Research Article

Delay Tolerant Networking over the Metropolitan Public Transportation

A. Bujari,[1] S. Gaito,[2] D. Maggiorini,[2] C. E. Palazzi,[1] and C. Quadri[2]

[1]*Department of Mathematics, University of Padua, Via Trieste 63, 35123 Padova, Italy*
[2]*Computer Science Department, University of Milan, Via Comelico 39, 20135 Milan, Italy*

Correspondence should be addressed to C. E. Palazzi; cpalazzi@math.unipd.it

We discuss MDTN: a delay tolerant application platform built on top of the Public Transportation System (PTS) and able to provide service access while exploiting opportunistic connectivity. Our solution adopts a carrier-based approach where buses act as data collectors for user requests requiring Internet access. Simulations based on real maps and PTS routes with state-of-the-art routing protocols demonstrate that MDTN represents a viable solution for elastic nonreal-time service delivery. Nevertheless, performance indexes of the considered routing policies show that there is no golden rule for optimal performance and a tailored routing strategy is required for each specific case.

1. Introduction

The mobile revolution has already begun and is gaining momentum. This revolution has boosted the demand for mobile-based versions of any information delivery system which is offered today by exploiting infrastructure-based connectivity [1, 2]. However, infrastructure-based service access might not always be available or convenient. Infrastructure networks like the Internet suffer from the so-called "*last mile problem*," that is, the lack of a total coverage area. On the other side, 3G/LTE and cellular networks in general are not the optimal next solution due to service coverage gaps, interoperability issues between carriers, and their pull (and cost-based) service model [3]. To this purpose, a lot of research efforts have been devoted to find new decentralized and infrastructure-less solutions [4–8].

Interestingly enough, the demand for new and innovative service provisioning techniques could be fulfilled by mobile devices themselves [9–11]. Their growing availability and pervasive coverage in our metropolitan areas could be exploited to interconnect users in an ad hoc fashion, extending infrastructure connectivity and avoiding service access costs.

Opportunistic Networks (OppNets [12]), as a way to involve multihop Delay/Disruption Tolerant Networking (DTN [13]), have proven to be a viable alternative to infrastructure-based connectivity. OppNets are not subject to charging and can be employed to extend infrastructure coverage where it is not available. In this domain, delay tolerant ad hoc networking techniques have gained a lot of popularity thanks to their potential of providing service connectivity by leveraging on node mobility to move data. Possible application scenarios span from social-local-based applications to classic e-mail, microblogging, advertisement, and games [14–17].

Solutions leveraging on human mobility have been extensively studied; however, the unpredictability of human movements poses severe challenges to network management. Instead, opportunistic solution deployed on top of a Public Transportation System (PTS) has proven more viable than its human counterpart. In this context, buses move along predetermined paths and follow an *a priori* known schedule, offering a quasideterministic encounter model [18]. Routing algorithms can thus be devised on reasonable assumptions and probabilistic predictions of encounters [19, 20]. Despite these positive features, even this carrier-based approach

has an important, and still unresolved, technical challenge: network scalability when considering a metropolitan-wide area with a growing number of lines and a potentially huge offered load. Since the PTS topology and size are tied to human and organizational factors, network delivery delay may ramp up with the covered area due to the increasing number of hops that each packet must traverse.

In this context, we analyze the performance of Mobile Delay/Disruption Tolerant Network (MDTN): a delay tolerant application platform built on top of a PTS and able to provide opportunistic service connectivity. In essence, the PTS is used as a communication backbone. However, different from other approaches relying on a backbone of vehicles to support connectivity (e.g., [21, 22]), data forwarding is achieved by implementing a delay tolerant, store-carry-and-forward, communication model where a mobile user can delegate a request for service involving Internet access to a carrier entity (e.g., an access point on a bus). The request also contains the information regarding the bus line where the user is expecting the response to be available. Depending on the model of the accessed service (pull or push), the user is later on provided with the requested content or a simple notification of request satisfiability.

Once a request has been issued, it is opportunistically forwarded by the carrier toward Internet Gateways (IGs) located at bus terminals and, finally, the response has to reach the bus line where it will be collected. This process involves multihop opportunistic routing of both request and response, from carriers toward one of the IGs and from the IG toward the bus line hosting the destination, respectively. Along its trip, the request can be served by an IG at any traversed line end. From there on, the message will be considered a response and forwarded toward the destination line. If, during forwarding, the request ends up on the destination line, it will be up to the destination line to satisfy the request using its terminal IG.

To analyze the performance of MDTN, we have exploited two realistic deployment scenarios where carriers are public buses with routes corresponding to actual PTS lines in Milan (Italy) and Chicago (IL, USA), where users are roaming entities owning handheld devices. The remainder of this paper is organized as follows. In Section 2 we summarize background information needed to comprehend the applicative domain under scrutiny. Section 3 introduces a case study exemplifying MDTN *modus operandi* along with some representative application fields it can support. In Section 4 we provide a concise survey of related works following in this context. Section 5 discusses MDTN design choice, the service architecture, and deployment strategy while Section 6 describes the simulated environment. Section 7 discusses the experimental results and brings a testimony from fields trials performed with proof-of-concept implementation of the system. Finally, Section 8 concludes the paper.

2. Background

Opportunistic networks are considered as an evolution of the original outer space DTNs where (un)planned contact opportunities are exploited to move data between interested parties. While in outer space communication link uptimes can be predicted with a high degree of confidence, OppNets in the general case are characterized by unpredictable delays [12]. Moreover, networking techniques devised for Mobile Ad Hoc Networks (MANETs) might not be suitable due to their underlying assumptions: the existence of a path between the source and the destination and the willingness of intermediary nodes to act as relays for data forwarding [12, 23].

