{\mathbf{P}\mathbf{f}}_2^w p_{f1} \Lambda\mathcal{S}\mathcal{F}(t_{c1})$.
	$-\Lambda\mathcal{I}_1\mathcal{F}$	$+\overline{\mathbf{P}\mathbf{n}}_2^2 p_{n1} \lambda\mathcal{S}\mathcal{I}_2(t_{c1})$	$+\overline{\mathbf{P}\mathbf{f}}_1^{w-1} \Lambda\mathcal{I}_1\mathcal{F}(t_f)$
	$-\lambda\mathcal{I}_1\mathcal{S}$	$+2p_{n1} \lambda\mathcal{I}_1\mathcal{S}(t_{c1})$	$+\overline{\mathbf{P}\mathbf{n}}_1^1 \lambda\mathcal{I}_1\mathcal{S}(t_n)$
	$-\lambda\mathcal{I}_1\mathcal{I}_2$	$+\overline{\mathbf{P}\mathbf{n}}_1^1 \lambda\mathcal{I}_1\mathcal{I}_2(t_n)$

	$-\lambda\mathcal{I}_1\mathcal{I}_w$	$+\overline{\mathbf{P}\mathbf{n}}_1^{w-1} \lambda\mathcal{I}_1\mathcal{I}_w(t_n)$
$\mathcal{I}'_2 =$	$+\overline{\mathbf{P}\mathbf{f}}_3^w p_{f2} \Lambda\mathcal{S}\mathcal{F}(t_{c2})$.	.
	$-\Lambda\mathcal{I}_2\mathcal{F}$	$+\overline{\mathbf{P}\mathbf{f}}_2^{w-1} p_{f1} \Lambda\mathcal{I}_1\mathcal{F}(t_{c1})$	$+\overline{\mathbf{P}\mathbf{f}}_1^{w-1} \Lambda\mathcal{I}_1\mathcal{F}(t_f)$
	$-\lambda\mathcal{I}_2\mathcal{S}$	$+2p_{n2} \lambda\mathcal{I}_2\mathcal{S}(t_{c2})$	$+\overline{\mathbf{P}\mathbf{n}}_2^2 p_{n1} \lambda\mathcal{I}_2\mathcal{S}(t_{c1})$	$+\overline{\mathbf{P}\mathbf{n}}_1^1 \lambda\mathcal{I}_2\mathcal{S}(t_n)$
	$-\lambda\mathcal{I}_2\mathcal{I}_1$	$+2p_{n1} \lambda\mathcal{I}_2\mathcal{I}_1(t_{c1})$	$+\overline{\mathbf{P}\mathbf{n}}_1^1 \lambda\mathcal{I}_2\mathcal{I}_1(t_n)$

	$-\lambda\mathcal{I}_2\mathcal{I}_w$	$+\overline{\mathbf{P}\mathbf{n}}_1^{w-2} \lambda\mathcal{I}_2\mathcal{I}_w(t_n)$

$\mathcal{I}'_w =$	$+p_{fw} \Lambda\mathcal{S}\mathcal{F}(t_{cw})$	$+p_{fw-1} \Lambda\mathcal{I}_1\mathcal{F}(t_{cw-1})$
	$-\lambda\mathcal{I}_w\mathcal{S}$	$+2p_{nw} \lambda\mathcal{I}_w\mathcal{S}(t_{cw})$	$+\overline{\mathbf{P}\mathbf{n}}_w p_{nw-1} \lambda\mathcal{I}_w\mathcal{S}(t_{cw-1})$	$+\overline{\mathbf{P}\mathbf{n}}_3^w p_{n2} \lambda\mathcal{I}_w\mathcal{S}(t_{c2})$	$+\overline{\mathbf{P}\mathbf{n}}_2^w p_{n1} \lambda\mathcal{I}_w\mathcal{S}(t_{c1})$	$+\overline{\mathbf{P}\mathbf{n}}_1^w \lambda\mathcal{I}_w\mathcal{S}(t_n)$
	$-\lambda\mathcal{I}_w\mathcal{I}_1$	$+2p_{nw-1} \lambda\mathcal{I}_w\mathcal{I}_1(t_{cw-1})$...	$+\overline{\mathbf{P}\mathbf{n}}_3^{w-1} \lambda\mathcal{I}_w\mathcal{I}_1(t_{c2})$	$+\overline{\mathbf{P}\mathbf{n}}_2^{w-1} p_{n1} \lambda\mathcal{I}_w\mathcal{I}_1(t_{c1})$	$+\overline{\mathbf{P}\mathbf{n}}_1^{w-1} \lambda\mathcal{I}_w\mathcal{I}_1(t_n)$

