

Precise: Predictive Content Dissemination Scheme Exploiting Realistic Mobility Patterns

Noelia Pérez Palma^{a,b,*}, Falko Dressler^c, Vincenzo Mancuso^a

^a*IMDEA Networks Institute, Leganés (Madrid), Spain*

^b*University Carlos III of Madrid, Spain*

^c*School of Electrical Engineering and Computer Science, TU Berlin, Germany*

Abstract

Device-to-Device (D2D) communications have expanded the way of managing available network resources to efficiently distribute data between users. D2D exploits communication alternatives, in Opportunistic Networks, based on short range wireless radio technologies such as Bluetooth and WiFi-Direct. Besides, nowadays in most urban areas, realistic human mobility is characterized by often repeated patterns that can be used to accurately predict the next visited regions—we call these regions hotspots (or Replication Zones (RZs)). In this work, we present Predictive Content Dissemination Scheme (Precise), to explore and combine the D2D paradigm along with real mobility and predictions focused on the dissemination of content among hotspots. To analyze the viability of such scheme, we show simulation results and evaluate the average content availability, lifetime and delivery delay, storage usage and network utilization metrics. We compare the performance of Precise with state-of-the-art approaches, such as Epidemic, restricted Epidemic, and Proximity-Interest-Social (PIS) routing protocols. Our results underline the need for smart usage of communication opportunities and storage. We demonstrate that Precise allows for a neat reduction in network activity by decreasing the number of data exchanges by up to 92%, requiring the use of up to 50% less of on-device storage. This comes at negligible costs. In particular, the delivery delay with Precise shows an increase with respect to epidemic dissemination schemes that varies from 0.03 seconds in the most dynamic case to at most 1.91 seconds for the least dynamic case, and which however does not hinder the possibility to use Precise for real-time applications. Regarding how contents are spread, we observe that Precise requires 2% to 20% less mobile users to carry them within a target hotspot, especially under slow dynamics. This however does not impact on the probability that mobile users entering the hotspots obtain contents, and barely shortens the lifetime of contents in our experiments from 100 minutes down to about 95, in the worst case. This demonstrates that the reduction of content availability among mobile users with Precise is either negligible or not impactful, thus guaranteeing the dissemination of contents as with legacy epidemic dissemination protocols.

Keywords: Opportunistic networks, mobility pattern, predictive communication scheme, content dissemination

1. Introduction

Our society is experiencing a massive growth in the number of active devices connected over the Internet generating vast amounts of data. For many applications, there is need to offload the communication from cellular networks to direct Device-to-Device (D2D) communications [1] through Opportunistic Networks (OppNets) [2]. This is useful, e.g., when users are experiencing poor network connection or they are unable to connect, or when the available network infrastructure cannot be trusted. For the above mentioned cases, and for the ones in which it results to be more cost-effective than infrastructure-based access, the use of D2D is favorable [3]. This also aligns with next generation virtualized edge computing concepts [4].

For instance, consider that the number of users concerned about their privacy keeps steadily growing, so that many services will rather not trust intermediary parties to distribute—and be in possess of—certain information. They would rather trust “friend” devices (e.g., devices owned by people belonging to the same community or a social network) than network infrastructures and service operators. Examples of this case range from context-aware social networking to covert communications during protests.

During the last decade, a multitude of dissemination techniques have been developed [5, 6, 7]. Survey papers such as [8, 9, 10, 11] provide deep insights into the different perspectives adopted. Nevertheless, heterogeneous and limited resources and capabilities at nodes still impose many limitations for real-world scenarios. Most importantly, the dynamically evolving network topology still determines one of the main challenges.

The presence of memory-constrained devices and net-

*Corresponding author.

Email addresses: noelia.perez@imdea.org (Noelia Pérez Palma), dressler@ccs-labs.org (Falko Dressler), vincenzo.mancuso@imdea.org (Vincenzo Mancuso)

work congestion are some of the causes of these limitations. The scheme proposed in this paper aims to introduce a significant reduction in the amount of data kept in mobile devices’ memory, along with a drastic alleviation of network traffic. We achieve this goal by cutting data exchanges down to only meaningful ones. Furthermore, we aim at exploiting contact opportunities leveraging nodes movement pattern based on typical daily routines, which leads to accurate predictions of the network users’ future behavior. We present a set of configurations that make use of the previously gathered information to manage network and device resources efficiently and, therefore, to deliver messages in a more effective manner with respect to legacy content dissemination schemes.

The social component is also a great asset to boost forwarding strategies [12, 13, 14]. Similar to Pannu et al. [15], both in the algorithmic design as well as in our evaluation, we support certain interest hotspots towards which pedestrians and vehicles are more likely to head. However, here we consider distinct mobility models with respect to that work, and substantially different timescales and dynamics, since we focus on spreading contents within hotspots and from one hotspot to another without the support of an infrastructure, rather than focusing on vehicular micro-clouds as in [15]. A key novel element of our study lays in the dyadic nature of hotspot-based dissemination, which requires understanding and supporting data exchange not only within the hotspots, and not just to move information from one hotspot to another, but rather solve the two problems at the same time. Focusing on hotspots leads us to use realistic mobility patterns and to be accurate in the management of data dissemination. In our example, population samples relate to two often visited hotspots, but our approach can be used with any number of hotspots. Furthermore, we claim that the presence of a social component in the forwarding scheme justifies the use of D2D.

Our service quality metrics are average delivery delay and content “availability”. The former is the average time needed for a mobile node to receive a piece of dissemination content after moving into a hotspot. The content availability is the probability that a piece of content be stored on mobile nodes in the hotspot and so be available for D2D dissemination to newly arrived users. As an overall objective, we want to achieve delivery delay and availability levels as if we were using epidemic diffusion schemes, except we want to reduce the overhead in terms of number of connections and use of storage on mobile devices. However, notice that availability levels above a critical threshold make little difference in terms of how probably a newcomer will obtain a content, and what really matters is to keep availability far from zero [16]. The efficiency of our proposal is evaluated in terms of on-device storage utilization, number of data exchanges required and content lifetime, the latter being the time a piece of content persists in a target area while mobile users come and go.

Note that, differently from epidemic routing schemes, any user in the hotspot is the destination of any content to

be disseminated. In this sense, heterogeneous and limited resources and capabilities of the involved nodes impose additional limitations for real-world applications. Note also that existing opportunistic schemes cannot capture the social-aware nature of the applications considered in this work. They can be used if need be, but, as shown in this paper, they end up wasting precious resources to disseminate information beyond the needs of the applications, with no tangible performance gain. For this purpose, we have designed application-dependent scheme that will serve information to users with same interests in an independent fashion and which take into account the specificity and predictability of mobility patterns by learning from past events. For example, nodes involved in a university environment will subscribe to the same specific channel and, consequently, share only related event advertisements. This detached approach, compared to state-of-the-art works where everyone’s devices are involved in the content distribution process, is crucial to avoid misusing resources from nodes that are not willing to cooperate to the routing process as well as spamming users with different interests.

In order to assess the performance of our Predictive Content Dissemination Scheme (Precise), we have implemented a simulation model, analyzed the collected data describing occurring events, and assessed and compared the performance of Precise to existing dissemination strategies. Our results provide essential insights on how to manage available resources in an efficient manner according to the studied scenarios and mobility requirements.

Our main contributions can be summarized as follows:

- We design a novel and powerful yet lightweight scheme, named Precise, whose algorithms improve data forwarding and storage efficiency in opportunistic communication scenarios by leveraging social behaviors and mobility predictions.
- With Precise, we propose a content forwarding and storing scheme that we implement in a custom-made simulator, which uses state-of-the-art approaches for mobility modeling based on maps and real user mobility; this approach allows us to experiment also with epidemic diffusion and benchmarking algorithms with realistic mobility traces.
- We assess the performance of Precise vs state of the art schemes in terms of content availability, storage load, network resource utilization, content lifetime and losses over time, and delivery delay.
- We compare Precise with three other benchmarks, such as, content dissemination restricted to Replication Zones (RZs), a classic full-fledged epidemic scheme (denoted as Epidemic), and the Proximity-Interest-Social (PIS) routing protocol, which is the state of the art algorithm for efficient epidemic routing. We show that our algorithm clearly outperforms other solutions by reducing between 65-92%

the needed number of connections and halving the use of storage. Precise produces comparable content availability most of the time, and even when we obtain less availability, we do not observe any perceivable quality degradation. Precise incurs negligible content lifetime reductions (up to 5%), and causes delivery delay increases of the order of a small fraction of a second most of the time, which results in Precise being suitable for real-time applications (e.g., to deliver warnings and advertisements timely when entering an area).

2. Background and Related Work

For more than a decade now, forwarding strategies for D2D communications in opportunistic networks have represented one of the most challenging questions to cope with in terms of content dissemination performance, due to high node mobility, dynamic evolution of networks and devices heterogeneity. In this paper, we mainly focus on providing enhanced heuristics and combinations of those for data sharing among devices contributing to resource usage efficiency and dissemination effectiveness. Devices involved in such scenarios are mostly carried by users who are considered to influence, to some extent, the behavior of their smartphones, tablets, etc. According to this perspective, previous works on content dissemination in opportunistic networks can be categorized into four main groups, as discussed in what follows.

2.1. Context-oblivious heuristics

Early logical and elementary techniques for content distribution can be classified into *context-oblivious heuristics*. For example, works like Spyropoulos et al. [6], Grossglauser and Tse [5] proposed different schemes that constrain the number of content copies in the system to improve bandwidth, storage capacity, and energy consumption. Beyond the previous concept and given the fact that dropping messages too early may reduce the speed of information diffusion, Hernández-Orallo et al. [17] introduce a dynamic expiration time setting to limit the effects of early content loss. With these techniques the authors try to overcome the shortcomings of basic flooding-based schemes but still pose limitations when mobility patterns are restricted. Chancay-García et al. [18] study the impact of contact duration for message broadcasting. They leverage the division of large messages into smaller parts to improve dissemination and demonstrate that a fixed size partition is the best approach. Our scheme does not include context-oblivious mechanisms since we believe introducing context and social-aware heuristics better adapt to the mobility dynamics of most urban scenarios, as explained in what follows and demonstrated in Section 6.

2.2. Context-aware heuristics

Obviously, there was still a need for more sophisticated methods to solve dissemination challenges in frequently disconnected networks, not only aiming at reducing flooding and overhead but also effectively distributing data content, i.e., providing valuable content to potential nodes at acceptable time delay. For that purpose, an advanced sort of *context-aware heuristics* to achieve smarter decision making processes has been explored. For instance, Dhurandher et al. [19] present a history-based routing protocol that exploits nodes mobility information to predict the best next hop for content exchange. Lindgren et al. [20] and Barrett et al. [21] combine history of previous encounters with probabilistic techniques. In both papers, nodes decide to which peer they will forward the content assessing various parameters to compute the probability that the chosen node will deliver the content to its destination. Furthermore, Burns et al. [22] incorporate information not only about past encounters but also about previous visited regions.

More recently, research studies have coupled several of the cited features to develop more accurate techniques for specific D2D communications scenarios. For example, Liu et al. [23] introduce a distributed online algorithm that focuses on the optimal node pause strategy in order to select the best transmission peer. Yamamoto et al. [24] propose a method that adaptively adjusts the transmission timing and effective radius of the area in which information is shared. This decision is based on terminal density and terminal encounter rate in order to estimate further communication opportunities. In Rizzo et al. [16], the authors present an information theoretical model of the storage capacity of probabilistic distributed storage systems where nodes are only allowed to exchange content based on their current position and storage capacity.

Some other works refer to this D2D paradigm with the *Floaty Content* term. For example, Pérez Palma et al. [25] and Rizzo et al. [26] go further and develop Android applications that support infrastructureless distributed content sharing among wireless devices using state-of-the-art technologies, such as Bluetooth and Wi-Fi Direct. The authors also discuss results gathered from real experiments and conclude that high device densities determine the performance.

What is missing in all these studies is the social factor, which we instead leverage to increase the efficiency of dissemination schemes. In this work, we assume that nodes move according to similar patterns every weekday following social behaviors like going to their work place, returning home or to some other frequently visited places. This allows the application to predict with high accuracy whether passing information to a user is going to be useful or not, which reduces unnecessary information exchanges typical of epidemic schemes.

2.3. Social-aware heuristics

Several studies emerged using *social-aware heuristics*. Researchers started developing dissemination strategies initially based on the idea that human mobility presents certain behavioral patterns that can benefit forwarding decision making.