While medium access and transmission techniques can be addressed by means of existing solutions, routing in such environments is still an open issue. The proposed solutions vary depending on whether infrastructure support is present or not. However, they all adopt the same basic strategy whereby node mobility is exploited to move data between involved parties. Routing/forwarding strategies alternate between muling and transfer of the data, the latter taking place when communication opportunities arise. When this happens, entire chunks of messages are transferred from one storage place to another, travelling along a path that is expected to eventually reach the intended destination. To counteract the unpredictability of message delivery some proposals employ redundancy, injecting into the network multiple copies of the same data, augmenting the chances of data delivery.

Figure 1 depicts the taxonomy of opportunistic forwarding schemes. Depending on whether the system involves infrastructure entities in the forwarding process, they can be categorized into two subgroups, namely, (i) infrastructure-based and (ii) infrastructure-less forwarding schemes.

Data dissemination strategy diffuses the message in the entire network, as at each opportunistic encounter nodes are infected with the message [24]. The rationale behind this policy is that since there is no knowledge of a possible path toward the destination or of an appropriate next-hop node, a message should be sent everywhere, and eventually it will reach the destination. This scheme increases the delivery probability as the message is diffused onto all possible paths. However, this comes at an additional cost in terms of wasted bandwidth and network storage.

Other schemes employ a controlled data dissemination technique whereby nodes maintain a local sate, history of the encounters, and the next-hop forwarding decision is made based on some utility metric [23, 25]. In this way, system redundancy is limited at the cost of a lower delivery probability and higher delivery delays. Despite these efforts, data dissemination techniques are undermined by the unpredictability of human behavior. More viable solutions are commuted under the infrastructure-based approach, where mobile infrastructure entities are involved in data forwarding.

Carrier-based solutions employ mobile infrastructure nodes acting as data collectors. Nodes move around in the network area, following either predetermined or arbitrary routes, gathering messages from the nodes they pass by. In particular, carrier-based solutions deployed on top of the PTS have drawn the attention of researchers because they inherently help to mitigate the well-known criticalities of the human counterpart and at the same time exhibit some peculiar behaviors. First, when compared to portable wireless
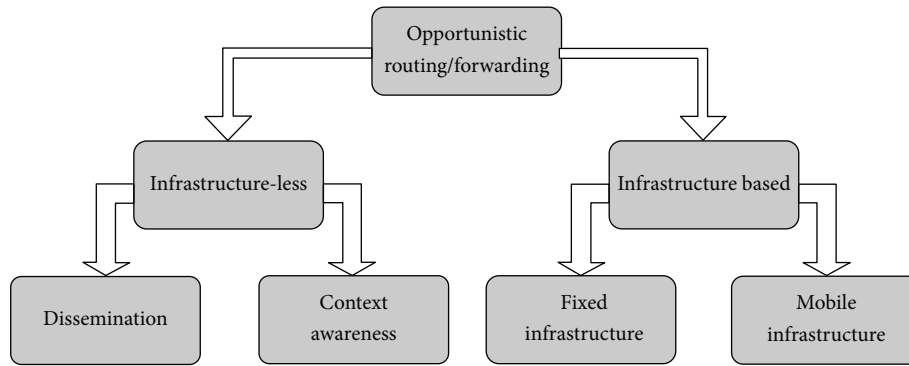
FIGURE 1: Opportunistic routing taxonomy.

devices buses are powered nodes whose lifespan is not affected by routing operations. Second, even though everyday experience leads to mistrust, bus mobility and schedule can be considered as quasideterministic; hence link uptimes are predictable. Last, PTS buses ensure a pervasive coverage of the metropolitan area. These properties taken all together promise for packet delivery platform which may support the deployment of a robust, metropolitan-wide, infrastructure-free, and provider-less wireless platform.

In the following we are going to describe a use case which we consider meaningful to motivate and explain MDTN functionalities. Next, through Section 4 we survey similar approaches falling into this context.

3. Case Study

In our case study we consider a user, Alice, who is living outside the main body of a huge city and is required to commute every day to her workplace. As sometime happens, villages outside the city are not well covered by LTE and it might also happen that wireline connection is limited to dial-up due to a low number of potential subscribers. In this situation, Alice cannot get a good data connection from home and must rely on a public service or needs to go in an area with better wireless coverage in order to perform a proper data transfer.

Every day, in the morning, Alice leaves her home and starts a trip to go to the office. The trip is quite long since she lives outside the city. As soon as the closest bus station is reached, LTE coverage is good enough to download all data required for the day; this includes e-mail, some newspapers, and a big digest with her favorite social networks. Unfortunately, as it is for the majority of subscribers, Alice's data plan is limited and her monthly allowance will be depleted in a few days if all the transfers we just mentioned would be performed using the cellular network. Nevertheless, the access point on the bus which Alice takes regularly has already preloaded all the newspapers she is subscribed to and the social network digest. This is because Alice requested the data to be here earlier on. The bus collected this request and used the broadband connection available at the end of the line to download everything that was requested. After retrieving

everything from the bus via WiFi, only the e-mail is left for Alice, but that alone is now acceptable to do over cellular network.

While on the move, Alice starts consuming the data collected from the bus. While reading one of the newspaper she reads an interesting advertisement about discounts from a travel agency. The whole catalog might be huge to download and also long to read. Anyway, this is not a problem for Alice which is leaving the request to the bus along with an indication about the time she will be leaving the office and which bus lines she will be using today on her trip back. The bus caches the requests and Alice goes on reading the newspaper. Alice is not concerned about receiving the catalog later in the afternoon: she will read it anyway while at home and the travel agency must be visited in one of her off days.