	$-\lambda\mathcal{I}_w\mathcal{I}_{w-1}$	$+2p_{n1} \lambda\mathcal{I}_w\mathcal{I}_{w-1}(t_{c1})$	$+\overline{\mathbf{P}\mathbf{n}}_1^1 \lambda\mathcal{I}_w\mathcal{I}_{w-1}(t_n)$

(A.1)

Accordingly, the communication times also [range](#) from t_{c1} for one chunk transmission to t_{cw} for transmitting w chunks (that is, the whole message). The waiting times for unsuccessful message transmissions, that is, t_n and t_f , are the same [as](#) in the two chunks model. Regarding the probability of transmission, for the contacts among mobile nodes we have p_{ni} , $i = 1 \dots w$ that stand for the probability of successful transmission of i chunks, which are obtained from the cCDFs of the contact durations: $p_{ni} = \bar{F}_n(t_{ci})$; and for the contacts between fixed and mobile nodes we have similar probabilities $p_{fi} = \bar{F}_f(t_{ci})$ $i = 1 \dots w$. On the other hand, $(1 - p_{n1})$ will be the probability of unsuccessful transmission of one chunk between nodes, and thus, $\prod_{j=1}^n (1 - p_{nj})$ will be the unsuccessful transmission of n chunks. The same expression can be derived for the unsuccessful transmission of n chunks between a fixed and mobile nodes: $\prod_{j=1}^n (1 - p_{fj})$. For these expressions,