A good example has been introduced by Boldrini et al. [27]. They present ContentPlace, a system that defines social-oriented policies and analyzes the behavior of users in pursuance of optimizing content availability by locating data content in appropriate spots. Boldrini et al. [12] also exploit a combination of social information to pick the most suitable next hop based on the similarity of each peer node context to the destination context. Ying et al. [28] introduce a Markov chain model of users' social ties. They formulate the problems of unfair traffic distribution and unfair delivery success ratio based on the evaluation of users' social relationship. Rahim et al. [29] present a social Acquaintance based Routing Protocol (SARP) for Vehicular Social Networks (VSNs). SARP considers the global and local community acquaintance of nodes to make a forwarding decision. Moreover, Ullah et al. [30] developed a reputation mechanism that calculates a trust-score for each node based on its social-utility behavior and contribution to the network. Built on that idea, the authors propose a Trust based Dissemination Scheme (TDS) for Emergency Warning Messages (EWMs) to detect malicious alarms. Hui et al. [13] analyze the contact patterns between nodes and infer the social communities which these nodes belong to. This system aims to exchange data to nodes belonging to the destination community based on previous context information and assuming sociable nodes will have more chances to forward the content to its destination. Vegni et al. [31] assess a previously introduced probabilistic-based broadcasting scheme for vehicular communications leveraging the computation of nodes' social degree. They demonstrate its effectiveness in packet transmission reduction while guaranteeing network dissemination in realistic scenarios with real traffic traces. They also compare it with state-of-the-art schemes showing a significant improvement in terms of delivery ratio.

A very relevant work in this field is also introduced by Xia et al. [32], where authors propose PIS a routing protocol based on three different social factors, and disclosing next slots social information, in order to decide the best next hop for content sharing. They present their results applying the proposed approach to SIGCOMM09 [33] and INFOCOM06 [34] data sets. The results show that PIS outperforms other well-known protocols such as Epidemic [7], PROPHET [20] and SimBet [35]. Same authors developed a similar approach in [36] that integrates vehicles' social factors into their geographical information. They introduce a new concept called geo-social distance and combine it, among other processes, with the message copy control protocol used in PIS.

Our work is partially shared with this category. However, unlike previous mentioned works, we use only informa-

tion from past traces in order to predict future positions of the nodes and make decisions according to it. Furthermore, we also take into account for how long nodes remain in their positions based on typical social standards like 7-8 hours work day, 7-9 hours sleep, etc. Filling the gap of previous approaches, we consider a set of nodes to be the final destination of the data content instead of targeting for an individual. We assume that the pieces of information shared using our paradigm will be relevant for the whole portion of the population subscribed to a given communication channel. We are able to significantly reduce network load by leveraging nodes mobility predictions and light computation for decision making, contrary to existing social-based data dissemination approaches, which still fail to achieve efficient data broadcasting due to high volumes of overhead and redundant connections.

Part of our work is also devoted to real scenarios, for instance Rome city center. We have worked with real taxi cabs traces obtained from [37] and applied Precise to carry out content dissemination. Additionally, we have implemented the previous mentioned PIS approach to compare with our solution over a more realistic scenario, closer to what can be found in urban areas.

2.4. Cognitive heuristics

Cognitive heuristics conform to a whole new set of forwarding algorithms to which researchers are paying great attention now. The central concept behind cognitive science applied to forwarding protocols is to build algorithms based on human information processing schemes.

In this direction, Mordacchini et al. [38] introduce Service Channel (SCH). Their proposal is to evaluate not only the importance of the content according to the individual but also take into account the judgement of its community. In a similar way, Khelifi et al. [39] focus on vehicular networks from the Information-centric networking (ICN) perspective and discuss the role of Named Data Networking (NDN) providing a detailed and systematic review of NDN-driven Vehicular Ad Hoc Network (VANET).

Our perspective focuses on how valuable a piece of content is for the population at a given time. For example, if a node is traveling to a hotspot where the content is relevant, after a too long travel period, it will be considered as outdated.

3. A Social-aware Opportunistic Dissemination System

We study the dissemination of information between mobile users within a geographical context characterized by the presence of separated hotspots inside which the disseminated information is relevant.

We take the case in which mobile users opportunistically leverage D2D and do not rely (because they cannot or do not want to) on the support of a network infrastructure and controllers running outside the mobile devices. Instead,

users leverage the knowledge about social behavior of other users and the possibility to predict their position in the short run. We refer to the information to be disseminated as *pieces of content* (or *contents*, for short), and we assume that each piece of content is exchanged via a single message, which contains a complete set of instructions and data that can be processed by a user application. Hence, each piece of content can be disseminated independently of the others. The specific mobile dissemination framework and the system model are presented in what follows.

3.1. Infrastructureless multi-hotspot mobile dissemination

The kind of mobile application we target makes use of contents carried and spread by mobile users via D2D. Each content is of use within any of two or more hotspots, and the goal of the system is to make contents available to devices entering a hotspot as soon as possible, and certainly before they leave. Therefore, mobile users need to spread contents not only within the hotspots in which they are, but also carry or forward them towards other hotspots while using precious and limited storage and D2D communication resources parsimoniously.

Note that the goal of the dissemination scheme described does not consist in reaching a specific destination device nor a fully indiscriminate epidemic broadcast of the pieces of content, but rather in making the information available in the hotspots, for devices passing by. Therefore, what is important in the considered dissemination scenario is not the time to delivery to a specific destination a piece of content since its generation instant, but the number of devices that pass through the hotspots and get the piece of content. It is also important to evaluate the time that elapses since when a user enters a hotspot to when it gets an available piece of content (which can be 0 in case the content was obtained before entering the hotspot). We call *availability* of a hotspot, for a given piece of content and at a given time instant, the ratio between the number of mobile devices within the hotspot possessing the content and the total number of devices in that hotspot. High availability values are desirable, but it is not strictly necessary to have an availability close to 1 in order to guarantee that the content persists while mobile users enter and leave the hotspots. It is however important that the availability be far from zero, otherwise newcomers will hardly obtain a copy of the content. In Section 5 we will formally define how to measure the availability.

3.2. System model

We consider a planar region with topological constraints for the mobility of users, e.g., a 2D city map with paths, buildings and obstacles. We assume that hotspots are disk-shaped areas within the selected region. Hotspots are also referred to as RZs in the rest of the article. The radius of an RZ disk is denoted by r_{RZ} , and we assume that there are K RZs, each denoted as RZ_k , $k = 1, 2, \dots, K$. RZs are visited regularly by mobile users, and there are $N_k(t)$

mobile devices in RZ_k at time t . More in general, in a planar region, we identify a large set of points where users can *dwell*, which we refer to as dwelling points. Some dwelling points, but not all of them, are within the RZs. Those points are selected based on the nature of the place (e.g., they correspond to an apartment, a university building, an office, etc.). Dwelling points within an RZ are chosen by mobile users uniformly at random, because we consider that an RZ is a homogeneous area including equally important dwelling points. However, each RZ has a given probability of being visited as a whole, and different RZs have different probabilities of being visited. Dwelling points outside the RZs are chosen uniformly at random, with a total probability equal to 1 minus the cumulative probability to select dwelling points within RZs. The users move from one dwelling point to another by following a path on the planar region, which is not necessarily a straight line due to topological constraints (e.g., in a taxi scenario, devices can only follow the roads reported on the city map). The specific mobility pattern reflects the social behavior of mobile users by specifying the probability to visit a random dwelling point within an RZ or to move towards a random destination outside the RZ. To realistically model user patterns, we consider that users alternate movements and pauses, and we consider two cases: (i) pedestrian mobility with synthetic traces alternating moves over shortest path trajectories at a constant speed (chosen uniformly at random for each move) and pauses with uniformly distributed duration; and (ii) trace-based vehicular mobility, in which the speed and trajectories reflects realistic traffic conditions, and the duration of pauses represent realistic inactivity periods of taxi drivers.

Since users express *stochastic preferences* when they decide to move to the next dwelling point, and preferences depend on the profile of the user, we assume that it is possible for a mobile device to predict with high accuracy where it will be dwelling next, given that it is at a certain location. Likewise, when a device meets another one and they are in transmission range, i.e., within T_r space units, it can not only start exchanging contents, but also predict whether it will reach or not a new RZ within a certain time. The user can rely on predictions to decide whether to probe the other user and initiate a content exchange or not. The specific prediction mechanism is out of the scope of this work, but we remark that devices can build statistics or use machine learning (e.g., Q-learning) to estimate if and when they will reach a hotspot.

With the above, it is clear that content exchange can be limited to cases in which mobile users are within an RZ or predicted to move to RZs soon. The dissemination of pieces of content outside RZs, in general, is far less critical than in the case of legacy opportunistic routing schemes. As such, we claim that, although traditional opportunistic routing schemes would serve the purposes of the described application scenario, more specific schemes are needed to make content dissemination efficient when the pieces of content are relevant for the RZs only. Indeed, existing

schemes cannot be efficient in terms of use of resources, i.e., the way they use connection opportunities and buffer space on mobile devices, which are often limited resources. To this purpose, in the next section we propose a new content forwarding and storage scheme that leverages the social behavior of mobile users, and specifically their ability to predict their location in the near future.

For what concerns data exchange, we assume that mobile nodes continuously send beacons and scan the wireless spectrum looking for beacons sent by other peers, like, e.g., in a Bluetooth system. For simplicity, we assume that the transmission range is fixed and constant for all devices, because we assume that D2D connections are established only upon the detection of a strong link. We also assume that, once a connection is established, mobile nodes exchange pieces of content simultaneously in the two directions, at a constant speed, which is realistic if again we consider that only strong links are used. As a consequence, the impact of channel errors and transport protocol dynamics is neglected, also because we assume short range transmissions of small pieces of content (and one might think of using smart transport protocols like QUIC [40] to make this assumption realistic even in the presence of large pieces of content). However, a content transfer can fail when mobile devices exchanging data get out of their transmission range before the transfer is complete. In other words, differently from many epidemic-like dissemination schemes, we assume that content transfer is not instantaneous.

The notation used in the rest of the paper is presented in Table 1 with a short description of each quantity later used in algorithms and mathematical expressions. Full details are given in the following two sections, which describe Precise and the relevant key performance indicators, respectively.

4. Precise: Data Communication and Storage Paradigm

Due to frequent link disruptions in opportunistic networks, the fundamental forwarding approach is to adopt pervasive forwarding solutions.

Epidemic spreading techniques, for instance, provide the most elementary and effective manner of content dissemination where nodes simply exchange content at any given opportunity [12, 7]. In general, at any encounter nodes will try connect and send content to their peers, sometimes restricted to a zone of interest. However, epidemics-based dissemination schemes introduce a high overhead, which not only causes network congestion but also high energy consumption at each node. Often, the number of content replicas and connections exceeds by far what would be essential for an efficient distribution of the data content in a realistic environment. For that reason, our work introduces a simple yet smart scheme to avoid some of the unnecessary connections and content replication, focusing at the same time on finding more beneficial exchange opportunities with no significant increase in computational cost.

Table 1: Notation

Symbol	Meaning
α	Discount coefficient of the Autoregressive (AR) filter (adapted to the distance between Autoregressive (AR) updates so as to obtain an exponential decay of past values with T).
A_{ki}	Availability of content i in RZ_k , i.e., the fraction of nodes that possess content i with respect to the total number of nodes inside the k -th RZ.
$A(t)$	Mean availability computed over all existing contents and RZs, at time t .
$\bar{A}_i(\tau)$	Time-average availability for content i generated τ seconds after its injection in the network over all RZs
$\bar{A}_G(\tau)$	Statistical mean of the time-average availability for the group of contents G , computed for content lifetime equal to τ .
C_i^j	Binary variable indicating whether a certain content i is available at node j .
C_{ki}	Number of nodes possessing content i within the k -th RZ.
C_{te}	Time elapsed since a node has left an RZ without re-entering in another RZ.
$D(t)$	Set of contents injected in the network until time t .
$F_k(t_n)$	Autoregressive (AR) filtered value of $M_k(t_n)$.
G	Target group of contents.
g_i	generation time of content i .
K	Number of RZs.
$L_j(t)$	Buffer load of a node j .
$L(\tau_1, \tau_2)$	Time-average of the buffer load of all nodes, for interval $[\tau_1, \tau_2]$.
$M_A(t_n)$, $M_B(t_n)$	Number of contents that node A (or B) attempts to retrieve upon a connection is established at t_n .
$M_k(t_n)$	Number of contents to exchange upon a connection is established at time epoch t_n in the k -th RZ ($k = 0$ outside of RZs).
N_k	Set of nodes within RZ_k .
P_k	Probability to decide to exchange content upon a meeting, with Precise.
r_{RZ}	RZ radius.
$R(t)$	Mean fraction of injected contents held by a node at time t .
RZ_k	The k -th replication zone.
t, τ	Continuous values of time.
t_n	Current time epoch (discrete value). n is the time slot index
T	Exponential decay time of the Autoregressive (AR) filter.
T_e	Maximum time that a node can spend outside an RZ before emptying its buffer when using Precise; similarly, nodes outside RZs accept to receive contents if they predict to reach an RZ within T_e .
T_r	Transmission range of mobile devices.
$x(t_n)$	Variable containing the time epoch of the last connection, as observed at time t_n .