Once in the office, Alice can enjoy a broadband connection but as it happens in many workplaces, access is limited to professional websites and content. Alice is able to refresh her newspapers with the afternoon edition but does not have open access to social networks.

Later in the afternoon, when the office work is over, Alice is on his way back to home. By using the information she provided in the morning, the travel agency catalog and the social network digest are already on the bus waiting. At the same time this same information has been leveraged by the public transportation authority: a strike is planned tonight and Alice receives a warning with suggestions about possible alternative routes. Moreover, as an added value, the travel agency catalog in its way to Alice has been associated to a number of local data sources; additional information has been piggybacked to the original document to report discounts in nearby travel agencies. This is convenient because Alice can go there during the lunch break.

In the following section we provide a thorough and concise survey of approaches employing the PTS as a service delivery platform.

4. Related Work

Among all the possible real-field applications of DTN, Opp-Nets built on top of a PTS are characterized by specific features: buses intercontact times are quite long, contacts

occur according to a schedule, and their connections are mostly respected. Routing policies are heavily influenced by the determinism of such a scenario.

Despite the fact that the everyday metropolitan experience suggests mistrusting determinism in bus contacts, some routing proposals in the literature still attempt to exploit it through different types of oracles. In the considered scenario, time tables are required to take into account the traffic conditions, and the concept of connection between two bus lines has a relaxed time constraint, which is eventually respected by increasing the number of buses belonging to each line. Furthermore, intermittent connectivity is caused by a variety of reasons, yet the topology often has an underlining stability. In [26], a *Contacts Oracle* is proposed that, given two buses IDs, returns the next encounter time. This oracle requires *a priori* knowledge of encounters and is impossible to implement in a real system. In the same work, a more feasible approach uses a *Contacts Summary Oracle* that, given two buses IDs, returns the average intercontact time.

In literature, proposed approaches are usually classified in three categories based on the environment where they operate: rural, campus, or urban.

A rural environment [19, 27, 28] is comprised by a number of villages spread over a large territory and connected typically by buses. In this context, PTS schedule is not subject to changes over time and contact opportunities are rather infrequent. Data delivery failure in this environment is mostly as the result of a missed transfer opportunity rather than unpredictable node mobility. The proposals that fall into this category do not employ routing. Both requests and responses are locally stored at an infrastructure entity, acting as proxy servers between end-users and the Internet, and it is up to the carrier to download the queued requests and upload the responses. In [27] the PTS is used as an opportunistic backbone to carry messages between collection points (kiosks) located in the participating villages; buses are used as *data mules* with a best-effort approach. A more refined approach is proposed by [28], where bus-to-bus connectivity is exploited to forward messages over multiple hops. In this work timetables are considered: authors propose an algorithm computing message delivery probability along a forwarding path. In [19], authors propose a modified link state routing protocol capable of exploiting link uptime predictability. The proposed algorithm builds the forwarding path by exploiting link state advertisements pushed and cached into intermediate nodes of the network. The shared goal of all mentioned projects is to provide network access for elastic nonreal-time applications in order to enable base Internet services to the population (e.g., mail and nonreal-time web browsing).

Further in this steam of research we find campus bus networks which are deployed to commute students and faculties between campus areas or from/to surrounding areas [29, 30]. This kind of service is usually characterized by a higher number of nodes when compared to a rural environment which translates into a higher number of transfer opportunities. The reference contribution in this context is represented by [29], where five colleges are connected with nearby towns and between each other. In their work, the authors propose MaxProp, a multicopy forwarding scheme based on node encounter history and message priority, both taken into account for computing the path likelihoods to destination nodes. By means of simulation MaxProp is shown to outperform protocols based on knowledge of deterministic meetings between peers. In a similar spirit, the authors in [30] analyze intercontact times distribution at both bus and line level. They propose a generative model for intercontact times between buses validated against real traces which could be employed to drive simulations on routing protocol performance.

Scaling up in terms of PTS size and shape, we find the urban environment comprising a considerable number of bus lines deployed to assist people commuting inside a city. Bus networks in urban environment [31–33] are usually characterized by many periodic contact opportunities. In [31] authors propose a commercial application, namely, *Ad Hoc City*, based on a hierarchical wireless ad hoc network architecture. The system provides service support for elastic, nonreal-time traffic by means of access points responsible for different geographical areas. To this end, the PTS is employed as an ad hoc routing backbone, carrying messages from/to mobile phones to/from the access points. The authors validate their approach against real mobility traces taken by the King County Metro bus system in Seattle, WA. Using the same traces as in [31], authors of [32] propose a forwarding scheme for intracity message delivery. The proposed scheme employs a large-scale clustering methodology based on node encounter frequency. This resulting structure is then exploited by the forwarding scheme which switches to a multicopy strategy toward members of the same cluster as the destination.

However, adopting a multicopy approach is scarcely appealing in a wide urban environment. To this end, authors of [34] model the forwarding process as an optimal stopping rule problem; this way traffic overhead is sensibly reduced while delivery ratio is still comparable with a fully epidemic approach. All the city-oriented strategies mentioned above follow one common approach: they perform a multiple-copy routing. As already stated in this paragraph, having multiple copies of the same packet competing for network resources may be an hindrance when scaling up to city level using dozens, if not hundreds, of routes counting many hundreds of buses. Single-copy strategies have not been really considered in literature due to the possibly low delivery ratio and long delivery time.