	+w chunks	+(w-1) chunks	...	+2 chunks	+1 chunk	NO chunk	
$C'_f =$	$+2\lambda S\mathcal{F}$	$-2p_{fw}\lambda S\mathcal{F}(t_{cw})$	$-2\overline{\mathbf{P}\mathbf{f}}_w^w p_{fw-1}\lambda S\mathcal{F}(t_{cw-1})$	\dots	$-2\overline{\mathbf{P}\mathbf{f}}_2^w p_{f2}\lambda S\mathcal{F}(t_{c2})$	$-2\overline{\mathbf{P}\mathbf{f}}_1^w \lambda S\mathcal{F}(t_{c1})$	$-2\overline{\mathbf{P}\mathbf{f}}_1^w \lambda S\mathcal{F}(t_f)$
	$+2\lambda \mathcal{I}_1 \mathcal{F}$		$-2p_{fw-1}\lambda \mathcal{I}_1 \mathcal{F}(t_{cw-1})$	\dots	$-2\overline{\mathbf{P}\mathbf{f}}_3^{w-1} p_{f2}\lambda \mathcal{I}_1 \mathcal{F}(t_{c2})$	$-2\overline{\mathbf{P}\mathbf{f}}_2^{w-1} p_{f1}\lambda \mathcal{I}_1 \mathcal{F}(t_{c1})$	$-2\overline{\mathbf{P}\mathbf{f}}_1^{w-1} \lambda \mathcal{I}_1 \mathcal{F}(t_f)$
	$+2\lambda \mathcal{I}_2 \mathcal{F}$			\dots	$-2\overline{\mathbf{P}\mathbf{f}}_3^{w-2} p_{f2}\lambda \mathcal{I}_2 \mathcal{F}(t_{c2})$	$-2\overline{\mathbf{P}\mathbf{f}}_2^{w-2} p_{f1}\lambda \mathcal{I}_2 \mathcal{F}(t_{c1})$	$-2\overline{\mathbf{P}\mathbf{f}}_1^{w-2} \lambda \mathcal{I}_2 \mathcal{F}(t_f)$
	\vdots			\dots	\vdots		
	$+2\lambda \mathcal{I}_{w-1} \mathcal{F}$			\dots	$-2p_{f1}\lambda \mathcal{I}_{w-1} \mathcal{F}(t_{c1})$	$-2\overline{\mathbf{P}\mathbf{f}}_1^1 \lambda \mathcal{I}_{w-1} \mathcal{F}(t_f)$	
$C'_{n1} =$	$+2\lambda S\mathcal{I}_1$			\dots	$-2p_{n1}\lambda S\mathcal{I}_1(t_{c1})$	$-2\overline{\mathbf{P}\mathbf{n}}_1^1 \lambda S\mathcal{I}_1(t_n)$	
$C'_{n2} =$	$+2\lambda S\mathcal{I}_2$			\dots	$-2p_{n2}\lambda S(t_{c2}\mathcal{I}_2)$	$-2\overline{\mathbf{P}\mathbf{n}}_2^2 p_{n1}\lambda S\mathcal{I}_2(t_{c1})$	$-2\overline{\mathbf{P}\mathbf{n}}_1^2 \lambda S\mathcal{I}_2(t_n)$
	$+2\lambda \mathcal{I}_1 \mathcal{I}_2$			\dots		$+2p_{n1}\lambda \mathcal{I}_1 \mathcal{I}_2(t_{c1})$	$+2\overline{\mathbf{P}\mathbf{n}}_1^1 \lambda \mathcal{I}_1 \mathcal{I}_2(t_n)$
	\vdots			\dots	\vdots		
	\vdots			\dots	\vdots		
	\vdots			\dots	\vdots		
$C'_{nw} =$	$+2\lambda S\mathcal{I}_w$	$+2p_{fw}\lambda S\mathcal{I}_w(t_{cw})$	$-2p_{fw-1}\lambda S\mathcal{I}_w(t_{cw-1})$	\dots	$-2p_{nw}\lambda S\mathcal{I}_w(t_{c2})$	$-2\overline{\mathbf{P}\mathbf{n}}_2^2 p_{n1}\lambda S\mathcal{I}_w(t_{c1})$	$-2\overline{\mathbf{P}\mathbf{n}}_1^2 \lambda S\mathcal{I}_w(t_n)$
	$+2\lambda \mathcal{I}_1 \mathcal{I}_w$		$-2p_{nw-1}\lambda \mathcal{I}_1 \mathcal{I}_w(t_{cw-1})$	\dots	$-2\overline{\mathbf{P}\mathbf{n}}_3^{w-1} \lambda \mathcal{I}_1 \mathcal{I}_w(t_{c2})$	$-2\overline{\mathbf{P}\mathbf{n}}_2^{w-1} p_{n1}\lambda \mathcal{I}_1 \mathcal{I}_w(t_{c1})$	$+2\overline{\mathbf{P}\mathbf{n}}_1^{w-1} \lambda \mathcal{I}_1 \mathcal{I}_w(t_n)$
	\vdots			\dots	\vdots		
	$+2\lambda \mathcal{I}_{w-1} \mathcal{I}_w$			\dots	$-2p_{n1}\lambda \mathcal{I}_{w-1} \mathcal{I}_w - (t_{c1})$	$-\overline{\mathbf{P}\mathbf{n}}_1^1 \lambda \mathcal{I}_{w-1} \mathcal{I}_w(t_n)$	