A basic example of an epidemic strategy, which we will use as baseline, consists of allowing nodes to exchange content with their peers when in communication range and only within an RZ. Nodes within an RZ can always keep the content they are carrying. Once they leave the RZ, they will automatically drop all contents in order to free-up resources. In the following, we introduce a set of more advanced predictive scheme, Precise, which helps (a) to also carry content to other RZs, (b) to specifically pass content to nodes “going” towards an RZ, and (c) to drop content after some expiration time.

4.1. Forwarding scheme of Precise

The way nodes forward their content in Precise varies depending on which zone they are located in. In our forwarding algorithm (cf. Algorithm 1), nodes within an RZ can always exchange content and nodes in the outer area are only allowed to exchange contents if any of them is likely to reach an RZ before a time expiration threshold T_e , which represents the maximum time during which nodes keep running their data exchange application while they are outside RZs. Since we identify similar node mobility patterns during weekdays, in order to predict whether the nodes are likely to visit an RZ at a given time, we analyze the information obtained from past traces at coinciding times of the day with a certain probability of failure. We do not use future information but we assume that the devices can predict their future position based on statistics collected in the past. Moreover, in case of nodes that are moving, they know where they are going (because they have selected a precise destination) and can reasonably predict when they will arrive. This process is represented in Algorithm 1 with the call to the function *isVisitingRZ(node/peer)*. More precisely, what the function *isVisitingRZ(node/peer)* does is, given that a node’s speed is constant, get its value and check whether the node, moving at its current speed, will enter an RZ before T_e expires. In case one of the node’s next hop lands inside one of the RZs, the output of the function *isVisitingRZ(node/peer)* will be a True value flag. When nodes are not expected to visit the RZs we assume the data are not relevant for them.

With Precise, when two nodes establish a connection, they will only transfer those pieces of content that are missing at the peer node. The order in which the contents are transferred is random: prior to the exchange, both nodes content lists are shuffled to prevent certain pieces from being repeatedly exchanged in the first place. This way, we guarantee that all contents have the same probability of being selected.

In case the data of one node do not completely fill their assigned capacity, the remaining quota will be relocated to the peer node. The established connection stays active until both peers have transmitted the total amount of contents. Therefore, connections are only interrupted in two cases: when nodes move away from each other beyond the transmission range or when both nodes fill up their storage capacity during the connection.

Besides, if nodes belonging to an established connection have nothing to exchange or their storage capacity is already full, the connection will be dropped right after a small fixed interval. Such interval represents the time needed to check each node status, which cannot be informed to the peer in advance with any current technology. In general, when a connection is interrupted, incomplete file transfers are dropped.

4.2. Decision making process in Precise

Decision making is a local process, running at each mobile node, and is based on the computation of the number of contents that nodes have to exchange when they meet, i.e., the number of missing contents at a generic node pair (A, B) , indicated as M_k ,¹ where $k = 0, 1, \dots, K$, depending on which area the nodes are located in the k -th RZ ($k = 0$ means that the nodes are outside any RZ):

$$M_k(t_n) = M_A(t_n) + M_B(t_n), \quad (1)$$

where t_n represents the current time epoch, and $M_A(t_n)$ (respectively, $M_B(t_n)$) is the number of contents that node

¹In the notation, we omit the dependency on the node when not necessary, but remark that each node has its own version of M_k and of the derived quantities.

Algorithm 1 Predictive forwarding scheme

Input: $\{m_0, m_1, \dots\}$: shuffled list of node’s neighbors
 T_e : maximum amount of time for data exchanging outside RZs
 C_{te} : node’s expiration time counter

```

1: if node in RZ then
2:   for peer in  $\{m_0, m_1, \dots\}$  do
3:     if peer not busy and peer in RZ then
4:       dataExchange(node, peer)  $\triangleright$  Local node
         exchanges data with peer
5:     break
6:   end if
7: end for
8: else
9:   if node not busy and node  $C_{te} < T_e$  then
10:    for peer in  $\{m_0, m_1, \dots\}$  do
11:      if peer not busy and peer  $C_{te} < T_e$  then
12:        node_prediction  $\leftarrow$  isVisitingRZ(node)
13:        peer_prediction  $\leftarrow$  isVisitingRZ(peer)
14:        if node_prediction or peer_prediction
           then
15:          dataExchange(node, peer)  $\triangleright$  Local node
             exchanges data with peer
16:          break
17:        end if
18:      end if
19:    end for
20:  end if
21: end if

```

B (respectively, A) possesses and are missing in node A (respectively, B).

Considering that not all the previous pieces of information might be interesting at a given time, a low pass filter that covers a predefined previous amount of events is needed, specially to filter out some noise from instantaneous measurements, as done in any real system. We then use $F_k(t_n)$ as the mean observed number of contents to exchange, which is obtained by applying an Autoregressive (AR) filter to $M_k(t_n)$ when a connection occurs, i.e.:

$$\begin{cases} \alpha = e^{-\frac{t_n - x(t_{n-1})}{T}}, \\ x(t_n) = t_n, \\ F_k(t_n) = \alpha F_k(x_{n-1}) + (1 - \alpha) M_k(t_n), \end{cases} \quad (2)$$

where t_n is the current time epoch, $x(t_n)$ stores the time epoch of the last connection started until t_n , and α is an adaptive value that accounts for the time elapsed in between two consecutive connections, so that the AR filter operates with an exponential decay time T . This corresponds to a negative exponential decrease of the importance of old samples. We used the time constant T of the order of hours because we follow realistic human patterns, which have to be measured in hours. If no connection occurs at t_n , the values of $F_k(t_n)$ and $x(t_n)$ are set as their respective values at t_{n-1} .

We need to wisely choose the value of T according to the amount of previous encounters that nodes are going to consider in order to derive the average number of contents that were exchanged in the past. Then, whatever happened before the decay time does not practically affect the value of the current average.

By computing $F_k(t_n)$ and comparing it with previously computed values, we tune an exchange probability P_k of actually starting a content exchange, i.e., the meeting devices might decide to skip a content exchange, to save resources. To this purpose, we use Algorithm 2. The proposed algorithm uses a negative control feedback: the more contents to exchange, the less nodes need to connect and exchange (because contents are already present in the scenario). The opposite is true when a node sees less contents than in the past: this is taken as a sign that less

Algorithm 2 Computation of Exchange probability per RZ

Input: t_n : current time slot

$F_k(t_n)$: current mean number of contents to exchange

$F_k(t_{n-1})$: previous mean number of contents to exchange

P_k : probability to exchange content

- 1: **if** $F_k(t_{n-1}) < F_k(t_n)$ **and** $P_k > 0$ **then**
 - 2: $P_k \leftarrow P_k - 0.01$ \triangleright *Decrease probability*
 - 3: **end if**
 - 4: **if** $F_k(t_{n-1}) > F_k(t_n)$ **and** $P_k < 0.1$ **then**
 - 5: $P_k \leftarrow P_k + 0.01$ \triangleright *Increase probability*
 - 6: **end if**
-

contents are around, so that the nodes must help the system more, by connecting and exchanging more frequently (with higher probability). The constant step chosen to adapt the exchange probability at every connection is set to 0.01, and we bound P_k to the interval $[0, 0.1]$ according to the sensitivity analysis presented in Section 6.3. Those values score a good tradeoff between avoiding large oscillations and adapting fast while reducing the number of data exchanges without paying in terms of dissemination performance.

It is important to note that nodes operate with different probability values depending on which RZ they are. This is enforced because nodes present different mobility patterns and therefore heterogeneous information exchange behaviors that will be more accurately analyzed separately.

4.3. Local storage management

To make smarter storage management decisions, we define a scheme to either preserve or drop the content from nodes' local buffers. In Precise, nodes within an RZ can always keep the content. If a node is in the outer area and visiting an RZ after a long period of time, we assume the content stored on its local buffer will be outdated and, thus, irrelevant for the RZ. Therefore, nodes should eventually discard such content to also reduce the resource consumption. We perform this by allowing nodes leaving an RZ to keep their stored content for a certain time, i.e., until the *time elapsed* since leaving the RZ reaches a maximum allowed value, which to be consistent with the forwarding scheme described in Section 4.1, is set to T_e .

Our storage scheme (described in Algorithm 3), along with the forwarding scheme, seeks to favor content availability by allowing nodes traveling between RZs to carry and exchange data in advance to other nodes heading down the RZs. Besides, nodes returning to the same RZ after a period of time shorter than T_e will also keep their data alive.

5. Key Performance Indicators

There are many parameters in Precise that can be fine-tuned to optimize the system. In the following, we briefly discuss the key performance indicators of the system.

Algorithm 3 Local storage scheme

Input: r_{RZ} : radius of RZ

T_e : max amount of time for data exchange outside RZs

C_{te} : node's expiration time counter

- 1: $pos \leftarrow getPosition()$ \triangleright *Set node's position*
 - 2: **if** pos not within r_{RZ} **then**
 - 3: **if** $C_{te} == T_e$ **then**
 - 4: $dropData()$ \triangleright *Drop all data*
 - 5: **else**
 - 6: $C_{te} \leftarrow C_{te} + 1$ \triangleright *Keep data and increase C_{te}*
 - 7: **end if**
 - 8: **end if**
-

We use $C_i^j(t)$ to denote a binary variable indicating whether a certain content i is available at node j at time t . Therefore, the number of nodes possessing content i within RZ_k is expressed as

$$C_{ki}(t) = \sum_{j \in N_k(t)} C_i^j(t); \quad (3)$$

note that $C_{ki}(t)$ is the number of replicas of content i available within RZ_k .

We therefore measure the availability per RZ and per content, $A_{ki}(t)$, as the fraction between the number of nodes that possess content i and the total number of nodes inside the RZ, i.e., nodes with contents outside the RZ are not considered. With the above, the availability at time t for content i in RZ k is expressed as

$$A_{ki}(t) = \frac{C_{ki}(t)}{|N_k(t)|}, \quad (4)$$

where $|\cdot|$ denotes the number of elements in a set. To evaluate the overall scheme, we will use the mean content availability at time t , which is the statistical average of the A_{ki} values computed over all RZs and contents, i.e.:

$$A(t) = \frac{1}{K|D(t)|} \sum_{k=1}^K \sum_{i \in D(t)} A_{ki}(t), \quad (5)$$

where $D(t)$ denotes the set of contents injected in the network until time t .

Another relevant metric is the time-average availability for content i generated at time g_i , computed τ seconds after its injection in the network over all RZs. This quantity can be expressed as

$$\overline{A}_i(\tau) = \frac{1}{\tau} \int_{g_i}^{g_i+\tau} \frac{1}{K} \sum_{k=1}^K \frac{C_{ki}(t)}{|N_k(t)|} dt, \quad (6)$$

and the overall statistical mean of the time-average availability (the *total content availability*, for short) for a target group of contents (denoted by G) is the statistical average of per-content availability values $\overline{A}_i(\tau)$, $\forall i \in G$:

$$\overline{A}_G(\tau) = \frac{1}{|G|} \sum_{i \in G} \overline{A}_i(\tau). \quad (7)$$

It is important to note that, at content generation time ($\tau = 0$), the total content availability \overline{A}_G is low given that each piece of content belongs to a unique node before the spreading process starts. Thus, the availability curve over time undergoes a transient period prior to stabilizing according to the system capacity.