To the best of our knowledge, [33, 35] are the only two contributions with a strong push toward single-copy forwarding on a very large scale. Reference [33] considers intracontact times as a metric for a link state protocol; as a result, routing is performed through the most encountered line in terms of time regardless of the real encounter probability. In [35], instead, an estimation of encounter probability is used as a metric: routing is performed along a path which minimizes the probability to leave a packet undelivered. In this second case the routing algorithm is link state as well; moreover, extemporaneous contacts that are branching to a more favorable path are also exploited in order to improve performances in an opportunistic way.
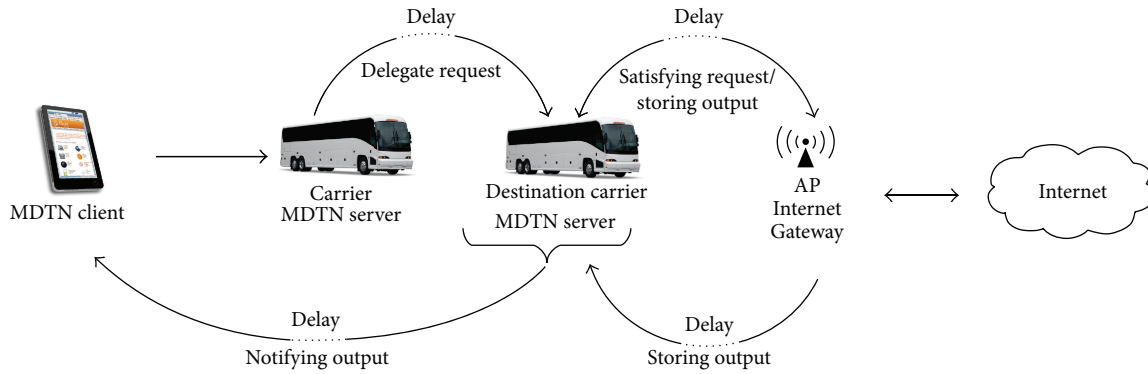
FIGURE 2: MDTN system orchestration.

Further contributing in this realm of research, we propose a concrete deployment architecture and an applicative scenario exploiting the PTS as a service delivery platform. To this end, we have modeled two realistic deployment scenarios where carriers are public buses with routes corresponding to actual PTS lines in Milan (Italy) and Chicago (IL, USA) while users are mobile entities owning handheld devices issuing data requests for data available elsewhere. We study the performance trend of our Mobile Delay Tolerant Network (MDTN) in an metropolitan-wide deployment under different routing strategies and data distribution schemes.

5. Mobile Delay/Disruption Tolerant Network (MDTN)

This section outlines MDTN. We start by introducing the system architecture along with the delivery process constituting the data flow in our network. Concluding this section is an overall description of the protocol implemented by MDTN entities.

5.1. Modus Operandi. In Figure 2 the MDTN entities are depicted taking part in our system: (i) the MDTN client or a mobile user operating a wireless-equipped portable device, (ii) the MDTN server or the wireless data carrier which gathers users requests trying to fulfill them when Internet connectivity is present, and (iii) the Internet Gateway (IG), which is an Internet-enabled wireless access point deployed at each bus end of line (e.g., bus terminals). We have considered positioning IGs at bus terminals since these are joint points between different bus lines and because the bus company may already have an infrastructure-based end point there (e.g., offices). However, we do not preclude the possibility of a future investigation of IG distribution in terms of a scientific optimization problem, even though less practical in real deployments.

The MDTN data flow starts with a user establishing a connection with the on-board hosted server (e.g., on board the bus), successively issuing a request for a specified data content. For simplicity we assume that the request consists of a content identifier (e.g., a uniform resource locator) and a destination bus line dictating the destination where the user

is going to pick up the response later on. The goal is to provide the user with the desired content at the specified destination line. After the request has been issued, it is opportunistically forwarded by the carrier toward one of the IGs and, finally, the response has to reach the bus line where the content is expected.

The process described above involves multihop opportunistic routing of both request and response from carriers toward one of the IGs and from the IG toward the destination bus line. The delivery process is independent from the underlying routing strategy and occurs as follows.

(i) If the carrier holding the request encounters during its trip a carrier belonging to a valid next-hop line, criterion based on the adopted routing strategy, the request is forwarded. It is now up to the new carrier to satisfy the request if it reaches its end of line.

(ii) If the carrier holding the request reaches its terminal, it is able to fulfill the request by itself; the response will be forwarded toward the destination line.

(iii) If the response reaches a destination line carrier, response dissemination among carriers of the same line can take place. In this phase, forwarding occurs during opportunistic contacts between destination line carriers (if any).

The rationale behind the last announced step of the delivery process is that there might be many carriers travelling on a destination line and outcome dissemination among them is needed, speeding up the delivery process.

From the end-user perspective, the requested content (if retrieved) is fetched when a communication opportunity with the destination line server arises. If this opportunity is missed the data is cached until the next encounter. This delivery process is entirely transparent to the client.