(A.2)

we use the following notation to simplify the mathematical description of the model:

$$\overline{\mathbf{P}\mathbf{f}}_a^b = \prod_{j=a}^b (1 - p_{fj}), \quad \overline{\mathbf{P}\mathbf{n}}_a^b = \prod_{j=a}^b (1 - p_{nj}). \quad (\text{A.3})$$

The classes and transitions are shown in Figure A.1, which is an extension of the model depicted in Figure 7. From these transitions, we define a set of Delay Differential Equations (DDEs), as shown in Eq. (A.1) and Eq. (A.2). These equations are arranged into columns to represent the rate of transition between classes when i chunks are successfully transmitted, ranging from w (whole message transmitted) to 0 (no chunk is transmitted). We can see that the transitions follow a fixed pattern, which was used to develop the numerical algorithm as explained below. For example, for the \mathcal{F}' equation, we can see that, when no message is transmitted (NO chunk column), the flow of nodes returning is determined by the probability of not transmitting any message: $\overline{\mathbf{P}\mathbf{f}}_{w-j}^2$ being j the row. For the \mathcal{I}'_w equation we can also see how the number of received chunks determines the flow of nodes arriving at the \mathcal{I}_w class, which is conditioned by the probability of transmitting the chunks.

This variable set of equations is solved numerically using the Euler's method for a given time T and using a h step. The method is described in Algorithm 1. It is based on iteratively updating the values of the different classes. The number of elements in each class along time is stored in vectors of size $\lceil T/h \rceil$. Note that the \mathcal{I}_i and \mathcal{C}_{ni} subclasses require matrix representations where the first index represents the subclass. Each class is updated using the Euler's method *i.e.*,

$$y_{i+1} = y_i + hf(t_i, y_i). \quad (\text{A.4})$$

In our algorithm, we have defined a set of functions ($\mathbf{f}\mathbf{X}$) to calculate the value of the previous function f , which have access to all the classes and input variables. The way the continuous equations are converted to a set of discrete equations was shown in Section 5 when the \mathcal{F}' equation

was converted into Eq. (4). The expansion of each equation is performed algorithmically in each \mathbf{fX} function, depending on the value of w . For example, the expansion of the first equation in Eq. (A.1):

$$\begin{aligned} \mathcal{F}' = & -\Lambda\mathcal{F}\mathcal{S} + p_{fw}\Lambda\mathcal{S}\mathcal{F}(t_{cw}) + \sum_{j=0}^{w-1} \overline{\mathbf{P}\mathbf{f}}_{j+1}^w p_{fj}\Lambda\mathcal{S}\mathcal{F}(t_{cj}) \\ & - \sum_{k=1}^{w-1} \Lambda\mathcal{F}\mathcal{I}_k + \sum_{j=1}^{w-1} p_{fj}\Lambda\mathcal{I}_{w-j}\mathcal{F}(t_{cj}) + \sum_{j=0}^{w-2} \sum_{k=1}^{w-j} \overline{\mathbf{P}\mathbf{f}}_{j+1}^{w-k} p_{fj}\Lambda\mathcal{I}_k\mathcal{F}(t_{cj}), \end{aligned} \quad (\text{A.5})$$

where, in order to simplify the sums, we define $t_{c0} = t_f$ and $p_{f0} = 1$. Now, we can derive the discrete equation:

$$\begin{aligned} \mathbf{fF}(\mathbf{i}) = & -\Lambda\mathcal{F}[i]\mathcal{S}[i] + p_{fw}\Lambda\mathcal{S}[i-icw]\mathcal{F}[i-icw] + \sum_{j=0}^w \overline{\mathbf{P}\mathbf{f}}_{j+1}^w p_{fj}\Lambda\mathcal{S}[i-icj]\mathcal{F}[i-icj] + \\ & - \sum_{k=1}^{w-1} \Lambda\mathcal{F}[i]\mathcal{I}_j[k] + \sum_{j=1}^{w-1} p_{fj}\Lambda\mathcal{I}_{w-j}[i-icj]\mathcal{F}[i-icj] + \\ & + \sum_{j=0}^{w-2} \sum_{k=1}^{w-j} \overline{\mathbf{P}\mathbf{f}}_{j+1}^{w-k} p_{fj}\Lambda\mathcal{I}_k[i-icj]\mathcal{F}[i-icj], \end{aligned} \quad (\text{A.6})$$