It is also critical to understand the load $L_j(t)$ of each node's local buffer, so to compare the efficiency of the different configurations applied to the system. The load of a node j is defined as

$$L_j(t) = \sum_{i \in D(t)} C_i^j(t), \quad (8)$$

and the following quantity expresses the mean fraction of injected contents held by a node at time t :

$$R(t) = \frac{1}{\sum_{k=1}^K |N_k(t)|} \sum_{j \in \cup_{k=1}^K N_k(t)} \frac{L_j(t)}{D(t)}. \quad (9)$$

Taking into account the nodes' load, we can observe the saturation value of the system as a whole, using the total average load L over the time interval $[\tau_1, \tau_2]$, which is defined as

$$L(\tau_1, \tau_2) = \frac{1}{\tau_2 - \tau_1} \int_{\tau_1}^{\tau_2} \sum_{k=1}^K \sum_{j \in N_k(t)} \sum_{i \in D(t)} C_i^j(t) dt. \quad (10)$$

Finally, other important performance indicators are: (i) the number of connections used by the nodes, which depends on mobility, frequency of meeting events and availability of nodes to connect when they are in transmission range, (ii) the fraction of contents that disappear from the network after being injected, (iii) the time during which a content survives in the network, and (iv) the delivery time needed by a node to receive a content after entering an RZ. The number of connections measures the communication load due to the dissemination process, while the loss count and the content lifetime express the ability of the dissemination process to keep information available over time without the help of any infrastructure. The delivery time tells how efficient the dissemination is with respect to RZ visits.

6. Performance Evaluation

We have built an opportunistic content dissemination simulator using Python, to reproduce the D2D-enabled application scenarios described in Section 3, with the algorithms of Section 4. New features are developed in our custom simulator that are not present in state-of-the-art simulators. For instance, the ability to exchange contents combining different circumstances, such as, certain periods of time and selected places in the scenario, basing the previous decisions on predictive information. This way, we can flexibly explore diverse scenarios according to our specified input parameters. It also facilitates the configuration of more complex scenarios, the post-processing of several metrics and opens doors to further modular extensions [16].

6.1. Geographical and mobility scenarios

To evaluate Precise, we consider two realistic geographical environments, which are suitable to simulate pedestrian and vehicular (taxi) mobility, respectively. The structure of our application scenarios is composed of the following mutually associated components:

- The scenario consists of a squared 2D region, extracted from either a map of Paderborn, Germany (for pedestrian mobility cases) or from the one of Rome, Italy (for taxi mobility patterns).

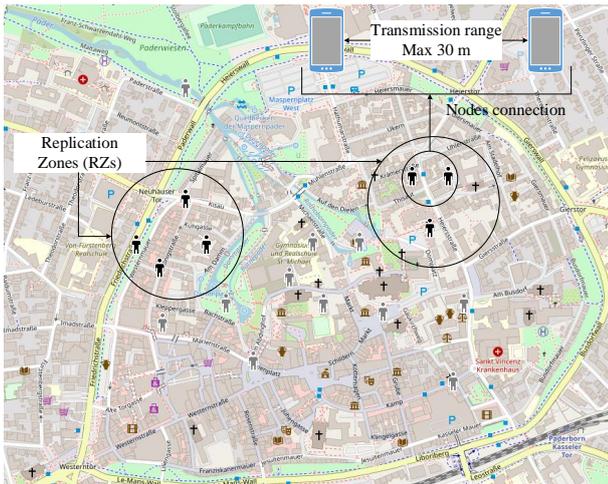


Figure 1: Scenario of Paderborn including two RZs and 50 users moving within the scenario

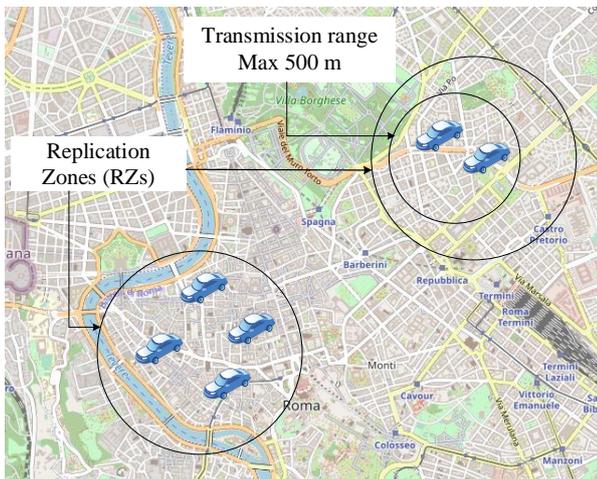


Figure 2: Detail of the full scenario of Rome including two RZs and up to 199 mobile devices moving on taxi cabs within the scenario.

- Within this region, we define two (circular) RZs that represent hotspots where disseminated data is especially valuable for nodes traversing them. RZs are placed at opposite locations of the maps and all data contents are aimed to travel between these two zones.
- The scenario also contains mobile users (*nodes*), that move according to two different methods: for pedestrians, we use synthetic traces based on the city map of Paderborn, while for taxis, we use real traces obtained from [37]. In case nodes follow synthetic traces, they are uniformly distributed in space with a certain user density. In all cases, nodes have an accurate estimate of their position and can forecast the time at which they will reach RZs.

Figs. 1 and 2 show the map sections used in this paper, from the city of Paderborn and Rome respectively. In the case of Rome, the figure only shows a detail of the full map. The traces used cover a larger area which cannot be modeled as a uniform square region. However, the taxis spend 67% of their time in a square area of dimensions $4\text{km} \times 6\text{km}$ within which we selected the RZs. Figures depict, approximately, the size and location of both defined RZs together with the type of nodes of each scenario, being pedestrians in the first case and vehicles in the second. We also indicate in the figures the node transmission range of each setting (see Table 2 for numerical values). We initialize the system based on a set of parameters explained in Section 6.2, which can be tuned according to the desired scenario. Note that the structure of a scenario can take more complex configurations, composed of one or multiple RZs and also supports multiple types of content per RZ.

We have used the ONE Simulator [41] to generate pedestrian traces based on the map of the center of Paderborn and a set of manually selected dwelling points which corresponds to gathering places in the city. Moreover, we consider three pedestrian mobility models, which show the different dynamicity levels typical of a *Businessman*, a *Clerk*, or a *Student*. *Business's* mobility assumes rather long pause time and inhomogeneous RZ visiting probabilities. This leads to a less routine movement of the nodes, which resembles business people's mobility. *Clerk's* mobility increases pause periods and RZ visiting probabilities. This mobility represents the case of commuters. *Student's* mobility represents an intermediate case, with nodes moving between positions at shorter periods and visiting a RZ with a probability higher than in the *Businessman's* mobility case. This reflects a university environment.

In the case of taxi cab's mobility in Rome, we count on 5 sets of 2 days traces each, obtained from cabs mobility measurements [37]. These traces were collected during consecutive complete days where cabs reported their position with 15s granularity. By previously studying the sets of traces, we could observe the most visited regions of the city along different days, such as the airport and the main streets of the city center. Thus, defining them as our RZs. The key characteristic of the traces from Rome, compared to Paderborn scenario, is the dynamicity of the nodes. In this case nodes are vehicles, not pedestrians, which move at a higher speed. This will entail larger number of visited places and nodes encounters but will impose shorter contact intervals.

In all cases, we assign nodes to two groups, each with different RZ visiting probabilities, so that each group has a preferred RZ to visit. This way, nodes will likely visit both RZs, but will stay for longer periods within one of them. By so doing, we test the ability of Precise to efficiently move contents between two disjoint hotspots located within a larger area.

Table 2: Input parameters

Parameter	Paderborn	Rome
Region size	2.25 km ²	24 km ²
Radius of replication (r_{RZ})	200 m	1000 m
Transmission range (T_r)	30 m	500 m
Channel rate	10 Mbit/s	10 Mbit/s
Memory limit	infinite	infinite
Number of users in the region	50	199
User speed	0.5 – -1.4 m/s	From traces
Slot length	1 s	1 s
Content size	5 Mbit	5 Mbit
Injected contents	2	2
Periodic injection time	1000 s	1000 s
Elapsed time (T_e)	400, 600 and 800 s	400, 600 and 800 s
Pause time	2–5 h (Businessman) 3–8 h (Clerk) 15–30 min (Student)	4.6 min on average
RZ visiting prob	0.2 and 0.1 (Businessman) 0.7 and 0.2 (Clerk) 0.4 and 0.3 (Student)	0.036 (RZ_1) 0.441 (RZ_2)

6.2. Parameter setting

In the simulations, time is subdivided into *slots* of 1 second. In each slot, nodes can connect and exchange data: connection establishment is always completed within a slot, while data exchange can involve one or more slots. Simulations were run for a duration of 172800 time slots, i.e., 48 h, which we identified to be sufficient to observe the system out of any transient. Besides, we have carried out 20 simulation runs per specific configuration in Paderborn and 15 for Rome scenarios, which was enough to gather sufficient statistics.

As for the simulation settings, the main parameters used are listed in Table 2. We differentiate among Paderborn and Rome scenarios due to the different characteristics and dimensions of each case.

6.2.1. Set up

In the simulations, the radius of the RZs has been set to 200 m for Paderborn and 1000 m in Rome. A randomly chosen node within an RZ injects a new piece of content every 1000 slots, and the initial number of injected contents in the first slot is two, i.e., one content per RZ. The max amount of stored contents per node is given by the storage capacity described by the *memory limit* parameter, which here we consider unbounded, for simplicity. We assume that when two nodes are in contact, the channel rate is constant over time and equal to 10 Mbit/s.

For pedestrian mobility, 50 nodes are uniformly distributed across allowed dwelling points, and when they move they follow the shortest path allowed by roads and

squares in the map. The two selected RZs have quite different characteristics: the RZ on the left of the map only includes dwelling points spaced more than 30 m, which is the transmission range, while the RZ on the right includes a cluster of attractor points within transmission range. This will enforce differences in the metrics observed in the two RZs, because dwelling points within transmission range behave like a single dwelling point with the sum of the respective mobile users.

For Rome, 199 nodes follow the trajectories provided by the taxi cabs deployed across the city from which not all nodes might be present in the scenario at initial time, neither during the whole simulation. Note that, as a consequence, nodes will remain within the area for limited periods of time.

6.2.2. Precise

Experiments with Precise are structured to study three fundamental features. The first feature is **node mobility**, defining speed and pause times. For Paderborn scenario, we consider *Businessman*’s, *Clerk*’s, and *Student*’s mobility scenarios, according to mobility dynamicity. In all scenarios, nodes are split in two equally sized groups: each group has a preferred RZ, which is visited more often (see Table 2). *Business*’s mobility assumes rather long pause time of 2–5 h, and RZs are visited with relatively low probabilities (0.2 and 0.1). *Clerk*’s mobility increases pause periods to 3–8 h and RZ visiting probabilities are quite high (0.7 and 0.2), leading to low probability to select a dwelling point outside the RZs. *Student*’s mobility sees nodes moving often, visiting for only 15–30 min the two RZs with mildly high probabilities of 0.4 and 0.3, respectively.

For Rome scenario, the pause time reported in Table 2 corresponds to the average pause time computed over the first 2 days of traces. A pause is described as every interval of time longer than 30 s where taxi cabs report the same position. We have also directly obtained from traces the RZ visiting probabilities for Rome by computing the average time spent by taxis at every RZ and outside.

The second feature concerns the **content exchange scheme**, which is mainly affected by T_e , i.e., the interval during which nodes can keep data stored and also exchange content when they are outside the RZs. We studied a wide range of configurations for this parameter, between 0–800 s.

The third feature concerns the **data exchange area**, i.e., the restrictions on where data exchange can take place. We label with ‘in’ the scenarios in which exchanges are allowed only inside RZs and with ‘out’ those in which exchanges are allowed everywhere.

6.2.3. Benchmarks

Besides, we consider two extreme benchmark cases. In the first, labeled as ‘Restricted Only In RZ’, nodes cannot carry any piece of content outside the RZ. This is the typical approach used in other works dealing with content dissemination in RZs. In a second benchmark, we use a scheme that allows to keep and exchange contents

limitless outside RZs, labeled as ‘Epidemic $T_e = \infty$ ’ in plots. This represents a typical uncontrolled epidemic diffusion scenario.