5.2. MDTN Protocol Stack. MDTNs functional entities adhere to the same communication protocol which in our case is a tuned version of the Bundle Protocol (Figure 3). Consequently, every node that is able to send and receive bundles is called a bundle node on top of which is layered the application agent, specific to the entity under consideration.
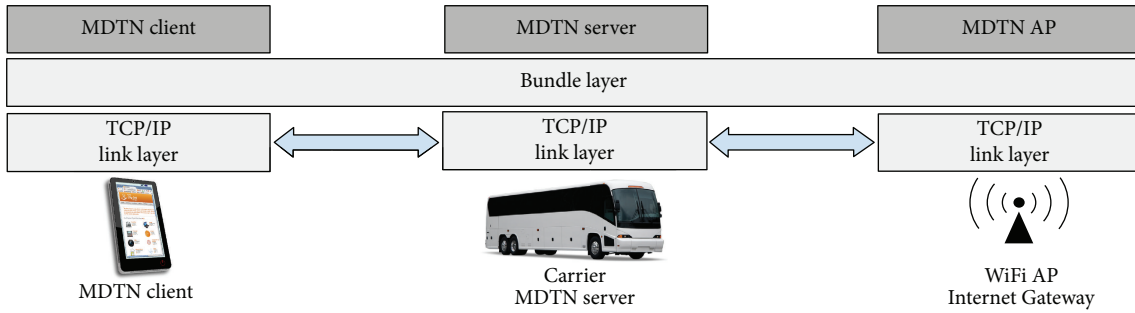
FIGURE 3: MDTN protocol stack.

In particular, the MDTN client can be in one of the following two states:

(i) *online*: a session is established and active with the MDTN server;

(ii) *offline*: otherwise.

Respectively, the MDTN server can be in one of the following two states:

(i) *gathering*: no Internet connectivity is available. The server is actively gathering user issued requests and forwarding previously gathered requests when the opportunity arises;

(ii) *digesting*: server reaches an IG where Internet connectivity is available and is able to accomplish (digest) pending user requests.

To avoid ambiguities, we point out that there is no substantial difference between the carrier and the AP application agent. The sole difference stands in the fact that the MDTN server makes use of additional forwarding metrics inherited by the routing strategy being adopted, while the application agent at the AP does not employ any ad hoc forwarding metrics. The MDTN AP can be viewed as lightweight MDTN server capable of Internet connectivity which has the ability to distinguish the delivery scheme being employed (later on). This is achieved by appropriately modifying a system configuration parameter prior to system boot.

6. Simulation Environment

In this section we start by providing a concise survey of the simulation framework adopted for the validation of MDTN. Next, we provide details about specific configuration parameters employed in the experimentation part.

6.1. The URBeS Simulation Framework. For our simulations we have adopted the Urban Routing BackbonE Simulator (URBeS): an ad hoc simulation environment introduced and validated in [35]. The functional scheme of URBeS is reported in Figure 4. Briefly, URBeS acquires real city topological data and the relative PTS timetable to accurately recreate bus movements in a real urban environment in order to simulate data forwarding between buses as well as between buses and
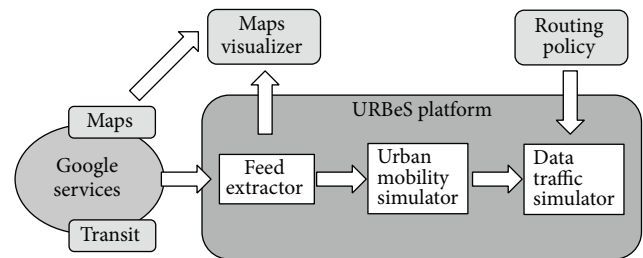

FIGURE 4: URBeS functional scheme.

road-side network devices. The simulation platform presents a flexible architecture, facilitating the adoption of external routing policies for comparison purposes.

URBeS acquires the topological data from a Google Transit [36] feed, which is a public transport route planner allowing access to raw data of a PTS. In order to prepare a simulation world reflecting the reality, the acquired data undergoes a complex process comprised of the following three sequential phases.

In the first phase of the process, the information is parsed producing a timetable of bus movements together with a topology of the PTS layout. In a next step the GPS coordinates comprised in the feed are converted to Cartesian coordinates.

The second phase produces the bus mobility traces taking into consideration the real PTS timetable and the data produced from the first phase. Buses move at constant speed between stops and pauses at stops (when planned) are simulated accordingly. In this phase, URBeS also computes statistics about bus contacts which are useful for understanding intra- and intercontact times and city coverage of the PTS.

The third and last phase introduce data traffic into the picture which is an essential element given the purpose of the framework. Currently, network traffic is randomly generated at each bus and delivered according to the configured routing policy. Detailed information regarding events occurring during the simulation is logged and could be later on used to gain insights on the routing algorithms. The simulator can provide a valid comparison between different routing policies while allowing easy testing on multiple urban environments. Results from our simulations have been positively validated against GloMoSim, for low traffic levels, using the same output coming from the second phase of URBeS.
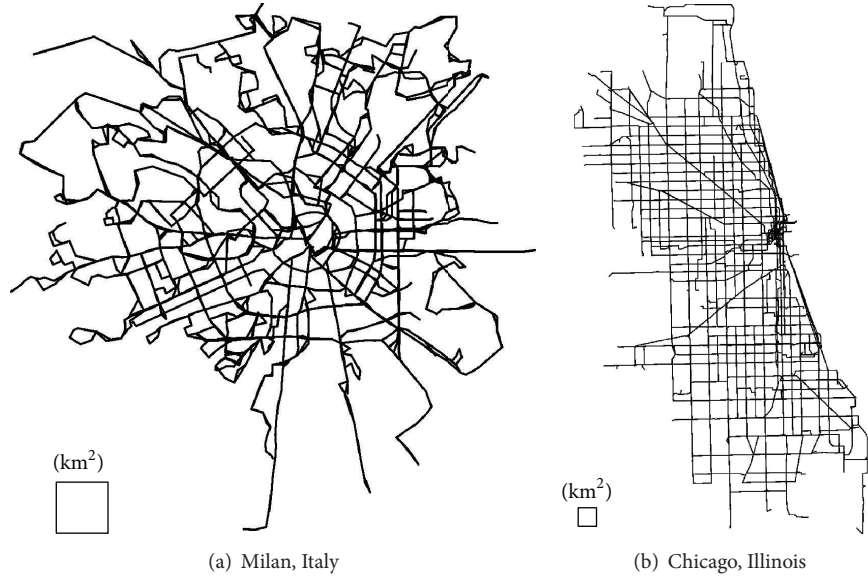
(a) Milan, Italy   (b) Chicago, Illinois

Figure 5: Maps of public transportation systems used in the simulations. Measurement unit in km^2.