705 where the delays t_{cj} are expressed with a shift on the indexes: $i - icj$, where $icj = \lfloor t_{cj}/h \rfloor$, for $(i - icj) \geq 0$. Note, that if this latter expression is not true (that is, a negative index), the value of the index is 0. Finally, Algorithm 1 shows how to obtain the *diffusion ratio* as $Rd[i] = \frac{Iw[i]}{n}$ and the *diffusion time* D_t as the first time when all nodes receive all the chunks ($Iw[i] = N$).

References

- 700 [1] F. Bai, A. Helmy, A Survey of Mobility Models in Wireless Adhoc Networks, Scientific Literature Digital Library and Search Engine, 2008., 2008.
- [2] A. Udugama, J. Dede, A. Förster, V. Kuppasamy, K. Kuladinithi, A. Timm-Giel, Z. Vatandas, My smartphone tattles: Considering popularity of messages in opportunistic data dissemination, *Future Internet* 11 (2) (2019).
- 705 [3] N. Benamar, K. D. Singh, M. Benamar, D. E. Ouadghiri, J.-M. Bonnin, Routing protocols in vehicular delay tolerant networks: A comprehensive survey, *Computer Communications* 48 (2014) 141 – 158.
- [4] Y. Li, D. Jin, Z. Wang, L. Zeng, S. Chen, Exponential and power law distribution of contact duration in urban vehicular ad hoc networks, *IEEE Signal Processing Letters* 20 (1) (2013) 110–113.
- [5] W. Wang, V. Srinivasan, M. Motani, Adaptive contact probing mechanisms for delay tolerant applications, in: *Proceedings of the 13th Annual ACM International Conference on Mobile Computing and Networking*, MobiCom '07, ACM, New York, NY, USA, 2007, pp. 230–241.
- 710 [6] S.-H. Kim, Y. Jeong, S.-J. Han, Use of contact duration for message forwarding in intermittently connected mobile networks, *Computer Networks* 64 (2014) 38 – 54.
- [7] E. Hernández-Orallo, L. Chancay-García, P. Manzoni, C. Calafate, J.-C. Cano, Assessing social aspects of urban vehicular scenarios for improving message diffusion, in: *28th International Conference on Computer Communication and Networks (ICCCN)*, 2019, pp. 1–8.
- 715 [8] X. Zhang, G. Neglia, J. Kurose, D. Towsley, Performance modeling of epidemic routing, *Computer Networks* 51 (10) (2007) 2867 – 2891.
- [9] N. Banerjee, M. D. Corner, D. Towsley, B. N. Levine, Relays, base stations, and meshes: Enhancing mobile networks with infrastructure, in: *Proceedings of the 14th ACM International Conference on Mobile Computing and Networking*, MobiCom '08, ACM, New York, NY, USA, 2008, pp. 81–91.
- 720 [10] C. S. De Abreu, R. M. Salles, Modeling message diffusion in epidemical DTN, *Ad Hoc Networks* 16 (2014) 197–209.
- [11] A. Vahdat, D. Becker, Epidemic routing for partially-connected ad hoc networks, Technical report number cs-200006, Duke University (2000).
- 725 [12] T. Spyropoulos, K. Psounis, C. S. Raghavendra, Spray and wait: An efficient routing scheme for intermittently connected mobile networks, in: *Proceedings of the 2005 ACM SIGCOMM Workshop on Delay-tolerant Networking*, WDTN '05, ACM, New York, NY, USA, 2005, pp. 252–259.
- [13] G. Karagiannis, O. Altintas, E. Ekici, G. Heijenk, B. Jarupan, K. Lin, T. Weil, Vehicular networking: A survey and tutorial on requirements, architectures, challenges, standards and solutions, *Communications Surveys Tutorials*, *IEEE* 13 (4) (2011) 584–616.
- 730