In addition, we have implemented PIS, a content dissemination protocol that bases its content dissemination decisions on three different social dimensions called similarities. First, PIS takes into account the physical proximity between nodes. Each node builds a so called Ego Matrix to keep track of, not only its own previously met contacts, but also the contacts of the peer devices that it encounters. Each node’s Ego matrix is updated at every time slot with the information of the neighboring nodes. Then, the physical proximity similarity is computed using the information stored in the matrix for the next n time slots, and we use $n = 1$. Second, PIS considers users interests to decide whether they will possibly meet the destination node and therefore, be selected as a next hop candidate. We have randomly generated lists of interests for PIS with the dwelling points selected in the application scenarios, which is the same as we do for Precise. Third, PIS spots friendship between nodes according to the number of unicast messages that they have exchanged in the past. Social relationships for PIS are also randomly generated since we assume that the only piece of information that a system will be able to retrieve is the mobility of nodes, thus avoiding extra overhead.

We run PIS routing protocol in our scenarios with the previous settings and, bearing in mind that we consider all nodes as potential destinations, we also inject a high number of content copies. When computing the exchange decision parameter $simPIS$, as explained in [32], a constant value of γ needs to be selected in order to assist or constrain the dissemination of contents. This value has an impact on the system when the number of content copies is low. Given that, when γ is high it allows for a rapid content dissemination, thus PIS takes the risk of sharing all pieces of content earlier than the destination node is reached according to the following equation:

$$simPIS + \gamma > 0. \quad (11)$$

To prove that, we have run experiments for the two extreme cases of γ : 0.2 and 0.8.

We compare our results with these three benchmark cases, whose results are reported within each figure.

6.3. Sensitivity analysis

We have carried out a parameter study of the probabilities involved in the decision making process explained in Section 4.2. Our goal is to reach a fair trade-off between the content availability and the overhead generated due to the number of connections established. This trade-off can be regulated by adapting the probability with which a content can be exchanged given the requirements of our predictive scheme.

We performed a set of one-day simulations for the group of student mobility nodes on Paderborn scenario, with

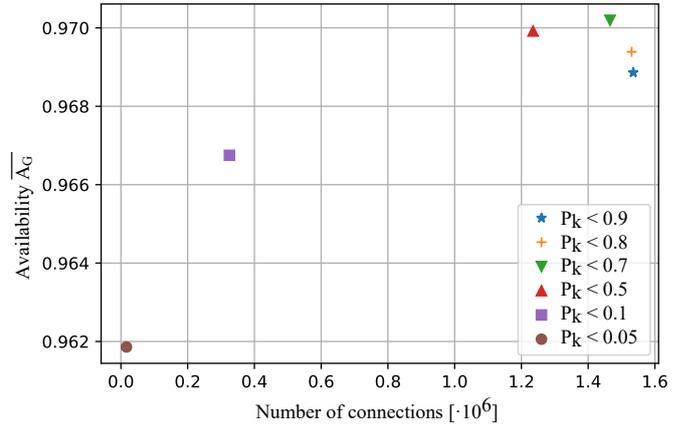


Figure 3: Average content availability and standard deviation versus number of connections for different values of the content exchange probability.

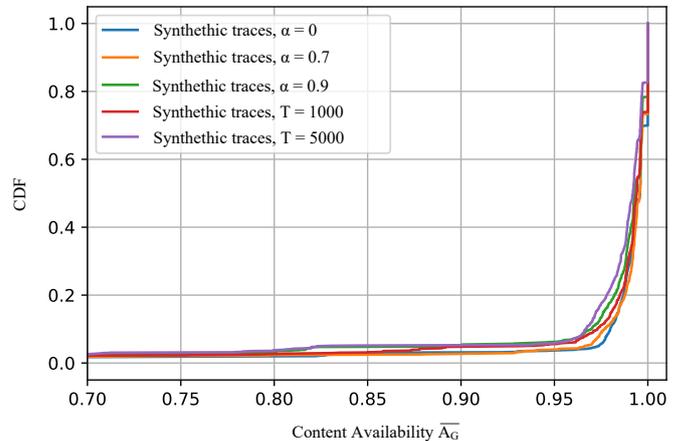


Figure 4: eCDF of content availability over one day experiment in Paderborn.

$T_e = 800s$ and allowing nodes to exchange contents also outside RZs. We test different probability values as shown in Fig. 3 and observe that using a very restrictive condition, i.e., forcing $P_k < 0.05$, gives a desirable reduction in the number of connections in exchange of some decrease in the content availability value. However, we aim at comparing Precise with other approaches such as Epidemic or PIS routing protocol, which are proven to be very competitive in terms of content availability [32]. We notice a sharp flop of performance in the availability from $P_k < 0.1$ to $P_k < 0.05$. For that reason, we observe that configurations closer to $P_k < 0.1$ are more beneficial because they increase content availability while still getting benefit from the overhead compared to the mentioned approaches. In the following results, we fixed the constraint to $P_k < 0.1$

In Fig. 3, we can see the significant impact of the parameter chosen for the probability. This is one of the reasons why, instead of using a fixed value for the probability condition, we have decided to use Algorithm 2, in which P_k is adjusted in steps, based on a quantity F_k which is the

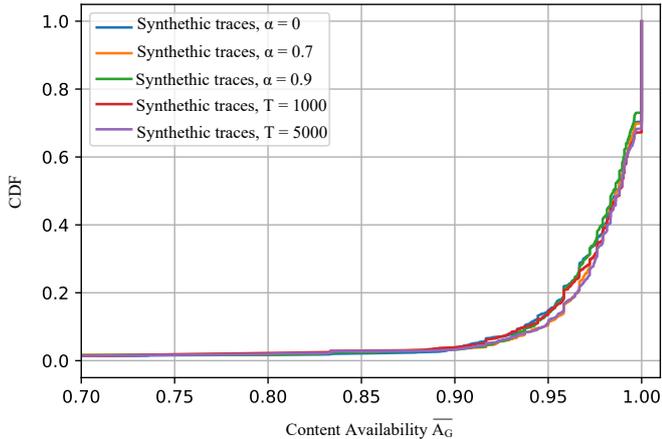


Figure 5: eCDF of content availability over one day experiment in Rome.

output of an AR filter. We based the probability on the average number of contents that two nodes have to exchange at every encounter as presented in Eq. (2).

Content availability results for adaptive values of α are depicted in Figs. 4 and 5 for synthetic traces and real traces from the literature [37], respectively. Note that, in both cases we run again a single one-day simulation which is a good representative example for the parameter study. From the figures, we can observe that the variations in content availability are almost negligible. In the case of synthetic traces in Paderborn scenario, the best results are grasped when α has a fixed value of 0.7. In the case of real traces in Rome, the content availability increase is sensed when adaptive α is computed from a large value of $T = 5000$, which means larger memory, corresponding to about 7 h^2 .

Surprisingly, Figs. 4 and 5 do not show a significant improvement when no memory is taken into account, i.e., $\alpha = 0$ which means that all connections are allowed at any time. Therefore, increasing the network load, as shown in Section 6.6, is not essential to get content availability values close to 1.

This way, we have verified that the value of T , or even fixing alpha, does not bring in notable differences, so we picked a round value of $T = 1000$, which is enough to keep in the memory of the filter whatever happens while a node is outside the RZs for about T_e .

6.4. Content Availability

One of the main performance parameters of the system is the mean content availability \bar{A}_G inside RZs. In each of the tested configurations, we have obtained \bar{A}_G by averaging per-content availabilities across all contents (i.e., we use $G = D(t)$). Fig. 6 shows this metric for each

²With an auto-regressive filter like the one used in this work, the importance of a sample fades exponentially with decay time T , therefore it becomes practically negligible after $5T$.

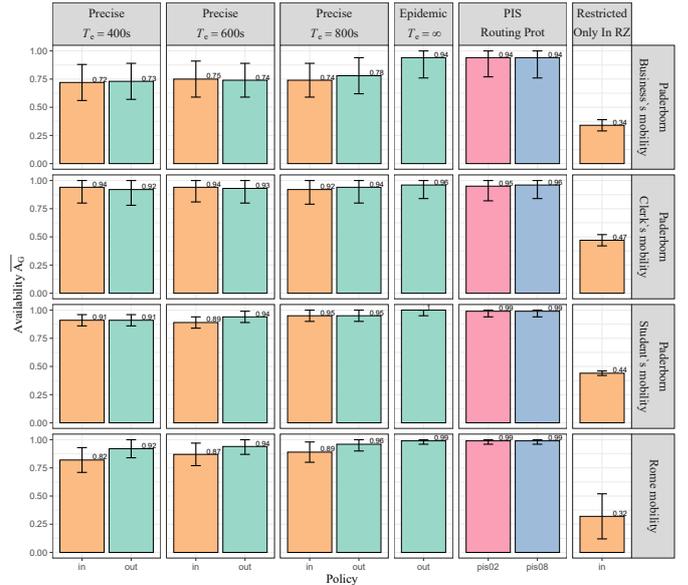


Figure 6: Time-average content availability (cf. Eq. (6)–Eq. (7)) for all forwarding and storage configurations, computed over the entire duration of the experiments and for all contents ($G = D(12\text{h})$ in Eq. (7)). Plotted is the mean together with the standard deviation.

configuration. The figure shows also the standard deviation observed in the simulations. The computed confidence intervals are small and hence we do not report them in the figure. Their value indicates that, with probability 95%, the actual average is within $\pm 0.19\%$ of the estimated average reported in the plot. As it can be seen, if data exchange outside the RZs is permitted, the content availability increases in comparison to the traditional paradigm, where content exchange is only allowed for nodes inside RZs. When it is possible to carry a piece of content outside the RZ, performance improves drastically: in fact, results for the case ‘Restricted Only In RZ,’ in which nodes clear their storage as they leave the RZ, are the worst. Moreover, configuring our proposed scheme so to keep contents stored on nodes for a limited amount of time performs comparably well with respect to the uncontrolled dissemination case (‘Epidemic $T_e = \infty$ ’ in the figures). These results provide valuable information on how to manage resources in the network.

Results show a similar trend over all mobility cases. However, if we compare results obtained from different mobility patterns in Paderborn, Clerk’s and Student’s mobility cases look similar while Businessman’s mobility depicts lower availability numbers. In the Student’s case, nodes move with the lowest pause time at hotspots and, therefore, the higher number of nodes encounters explains the rapid pervasion of contents. We can clearly appreciate this fact in Fig. 7, where we plot the mean content availability across all contents, $A(t)$, as it evolves over time³. Student’s mobility

³Note that, in Fig. 7, time is relative to content generation epoch.

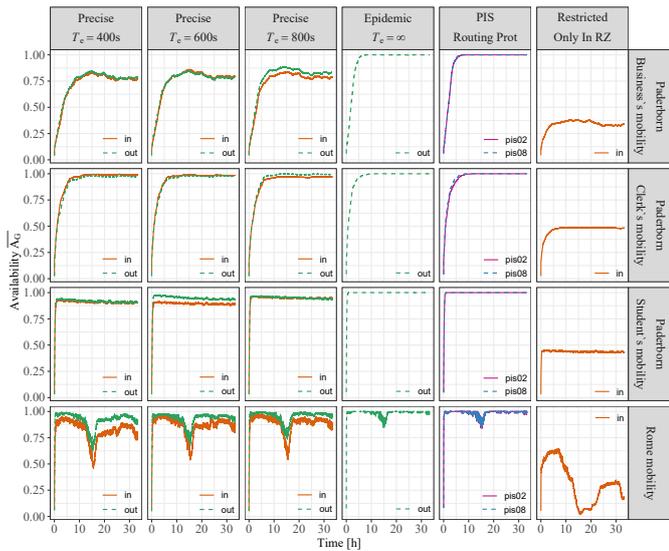


Figure 7: Evolution over time of the mean content availability (cf. Eq. (6)–Eq. (7)) observed under different forwarding and storage configurations.

subplots illustrate a quick content spreading among nodes and very small variation across different contents behavior, that is why standard deviation in Fig. 6 is smaller than under Businessman’s and Clerk’s mobility cases. Here, when contents are first generated, it takes around one hour only to hit the highest availability values compared to Clerk’s mobility patterns that reach its highest point later on but keeps it in steady state. Businessman’s mobility subplots depict a rising curve that, in most of the cases, does not reach either availability values as high as nodes under the Clerk’s or Student’s mobility over the entire run.