6.2. Urban Environment.

In order to evaluate our proposal we selected two cities: Milan (Italy) and Chicago (IL, USA). These cities have been selected as representative examples of a vast group of cities all around the world.

Milan is our first case study, representing a medium size town compared to typical European cities, with a high bus density (69 bus lines) where routes wind around a circular city plan. The topology presented in this case is clearly not Manhattan-like due to the presence of the old Roman historical center and the progressive annexing of small peripheral towns in the main city body. In this scenario, there is no constant space between intersections, crosses between bus lines occur frequently, and streets do not run parallel one to another for very long. In these settings, with no freeways crossing the city, bus speed is generally low and foremost uniform.

The second city we consider is Chicago, expressly chosen to test scalability due to its immense size. Chicago has many bus lines (142) with routes running along the coast and then almost uniformly in inland, with very low density. The PTS structure has a Manhattan-like plan, typical for many huge cities in USA. In this kind of setting we usually experience short contacts, mainly at intersections, at a low rate.

Figure 5 shows the PTS maps used in our simulations, while Table 1 presents a summary of the two cities and relative PTS layouts. In the table, bus line density is calculated as the mean number of km covered by buses over a km^2.

6.3. Simulation Parameters.

In our simulations each bus is equipped with a wireless network interface. The available bandwidth is 10 Mbps and radio range is 100 m. Urban canyons created by buildings are taken into account during simulation, thus allowing only line-of-sight contacts.

Simulation starts at 4 a.m. (the first bus starts its trip at 5 a.m.) and ends at 8 a.m. of the following day (the last bus

Table 1: Properties of PTS layouts for the two cities.

	Milan	Chicago
City size in km^2	125	972
Number of lines	69	142
Length (mean ± std. dev.)	15.8 ± 5.7	24.4 ± 11.2
Density	13.85	5.73

goes out of service around 6 a.m.). All buses departing before 4 a.m. are considered as on duty during the night and not in the morning.

With regard to traffic generation, our aim is to simulate data exchange during an ordinary working day. Data traffic generation is a continuous process occurring during working hours: from 8 a.m. to 8 p.m. During simulations, requests are generated at a constant rate for every operating bus. Each bus receives 10 service requests per hour from users on board. These requests are accepted as long as the bus is in service, even when waiting for the next departure at the end of the line. For each request, the user specifies a destination line where she/he wants the response to be delivered. The destination line is chosen randomly on the PTS using a uniform distribution.

We assume that each bus is equipped with enough storage capacity, in line with modern equipment for industrial PCs. In order to simulate both typical web-based and generic data traffic (mail, forum posts, social network interaction, etc.), the size of request packets varies uniformly from 1 KB to 40 KB, whereas the size of response packets varies from 10 KB to 64 KB, thus representing classic emails, web pages, and advertisement messages without heavy multimedia attachment so as to be able to actually provide useful services even with intermittent, opportunistic connectivity. Each issued request is gathered on the bus local storage and stays in there until a forwarding opportunity arises.

Buses reaching their terminal station may queue up for another scheduled departure. If the bus is still in service it will hold all the gathered data and welcome other requests issued in its surroundings. Otherwise, all the stored contents will be pushed to the first bus waiting in line. If there is no other bus around (i.e., there are no more scheduled departures) all the stored data, being either requests or responses, are dropped and considered lost.

Forwarding can take place depending on the routing policy and distribution scheme between the base station on board to any other bus and between a base station and an IG deployed at each line end. In the following, we identify a routing policy as the set of rules used to identify a feasible next hop whereas a distribution scheme will be the set of rules to decide whether a packet will be forwarded. In our simulation we considered the following three distribution schemes that could be exploited by MDTN:

(i) *Pure Muling* (PM). Each bus must keep all its requests until it reaches the end of line. When the end of line is reached, requests are satisfied and responses are immediately sent to the IG(s) of the destination line.

(ii) *Infrastructure-Less Delivery* (ILD). Requests are forwarded toward the destination line using a routing policy (later on). If, along the path between source and destination, an IG is encountered (i.e., the bus is in transmission range of an IG), requests are satisfied and changed into responses, which are forwarded toward their destination line adopting a specific routing scheme. Otherwise, requests are delivered to a destination line carrier, which will eventually meet an IG (e.g., at the end of the line), thus generating related responses.

(iii) *Infrastructure Aided Delivery* (IAD). Requests are forwarded toward the destination line as in the previous case. When an end of the line is reached and requests are satisfied, responses are sent immediately to the IG(s) of the destination lines using a wired network.

With regard to the routing policy we need to test how MDTN can benefit from legacy or innovative routing approaches. For this reason in our simulations we compared three different routing algorithms that can be considered representative of a broader set of solutions; they are Minimum hop (Min hop), MaxProp [29], and Op-HOP [35, 37].

Min hop follows a single-copy, link state routing approach, which should be able to exploit the specific design of a PTS, usually devised to bring people (as packets) across the metropolitan area minimizing the number of transits. Link state routing is extensively adopted in wired networks and we use it as a benchmark to understand how much MDTN can benefit from more sophisticated routing. Moreover, a link state approach is, in general, feasible for a PTS because all bus routes are known in advance and should not change without notice during the day.