Algorithm 1 Xspread diffusion model (numerical solution)

Input:

w : number of chunks
 n, f : Number of nodes and fixed nodes
 Λ, λ : Contact rates
 $p_n, p_f[]$: vectors with the probabilities of transmission of 1.. w chunks
 $t_c[]$: vector with the transmission time of 1.. w chunks
 t_n, t_f : waiting times for no transmission for mobile and fixed nodes
 T, h : evaluation time and step

Output:

$S[], Iw[]$: vectors with the susceptible and infected nodes (with all the chunks) from time 0 to T
 $Rd[]$: vector with the delivery ratio from time 0 to T
 Dt : diffusion time

- 1: $S[1] = n; F(1) = f;$
- 2: **firstTime** = **true**;
- 3: **for** $i=1:\lceil T/h \rceil$ **do**
- 4: $F[i + 1] = F[i] + h \times \mathbf{fF}(i);$
- 5: $S[i + 1] = S[i] + h \times \mathbf{fS}(i);$
- 6: **for** $j=1:w$ **do**
- 7: $I[j, i + 1] = I[j, i] + h \times \mathbf{fI}(j, i);$
- 8: **end for**
- 9: $Cf[i + 1] = Cf[i] + h \times \mathbf{fCf}(i);$
- 10: **for** $j=1:w$ **do**
- 11: $Cn[j, i + 1] = I[j, i + 1] + h \times \mathbf{fCn}(j, i);$
- 12: **end for**
- 13: $Iw[i] = I[w, i] + \frac{1}{2}Cn[w, i];$ // Nodes with all the chunks (including the ones communicating)
- 14: $Rd[i] = Iw[i]/n;$
- 15: **if** $Iw[i] == N$ **and** **firstTime** **then**
- 16: $Dt = h \cdot i;$
- 17: **firstTime** = **false**;
- 18: **end if**
- 19: **end for**