Finally, the simulations executed over the traces obtained from taxi cabs in Rome depict lower values of content availability due to higher nodes speed and the frequent disappearance of nodes from the considered area. On average, nodes are present in the scenario for 11 h, and once they leave the scenario the content stored in their local buffer is erased. In view of these results, we can perceive the influence of the nodes mobility pattern over the content availability. Fig. 8 shows the evolution over time of the number of users present at each RZ for each scenario. Comparing the results with the number of nodes in the scenario over time, and specifically for each RZ in Fig. 8d, we can appreciate that the rising trend in content availability corresponds to higher visits to the RZs, however we are clearly able to overcome the decreasing trend by applying our proposed scheme. Nodes leaving the RZs in a short period of time but remaining in the scenario are still a critical asset to spread the collected contents until *time elapsed* value is up.

This is to be able to homogeneously compute the statistical mean for a set of contents generated at different instants.

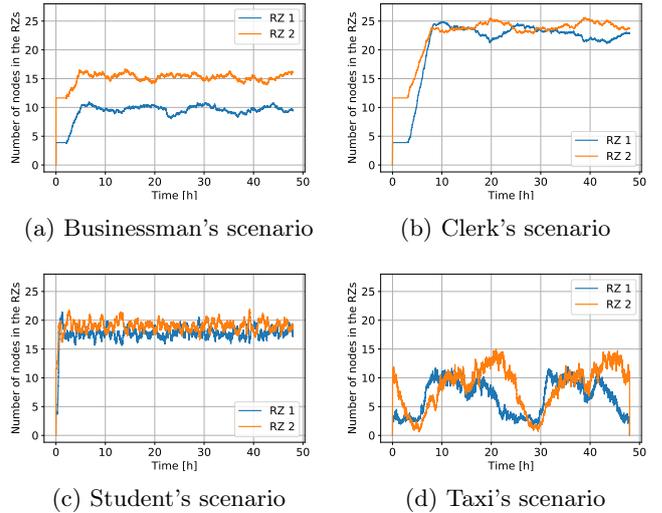


Figure 8: Evolution over time of the number of users present at each RZ for each scenario.

From the results, it is clear that availability increases dramatically when nodes are allowed to keep the contents while outside RZs. Likewise, it is clear that there is an extra benefit due to allowing nodes to connect and exchange data outside RZs, at least under the mobility models explored. More importantly, we demonstrate that there is no need to allow nodes to store and exchange contents for long periods of time. A fair performance can be reached by constraining the time elapsed parameter T_e according to the scenario settings while maintaining comparable values of content availability to those reached by epidemic dissemination techniques.

In all cases, we have observed high availability values. Although we can notice that Precise achieves slightly lower availability with respect to Epidemic and PIS, the availability is high enough to guarantee content persistence over long intervals. Indeed, we will see later in this section that content lifetime is not affected and losses are negligible. The only exception is the case in which we restrict content exchanges within RZs in the Rome scenario, where we can see that availability approaches zero dangerously when there are only a few cabs in the region. This confirms that allowing some content exchange outside RZs is key to obtain good performance.

6.5. Storage usage

Another significant system metric is the amount of storage used. As shown in Fig. 9 (we plot again the mean together with the standard deviation and we do not report confidence intervals in the figure since value indicates that, with probability 95%, the actual average is within $\pm 2.71\%$ of the estimated average reported in the plot.), our predictive scheme performs better than the benchmark that restricts storage and data exchanges to RZs, and is comparable to the uncontrolled dissemination scheme in

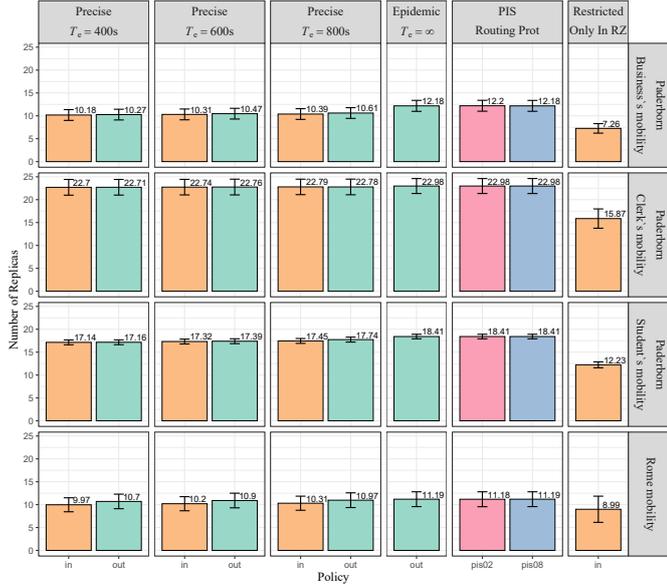


Figure 9: Average number of replicas of a content, computed across both RZs and time-averaged over the entire simulation. Plotted is the mean together with the standard deviation (taken with respect to the time-averaged number of replicas of each content).

terms of available replicas. Note that, with the high values of content availability achieved by our predictive scheme, the number of replicas should be comparable to the number of nodes inside each RZ (see Fig. 8). With the parameters chosen for the three mobility scenarios that we have simulated in Paderborn, 30%, 70% and 90% of nodes will lay within RZs, for Businessman's, Student's and Clerk's mobility cases, respectively. Therefore, with the availability values reported for our scheme in Fig. 6, we should expect to observe about 13 to 15 replicas for the Businessman's case, 19 replicas for Student's case, and 23 for the Clerk's case. For Rome, we expect to have between 9 and 11 replicas inside the RZs according to the average number of nodes visiting the selected RZs. However, Fig. 9 shows lower values because it accounts for the delay incurred for spreading newly injected contents. More in detail, Fig. 10 shows how the number of contents stored in each node tends to increase as time passes. Abrupt changes visible in the figure are due to the periodic injection of fresh contents.

In the figure, we can observe how fast the storage capacity of nodes fills up, according to each mobility pattern. It draws our attention that the Businessman's case presents a slower increase in memory usage compared to Clerks and Students nodes. This is due to the fact that, in Businessman's case, nodes do not visit RZs as often as in Clerk's case, therefore, they miss opportunities to retrieve the data; in the Businessman's case, nodes neither have as many new peers encounters as in the Students mobility case, due to higher pause times. However, if we analyze again Fig. 6, we see that Businessman's mobility settings provide fair avail-

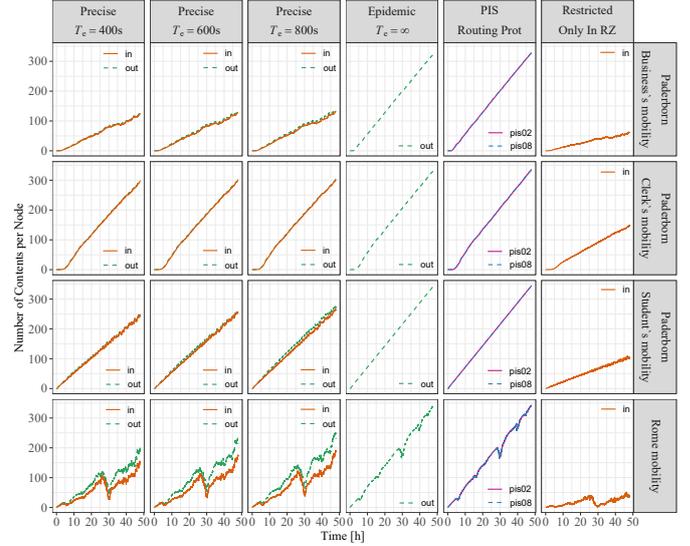


Figure 10: Average number of contents stored at a node (this quantity corresponds to $R(t) \cdot D(t)$). The final amount of injected contents in one simulation run is 345.

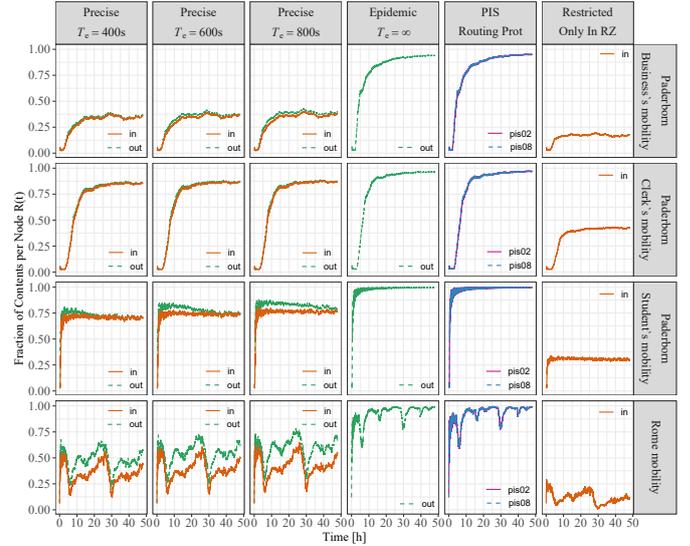


Figure 11: $R(t)$, mean fraction of contents possessed by a node (cf. Eq. (9)).

ability compared to Clerk's and Student's mobility cases, while occupying 50% of their storage and experiencing less content exchanges, as explained later in Section 6.6. Fig. 11 further shows the normalized number of per-node stored content, $R(t)$, i.e., the fraction computed with respect to the number of injected contents. The computed confidence intervals are small and hence we do not report them in the figure. Their value indicates that, with probability 95%, the actual average is within $\pm 0.22\%$ of the estimated average for the whole runs. Content injection epochs are clearly

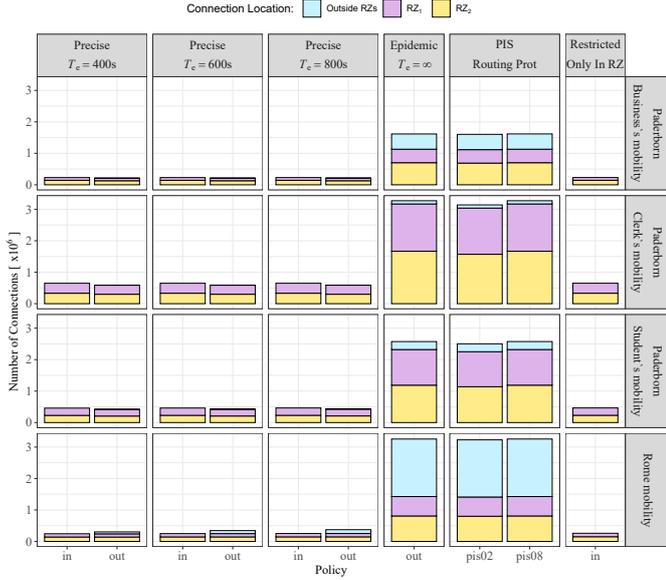


Figure 12: Total number of connections per simulation run in millions (average over 5 runs). Shown is a stacked plot of connections in each RZs and outside.

visible in this figure. The figure shows that the system tends to quickly spread fresh injected contents, though this process is strongly correlated with the speed and dynamicity of nodes. Indeed, it is clear that in Student’s mobility scenario contents are distributed more rapidly. Despite the fact that the results for Businessman’s and Clerk’s mobility traces present a slower increase, Clerk’s mobility figures illustrate a more agile dissemination of contents among nodes due to higher probability of visiting RZs.

Again in Fig. 10 and Fig. 11, we can notice the evident impact of real mobility traces in the average number of contents stored at the nodes. In Rome, even though the RZs are highly visited areas, the speed of the cabs and the low pause time in both zones introduce a considerable decrease in the storage usage of the nodes, due to faster content acquisition and imminent loss, as we could earlier notice when analyzing Businessman’s mobility case.

Still, when connections are allowed outside the RZs, we can observe a slightly higher usage of the system storage, although less than in the case in which dissemination operations are unbounded ($T_e = \infty$). However, while gaining some content availability from adding connections outside RZs, the overall storage capacity is not significantly affected.

6.6. Number of connections

Since any forwarding policy will introduce constraints in terms of data sharing attempts, the number of connections will inevitably affect the network performance and, therefore, also the content availability. Fig. 12 reports the average number of connections observed during simulations. Connections are divided into three groups depending on

Table 3: Comparison of the total number of connections for Epidemic and PIS schemes with and without the restriction of exchanging contents only inside RZs

Configuration	Number of connections with restriction	Number of connections without restriction
Epidemic Business	1.1684×10^6	1.6154×10^6
PIS08 Business	1.1693×10^6	1.6167×10^6
PIS02 Business	1.1568×10^6	1.6031×10^6
Epidemic Clerk	3.1815×10^6	3.272×10^6
PIS08 Clerk	3.1819×10^6	3.2722×10^6
PIS02 Clerk	3.0559×10^6	3.1399×10^6
Epidemic Students	2.2924×10^6	2.5739×10^6
PIS08 Students	2.2932×10^6	2.5746×10^6
PIS02 Students	2.2331×10^6	2.5028×10^6
Epidemic Rome	0.9048×10^6	3.2551×10^6
PIS08 Rome	0.9058×10^6	3.2560×10^6
PIS02 Rome	0.8877×10^6	3.2289×10^6

where they take place: inside RZ_1 , inside RZ_2 , or outside RZs. Under the Businessman’s mobility, the number of connections is lower than in the other cases for Paderborn city, which is due the lower probabilities of visiting the RZs. The same happens in Rome’s subplots, despite the fact that the RZs defined in this scenario are the most frequently visited areas, they are still not as visited as in Paderborn scenario plus nodes do not remain there for long periods.