MaxProp is considered the state of the art for DTN deployment over a PTS: it uses a multicopy routing algorithm and implements a sophisticated buffer management based on messages priority. Being multicopy, MaxProp uses more

Table 2: Summary of delivery delay in Milan (values in hours).

	Mean	Median	Std. dev.
Pure muling	1.56	1.41	0.86
Infrastructure-less delivery			
Min hop	3.89	3.10	2.63
Op-HOP	2.70	2.49	1.25
MaxProp	1.59	1.51	0.61
Infrastructure aided delivery			
Min hop	1.59	1.41	0.65
Op-HOP	1.82	1.59	1.13
MaxProp	0.71	0.65	0.36

Table 3: Summary of delivery delay in Chicago (values in hours).

	Mean	Median	Std. dev.
Pure muling	1.72	1.27	1.62
Infrastructure-less delivery			
Min hop	5.25	4.58	2.85
Op-HOP	5.14	4.69	2.44
MaxProp	3.11	2.69	1.77
Infrastructure aided delivery			
Min hop	2.24	1.63	1.92
Op-HOP	3.16	2.37	2.57
MaxProp	1.28	0.85	1.43

resources while generally providing good performances in terms of delivery delay.

The last protocol, Op-HOP, also uses a single-copy link state approach. Different from the Min hop, Op-HOP adopts a metric based on probability: paths are calculated based on lines encounter probability to maximize packet delivery. Moreover, unlike other solutions, Op-HOP probability is estimated based on the number of encounters rather than their duration [37]. Op-HOP has already been shown to scale better than MaxProp in a purely ad hoc environment.

7. Experimental Results

A prospect of average delivery delays under the various combinations of routing policies and distributions schemes is reported in Tables 2 and 3.

As we can see from Table 2, in Milan with the ILD scheme, responses will be delivered with an average delay varying from less than 2 hours to almost 4 hours, the Min hop policy being the worst case. Under these lenses, the system could be actually used for news retrieval, delay tolerant web browsing, and distribution of information regarding local events. In Chicago (Table 3), the situation is a little worse: in the ILD scheme delay ranges between about 3 and 5 hours. News retrieval is still feasible but other services might become unrealistic.

A different scenario comes forward when the distribution scheme changes and a request is routed only up to the first IG, while its response is transmitted to the IG of the destination line via wireline. In addition to the fact that the average
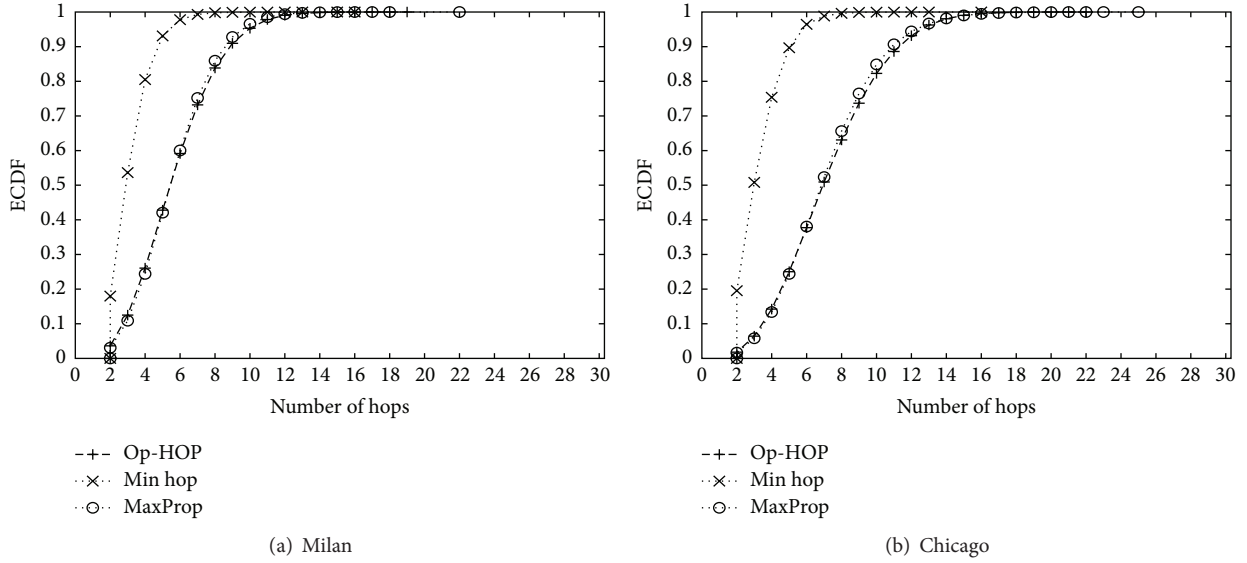
(a) Milan