- [14] A. M. Vegni, A. Stramacci, E. Natalizio, SRB: A selective reliable broadcast protocol for safety applications in VANETs, in: 2012 International Conference on Selected Topics in Mobile and Wireless Networking, 2012, pp. 89–94.
- [15] J. Zhao, Y. Zhang, G. Cao, Data pouring and buffering on the road: A new data dissemination paradigm for vehicular ad hoc networks, *IEEE Transactions on Vehicular Technology* 56 (6) (2007) 3266–3277.
- 735 [16] Y. Wang, L. Huang, T. Gu, H. Wei, K. Xing, J. Zhang, Data-driven traffic flow analysis for vehicular communications, in: *IEEE INFOCOM 2014 - IEEE Conference on Computer Communications*, 2014, pp. 1977–1985.
- [17] P.-C. Cheng, K. Lee, M. Gerla, J. Härri, Geodtn+nav: Geographic dtn routing with navigator prediction for urban vehicular environments, *Mobile Networks and Applications* 15 (1) (2010) 61–82.
- 740 [18] J. Nzouonta, N. Rajgure, G. Wang, C. Borcea, Vanet routing on city roads using real-time vehicular traffic information, *IEEE Transactions on Vehicular Technology* 58 (7) (2009) 3609–3626.
- [19] V. Naumov, T. R. Gross, Connectivity-aware routing (car) in vehicular ad-hoc networks, in: *Proceedings of the IEEE INFOCOM 2007 - 26th IEEE International Conference on Computer Communications*, IEEE Computer Society, Washington, DC, USA, 2007, pp. 1919–1927.
- 745 [20] X. Lin, R. Lu, X. Liang, X. Shen, Stap: A social-tier-assisted packet forwarding protocol for achieving receiver-location privacy preservation in vanets, in: *2011 Proceedings IEEE INFOCOM*, 2011, pp. 2147–2155.
- [21] F. Cunha, A. Carneiro Vianna, R. Mini, A. Loureiro, How effective is to look at a vehicular network under a social perception?, in: *Wireless and Mobile Computing, Networking and Communications (WiMob)*, 2013 IEEE 9th International Conference on, 2013, pp. 154–159.
- 750 [22] Z. Ning, F. Xia, N. Ullah, X. Kong, X. Hu, Vehicular social networks: Enabling smart mobility, *IEEE Communications Magazine* 55 (5) (2017) 16–55.
- [23] J. Ott, D. Kutscher, A disconnection-tolerant transport for drive-thru internet environments, in: *Proceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies.*, Vol. 3, 2005, pp. 1849–1862 vol. 3.
- 755 [24] T. Le, M. Gerla, Contact duration-aware routing in delay tolerant networks, in: *2017 International Conference on Networking, Architecture, and Storage (NAS)*, 2017, pp. 1–8.
- [25] B. J. Brachman, S. T. Chanson, Fragmentation in store-and-forward message transfer, *IEEE Communications Magazine* 26 (7) (1988) 18–27.
- [26] L. Torgerson, S. C. Burleigh, H. Weiss, A. J. Hooke, K. Fall, D. V. G. Cerf, K. Scott, R. C. Durst, Delay-Tolerant Networking Architecture, RFC 4838 (Apr. 2007).
- 760 [27] M. Pitkanen, A. Keranen, J. Ott, Message fragmentation in opportunistic dtns, in: *2008 International Symposium on a World of Wireless, Mobile and Multimedia Networks*, 2008, pp. 1–7.
- [28] M. Kim, Y. G. Kim, S. W. Chung, C. H. Kim, Measuring variance between smartphone energy consumption and battery life, *Computer* 47 (7) (2014) 59–65.
- 765 [29] T. Le, Q. Zhao, M. Gerla, Fragmented data routing based on exponentially distributed contacts in delay tolerant networks, in: *International Conference on Computing, Networking and Communications, ICNC 2019, Honolulu, HI, USA, February 18-21, 2019, 2019*, pp. 1039–1043.
- [30] S. Jain, M. Demmer, R. Patra, K. Fall, Using redundancy to cope with failures in a delay tolerant network, in: *Proceedings of the 2005 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, SIGCOMM '05, ACM, New York, NY, USA, 2005*, pp. 109–120.
- 770 [31] J. Widmer, J.-Y. Le Boudec, Network coding for efficient communication in extreme networks, in: *Proceedings of the 2005 ACM SIGCOMM Workshop on Delay-tolerant Networking, WDTN '05, ACM, New York, NY, USA, 2005*, pp. 284–291.
- [32] G. Sandulescu, S. Nadjm-Tehrani, Optimising replication versus redundancy in window-aware opportunistic routing, in: *2010 Third International Conference on Communication Theory, Reliability, and Quality of Service*, 2010, pp. 192–201.
- 775 [33] C. T. Calafate, G. Fortino, S. Fritsch, J. Monteiro, J.-C. Cano, P. Manzoni, An efficient and robust content delivery solution for iee 802.11p vehicular environments, *J. Netw. Comput. Appl.* 35 (2) (2012) 753–762.
- [34] V. Palma, E. Mammi, A. M. Vegni, A. Neri, A fountain codes-based data dissemination technique in vehicular ad-hoc networks, in: *2011 11th International Conference on ITS Telecommunications*, 2011, pp. 750–755. doi: 10.1109/ITST.2011.6060154.
- 780 [35] Q. Xu, Z. Su, K. Zhang, P. Ren, X. S. Shen, Epidemic information dissemination in mobile social networks with opportunistic links, *Emerging Topics in Computing*, *IEEE Transactions on* 3 (3) (2015) 399–409.
- [36] J. Whitbeck, V. Conan, M. Dias de Amorim, Performance of opportunistic epidemic routing on edge-markovian dynamic graphs, *Communications*, *IEEE Transactions on* 59 (5) (2011) 1259–1263.
- 785 [37] L. Bracciale, M. Bonola, P. Loretì, G. Bianchi, R. Amici, A. Rabuffi, CRAWDAD dataset roma/taxi (v. 2014-