In all considered cases, even if negligible in the plots, the number of connections outside RZs increases with T_e , and it is more pronounced in Businessman’s and Rome’s mobility scenarios, as we can clearly see when T_e is unbounded, in which case all nodes will always attempt to establish a connection when they meet, independently from their mobility characteristics. Therefore, since in the Businessman’s and Rome’s scenario nodes tend to travel to the defined RZs with lower probability, having more opportunities for complete new encounters, they also end up using the most amount of connections outside RZs, even when $T_e \rightarrow \infty$. Note that the number of connections outside RZs is quite limited for finite values of T_e , which means that our scheme does not require much use of network capacity when nodes are outside RZs. Therefore, our predictive scheme performs comparably to unbounded epidemic diffusion schemes for what concerns dissemination of contents, although they require much less network resources. In some cases, the number of connections in the two RZs is not symmetric. Indeed, the RZ in which we observe less connections is the one in which dwelling points are spaced apart, out of transmission range. When the average number of nodes in each RZ is low, they end up being, with high probability, far apart to establish connections, which results in less connections than in the other RZ. Instead, under mobility scenarios where the number of nodes in the RZs is higher, various nodes will be going towards the same dwelling point and thus they will be able to connect.

Besides, to assess the importance of bounding content

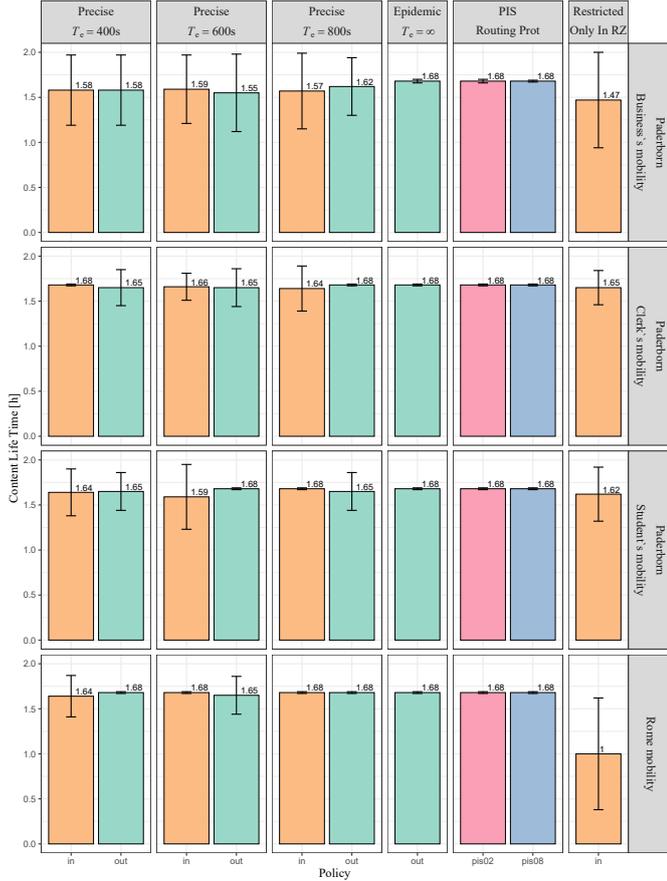


Figure 13: Average lifetime of contents generated between $t \approx 14-15$ h. Plotted is the mean together with the standard deviation.

exchanges within or nearby RZs, we have tested the behavior of Epidemic and PIS in case they are forbidden to operate outside RZs. Table 3 compares the number of connections with Epidemic and PIS (with $\gamma = 0.2$ and $\gamma = 0.8$) with and without such restriction. In the table, we can appreciate that, with the restriction, the number of connections experienced decreases substantially only in case of highly dynamic mobility—e.g., in the taxi cab scenario—, although in all cases it remains much higher than what observed for Precise (cf. Fig. 12).

6.7. Data lifetime, losses and delivery delay

We now compare our predictive scheme to the benchmarks in terms of their ability to keep contents alive without the support of an infrastructure. Fig. 13 shows the average lifetime of contents that were generated between $t \approx 14-15$ h, with a residual simulation duration of ≈ 33 h. The computed confidence intervals are small and hence we do not report them in the figure. Their value indicates that, with probability 95%, the actual average is within $\pm 0.25\%$ of the estimated average reported in the plot.

We observe that most of the contents are kept alive a bit longer than 1.5 h (for about 95-100 minutes), for all

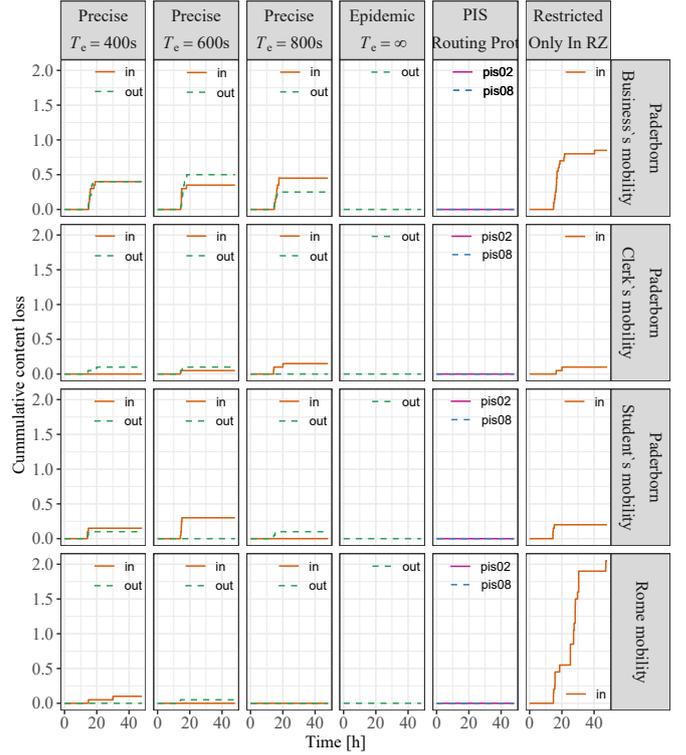


Figure 14: Average content pieces lost over time. Average losses are much less than a single piece of content in 6.5 simulated hours (in most of the runs, there are no losses at all).

settings and schemes. However, the baseline ‘Restricted Only In RZ’ scheme shows a smaller average lifetime, up to only ≈ 45 min in Rome scenario, and more variability across contents and simulations. This shows that, even with only 50 nodes in the case of Paderborn, our predictive forwarding scheme, along with realistic mobility patterns, is as resilient as pure unconstrained dissemination schemes, although it requires less use of communication resources.

Losses are very infrequent, as shown in Fig. 14. Indeed, the number of contents disappearing is less than 1 on average, under any of the tested configurations. The computed confidence intervals are small and hence we do not report them in the figure. Their value indicates that, with probability 95%, the actual average is within $\pm 0.03\%$ of the estimated average reported in the plot.

It is interesting that most of the lost contents disappear at the very beginning of their lifetime, when only one node possesses each specific piece of content injected. Nevertheless, contents that manage to survive to that transient period remain alive until the end of the simulation, thanks to a fast dissemination. Especially, in the cases of Clerk’s mobility with more nodes gathering for longer around the same areas and Student’s mobility with shorter pause times and, consequently, higher number of encounters. Additionally, with Businessman’s and Rome’s mobility, we notice a significant reduction of content loss when we allow ex-

changes outside RZs.

In most cases, and especially in Rome scenario, our predictive scheme suffers much less losses than the ‘Restricted Only In RZ’ benchmark, although not as good as for the uncontrolled case with $T_e \rightarrow \infty$. We can conclude that the introduction of the T_e parameter in our predictive scheme plays an interesting role: by varying its value, it is possible to trade-off reliability (losses) for costs (network resources) with only a limited impact on the availability of contents in RZs.

Finally, we have also measured the content delivery delay in Paderborn and Rome scenarios. Note that, the classic delivery delay metric cannot be applied for our scheme since the behavior of content exchanges is different for each case. I.e., in our predictive scheme, nodes drop the content when they are out of the RZs for too long ($> T_e$) and that does not happen in epidemics-like configurations. In epidemic (or PIS), nodes never drop the content so only the first time they get the content is taken into account. However, in our scheme, nodes can get the content soon for the first time and drop it after a while if they are out of RZs. Then, they can retrieve the content again if the conditions allow for it. Besides, the contents are needed only inside the RZs, so what matters is that they are received by the time a node enter an RZ or shortly after. Given these circumstances, we measure the delivery time as the time elapsed since a node enters an RZ. With this metric, in Fig. 15 we see that cases where content exchanges are allowed outside the RZs, the content delivery delay decreases since nodes that enter the RZs already possessing the content report a delay of 0s. Despite the fact that epidemic and PIS configurations show the lowest content delivery delay values, Precise presents comparable results with the smallest difference in the order of milliseconds for the most dynamic case (Rome). It also draws our attention the higher values obtained for Clerk’s mobility. As stated in Section 6.2.1, the RZs defined for Paderborn scenario contain some attractor points towards the nodes move with higher probability and, in this case, remain for longer. This will imply that, even though nodes eventually get the content because they remain during long periods inside the RZs, they will need more time to find someone to exchange content with. In one of the RZs, the distance between some attractor points is larger than the transmission range. This means that if nodes fall into separate points, they will not be able to retrieve the content until they move again after a long pause.

7. Conclusions

We presented Precise, a series of data forwarding, storing and decision making schemes to benefit the infrastructureless content dissemination process in opportunistic networks, particularly focusing on D2D data exchange between hotspots. Our predictive scheme leverages node mobility patterns to make effective forwarding decisions

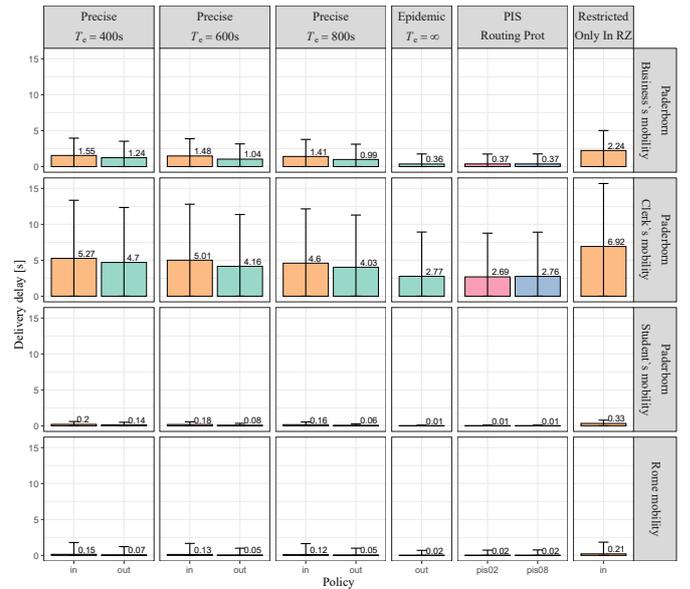


Figure 15: Average delivery delay of contents. Plotted is the mean together with the standard deviation.

and efficiently use network resources while maintaining fair content availability values.

Precise specifies how a node adapts its probability to decide to exchange contents when it meets another node, so as to speedup content diffusion when encounters are rare, or save resources when contacts are frequent. It also uses the knowledge of pedestrians and vehicles’ position and mobility pattern to carry out meaningful connections between nodes and improves the main performance indicators of our system especially in the selected RZs. Since node density and device limitations impair dissemination and system scalability, Precise encourages only the exchange of valuable data between potential peers according to their predicted movement, encounters and content expiration time. In such a way, we dramatically reduce the number of connections by 65-92% and the use of on-device storage by 50%, consequently avoiding network congestion and memory shortage. Indeed, our results show that existing schemes use unnecessary high amount of storage and communication resources, while Precise can easily provide the same quality of service (achieving real-time messaging delay and very high availability of contents) and the same content lifetime at a much lower cost.

References

- [1] M. Conti, M. Kumar, Opportunities in Opportunistic Computing, IEEE Computer 43 (2010) 42–50. doi:10.1109/MC.2010.19.
- [2] J. Dede, A. Förster, E. Hernández-Orallo, J. Herrera-Tapia, K. K. Kuladinithi, V. Kuppasamy, P. Manzoni, A. b. Muslim, A. Udugama, Z. Vatandas, Simulating opportunistic networks: Survey and future directions, IEEE Communications Surveys & Tutorials, 2 (2017) 1547 – 1573. doi:10.1109/COMST.2017.2782182.
- [3] G. Gorbil, No way out: Emergency evacuation with no internet access, in: 13th IEEE International Conference on Pervasive Computing and Communication (PerCom 2015), Workshop

- on Pervasive Networks for Emergency Management (PerNEM 2015), IEEE, St. Louis, MO, 2015. doi:10.1109/PERCOMW.2015.7134089.
- [4] F. Dressler, C. F. Chiasserini, F. H. P. Fitzek, H. Karl, R. Lo Cigno, A. Capone, C. E. Casetti, F. Malandrino, V. Mancuso, F. Klingler, G. A. Rizzo, V-Edge: Virtual Edge Computing as an Enabler for Novel Microservices and Cooperative Computing, cs.NI 2106.10063, arXiv, 2021.
 - [5] M. Grossglauser, D. N. C. Tse, Mobility increases the capacity of ad hoc wireless networks, *IEEE/ACM Transactions on Networking (TON)* 10 (2002) 477–486. doi:10.1109/TNET.2002.801403.
 - [6] T. Spyropoulos, K. Psounis, C. S. Raghavendra, Spray and wait: an efficient routing scheme for intermittently connected mobile networks, in: *ACM SIGCOMM Workshop on Delay-Tolerant Networking (WDTN 2005)*, ACM, Philadelphia, PA, 2005. doi:10.1145/1080139.1080143.
 - [7] A. Vahdat, D. Becker, Epidemic routing for partially connected ad hoc networks, Technical Report CS-200006, Duke University, 2000.
 - [8] A. Bujari, A Survey of Opportunistic Data Gathering and Dissemination Techniques, in: *IEEE International Conference on Computer Communication Networks (ICCCN 2012)*, IEEE, Munich, Germany, 2012. doi:10.1109/ICCCN.2012.6289225.
 - [9] R. Ghebleh, A comparative classification of information dissemination approaches in vehicular ad hoc networks from distinctive viewpoints: A survey, *Elsevier Computer Networks (COMNET)* 131 (2018) 15–37. doi:10.1016/j.comnet.2017.12.003.
 - [10] J. Wang, X. Kong, F. Xia, L. Sun, Urban Human Mobility, *ACM SIGKDD Explorations Newsletter* 21 (2019) 1–19. doi:10.1145/3331651.3331653.
 - [11] J. Zhang, W. Wang, F. Xia, Y.-R. Lin, H. Tong, Data-Driven Computational Social Science: A Survey, *Big Data Research* 21 (2020) 100145. doi:10.1016/j.bdr.2020.100145.
 - [12] C. Boldrini, M. Conti, J. Jacopini, A. Passarella, HiBOP: a History Based Routing Protocol for Opportunistic Networks, in: *8th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM 2007)*, IEEE, Helsinki, Finland, 2007. doi:10.1109/WOWMOM.2007.4351716.
 - [13] P. Hui, J. Crowcroft, E. Yoneki, Bubble rap: social-based forwarding in delay tolerant networks, in: *9th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2008)*, ACM, Hong Kong, China, 2008, pp. 241–250. doi:10.1145/1374618.1374652.
 - [14] Y. Zhao, W. Song, Survey on Social-Aware Data Dissemination Over Mobile Wireless Networks, *IEEE Access* 5 (2017) 6049–6059. doi:10.1109/ACCESS.2017.2693689.
 - [15] G. S. Pannu, F. Hagenauer, T. Higuchi, O. Altintas, F. Dressler, Keeping Data Alive: Communication Across Vehicular Micro Clouds, in: *20th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM 2019)*, IEEE, Washington, D.C., 2019. doi:10.1109/WOWMOM.2019.8792973.
 - [16] G. A. Rizzo, N. Pérez Palma, M. Ajmone Marsan, V. Mancuso, A Walk Down Memory Lane: On Storage Capacity in Opportunistic Content Sharing Systems, in: *21st IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM 2020)*, IEEE, Virtual Conference, 2020. doi:10.1109/wowmom49955.2020.00022.
 - [17] E. Hernández-Orallo, C. Borrego, P. Manzoni, J. M. Marquez-Barja, J. C. Cano, C. T. Calafate, Optimising data diffusion while reducing local resources consumption in Opportunistic Mobile Crowdsensing, *Pervasive and Mobile Computing* 67 (2020) 101201. doi:10.1016/j.pmcj.2020.101201.
 - [18] L. Chancay-García, E. Hernández-Orallo, P. Manzoni, A. M. Vegni, V. Loscrí, J.-C. Cano, C. T. Calafate, Optimising message broadcasting in opportunistic networks, *Elsevier Computer Communications* (2020) 162–178.
 - [19] S. K. Dhurandher, D. K. Sharma, I. Woungang, S. Bhati, HBPR: History Based Prediction for Routing in Infrastructure-less Opportunistic Networks, in: *27th IEEE International Conference on Advanced Information Networking and Applications (AINA 2013)*, IEEE, Barcelona, Spain, 2013. doi:10.1109/AINA.2013.105.
 - [20] A. Lindgren, A. Doria, O. Schelén, Probabilistic routing in intermittently connected networks, in: *4th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2003)*, ACM, Annapolis, MD, 2003. doi:10.1145/961268.961272.
 - [21] C. L. Barrett, S. J. Eidenbenz, L. Kroc, Parametric Probabilistic Sensor Network Routing, in: *9th ACM International Conference on Mobile Computing and Networking (MobiCom 2003)*, San Diego, CA, 2003.
 - [22] B. Burns, O. Brock, B. N. Levine, MV Routing and Capacity Building in Disruption Tolerant Networks, in: *24th IEEE Conference on Computer Communications (INFOCOM 2005)*, Miami, FL, 2005. doi:10.1109/INFCOM.2005.1497909.
 - [23] Y. Liu, A. E. Bashar, F. Li, Y. Wang, Multi-copy data dissemination with probabilistic delay constraint in mobile opportunistic device-to-device networks, in: *17th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM 2016)*, IEEE, Coimbra, Portugal, 2016. doi:10.1109/WoWMoM.2016.7523548.
 - [24] R. Yamamoto, A. Kashima, T. Yamazaki, Y. Tanaka, Adaptive Contents Dissemination Method for Floating Contents, in: *90th IEEE Vehicular Technology Conference (VTC 2019-Fall)*, IEEE, Honolulu, HI, 2019. doi:10.1109/vtcfall.2019.8891485.
 - [25] N. Pérez Palma, V. Mancuso, M. Ajmone Marsan, Infrastructureless Pervasive Information Sharing with COTS Devices and Software, in: *19th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM 2018)*, IEEE, Chania, Greece, 2018, pp. 1–9. doi:10.1109/WoWMoM.2018.8449733.
 - [26] G. A. Rizzo, V. Mancuso, S. Ali, M. Ajmone Marsan, Stop and forward: Opportunistic local information sharing under walking mobility, *Elsevier Ad Hoc Networks* 78 (2018) 54–72. doi:10.1016/j.adhoc.2018.05.011.
 - [27] C. Boldrini, M. Conti, A. Passarella, ContentPlace: social-aware data dissemination in opportunistic networks, in: *11th ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM 2008)*, ACM, Vancouver, Canada, 2008. doi:10.1145/1454503.1454541.
 - [28] B. Ying, K. Xu, A. Nayak, Fair and Social-Aware Message Forwarding Method in Opportunistic Social Networks, *IEEE Communications Letters* 23 (2019) 720–723. doi:10.1109/lcomm.2019.2900892.
 - [29] A. Rahim, T. Qiu, Z. Ning, J. Wang, N. Ullah, A. Tolba, F. Xia, Social acquaintance based routing in Vehicular Social Networks, *Future Generation Computer Systems* 93 (2019) 751–760. doi:10.1016/j.future.2017.07.059.
 - [30] N. Ullah, X. Kong, Z. Ning, A. Tolba, M. Alrashoud, F. Xia, Emergency warning messages dissemination in vehicular social networks: A trust based scheme, *Vehicular Communications* 22 (2020) 100199. doi:10.1016/j.vehcom.2019.100199.
 - [31] A. M. Vegni, C. Souza, V. Loscrí, E. Hernández-Orallo, P. Manzoni, Data transmissions using hub nodes in vehicular social networks, *IEEE Transactions on Mobile Computing (TMC)*, 7 (2020) 1570 – 1585. doi:10.1109/TMC.2019.2928803.
 - [32] F. Xia, L. Liu, B. Jedari, S. K. Das, PIS: A multi-dimensional routing protocol for socially-aware networking, *IEEE Transactions on Mobile Computing (TMC)* (2016). doi:10.1109/TMC.2016.2517649.
 - [33] A.-K. Pietiläinen, E. Oliver, J. LeBrun, G. Varghese, C. Diot, Mobiclique: middleware for mobile social networking, in: J. Crowcroft, B. Krishnamurthy (Eds.), *9th ACM SIGCOMM Conference on Internet Measurement (IMC 2009)*, Proceedings of the 2nd ACM workshop on Online social networks (WOSN '09), ACM, Barcelona, Spain, 2009, pp. 49–54. doi:10.1145/1592665.1592678.
 - [34] J. Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot, A. Chaintreau, *Crawdadd* dataset <https://crawdadd.org/cambridge/haggle/> (v. 2009-05-29), ??? URL: <https://crawdadd.org/cambridge/haggle/20090529>, . doi:10.15783/C70011.
 - [35] E. M. Daly, M. Haahr, Social network analysis for routing

- in disconnected delay-tolerant MANETs, in: 8th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2007), ACM, Montréal, Canada, 2007. doi:10.1145/1288107.1288113.
- [36] J. Li, Z. Ning, B. Jedari, F. Xia, I. Lee, A. Tolba, Geo-Social Distance-Based Data Dissemination for Socially Aware Networking, *IEEE Access* 4 (2016) 1444–1453. doi:10.1109/access.2016.2553698.
- [37] L. Bracciale, M. Bonola, P. Loreti, G. Bianchi, R. Amici, A. Rabuffi, CRAWDAD dataset roma/taxi (v. 2014-07-17), 2014. URL: <https://crawdad.org/roma/taxi/20140717/taxicabs>. doi:10.15783/C7QC7M, traceset: taxicabs.
- [38] M. Mordacchini, A. Passarella, M. Conti, Social Cognitive Heuristics for adaptive data dissemination in Opportunistic Networks, in: 16th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM 2015), IEEE, Boston, MA, 2015. doi:10.1109/WoWMoM.2015.7158159.
- [39] H. Khelifi, S. Luo, B. Nour, H. Mouncla, Y. Faheem, R. Hussain, A. Ksentini, Named Data Networking in Vehicular Ad Hoc Networks: State-of-the-Art and Challenges, *IEEE Communications Surveys & Tutorials* 22 (2020) 320–351. doi:10.1109/comst.2019.2894816.
- [40] A. Langley, A. Riddoch, A. Wilk, A. Vicente, C. Krasic, D. Zhang, F. Yang, F. Kouranov, I. Swett, J. Iyengar, J. Bailey, J. Dorfman, J. Roskind, J. Kulik, P. Westin, R. Tenneti, R. Shade, R. Hamilton, V. Vasiliev, W.-T. Chang, Z. Shi, The QUIC Transport Protocol: Design and Internet-Scale Deployment, in: *ACM SIGCOMM 2017*, ACM, Los Angeles, CA, 2017. doi:10.1145/3098822.3098842.
- [41] A. Keranen, J. Ott, T. Kämäräinen, The ONE simulator for DTN protocol evaluation, in: 2nd ACM/ICST International Conference on Simulation Tools and Techniques for Communications, Networks and Systems (SIMUTools 2009), ICST, Rome, Italy, 2009. doi:10.4108/ICST.SIMUTOOLS2009.5674.