07-17), Downloaded from <https://crawdad.org/roma/taxi/20140717> (Jul. 2014).

- [38] L. Chancay-García, E. Hernández-Orallo, P. Manzoni, C. T. Calafate, J. Cano, Evaluating and enhancing information dissemination in urban areas of interest using opportunistic networks, *IEEE Access* 6 (2018) 32514–32531.
- [39] M. Piorkowski, N. Sarafijanovoc-Djukic, M. Grossglauser, A Parsimonious Model of Mobile Partitioned Networks with Clustering, in: *The First International Conference on COMMunication Systems and NETWORKS (COMSNETS)*, 2009.
- [40] T.-C. Tsai, H.-H. Chan, Nccu trace: social-network-aware mobility trace, *Communications Magazine, IEEE* 53 (10) (2015) 144–149.
- [41] Z. J. Haas, T. Small, A new networking model for biological applications of ad hoc sensor networks, *Networking, IEEE/ACM Transactions on* 14 (1) (2006) 27–40.
- [42] A. Passarella, M. Conti, Analysis of individual pair and aggregate intercontact times in heterogeneous opportunistic networks, *IEEE Transactions on Mobile Computing* 12 (12) (2013) 2483–2495.
- [43] E. Hernández-Orallo, J. C. Cano, C. T. Calafate, P. Manzoni, New approaches for characterizing inter-contact times in opportunistic networks, *Ad Hoc Networks* 52 (2016) 160–172.
- [44] T. G. Kurtz, *Approximation of Population Processes*, SIAM, 1981.
- [45] E. Hernandez-Orallo, J. Herrera-Tapia, J.-C. Cano, C. Calafate, P. Manzoni, Evaluating the impact of data transfer time in contact-based messaging applications, *Communications Letters, IEEE* 19 (10) (2015) 1814–1817.
- [46] E. Hernández-Orallo, M. Murillo-Arcila, C. T. Calafate, J. C. Cano, J. A. Conejero, P. Manzoni, Analytical evaluation of the performance of contact-based messaging applications, *Computer Networks* 111 (2016) 45 – 54.
- [47] J. Dede, A. Förster, E. Hernández-Orallo, J. Herrera-Tapia, K. Kuladinithi, V. Kuppusamy, P. Manzoni, A. bin Muslim, A. Udugama, Z. Vatandas, Simulating opportunistic networks: Survey and future directions, *IEEE Communications Surveys and Tutorials* 20 (5) (2018) 1547–1573.
- [48] A. Keränen, J. Ott, T. Kärkkäinen, The one simulator for dtn protocol evaluation, in: *Proceedings of SIMU-Tools'09*, 2009, pp. 55:1–55:10.
- [49] Z. Zhao, B. Mosler, T. Braun, Performance evaluation of opportunistic routing protocols: A framework-based approach using omnet++, in: *Proceedings of the 7th Latin American Networking Conference, LANC '12*, ACM, New York, NY, USA, 2012, pp. 28–35.
- [50] J. Herrera-Tapia, E. Hernández-Orallo, A. Tomás, P. Manzoni, C. T. Calafate, J. Cano, Selecting the optimal buffer management for opportunistic networks both in pedestrian and vehicular contexts, in: *2017 14th IEEE Annual Consumer Communications Networking Conference (CCNC)*, 2017, pp. 395–400.