(b) Chicago

FIGURE 6: ECDF of number of hops using the ILD distribution scheme.
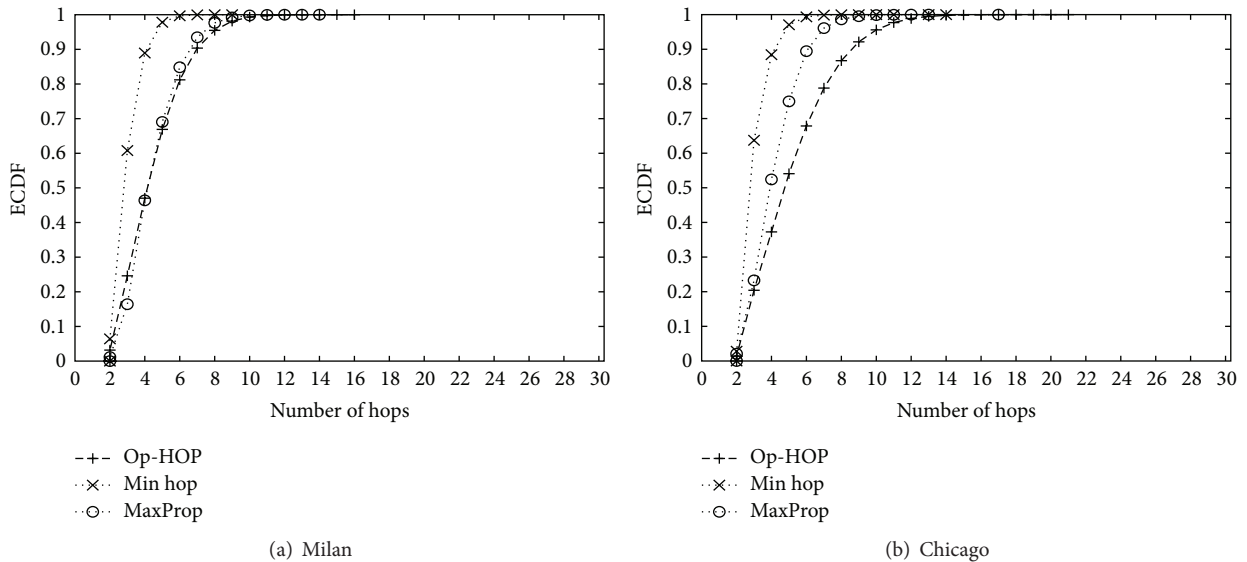

(a) Milan

(b) Chicago

FIGURE 7: ECDF of number of hops with the IAD distribution scheme.

delay time is reduced, Min hop starts performing better than Op-HOP. The explanation of this sits in the optimization of Op-HOP against Min hop: the latter is more likely to miss contacts and carry the request toward the IG of the first line, thus reducing the average delay more than Op-HOP. In all cases, delay is adequate for metropolitan-wide, nonreal-time services.

If we combine Table 2 with Table 3, it is possible to draw some more general conclusions. First of all, the Min hop routing policy albeit designed for wired networks performs in a way comparable with the other routing policies. Second, MaxProp seems to be constantly outperforming the other two routing policies. In particular, regarding Op-HOP, in [37] we demonstrated that it scales better than MaxProp in terms of network load, but it seems to follow the same trend

when scalability comes in terms of PTS extension. This can be explained thanks to the multicopy forwarding approach of MaxProp.

7.1. Number of Hops. A performance metric closely connected to the delivery delay is the number of traversed hops. This is because every hop is the result of a contact opportunity or a (sometimes long) travel toward an IG.

Through Figures 6 to 7 the ECDF of the number of traversed hops for each of the studied topologies under the different distribution policies is reported.

In the charts, the *pure muling* policy is not reported since it would not bring any meaningful information (the number of hops is fixed) and all profiles have a minimum of two hops because the data have to traverse at least an
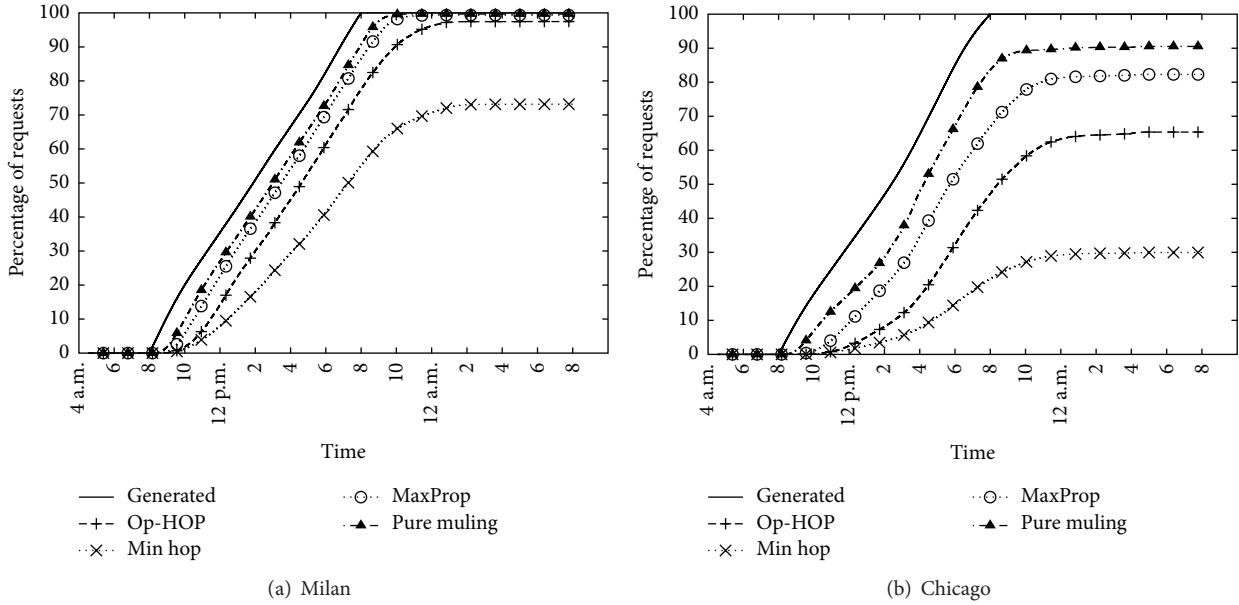
(a) Milan

(b) Chicago

FIGURE 8: Request delivery rates with the ILD distribution scheme.
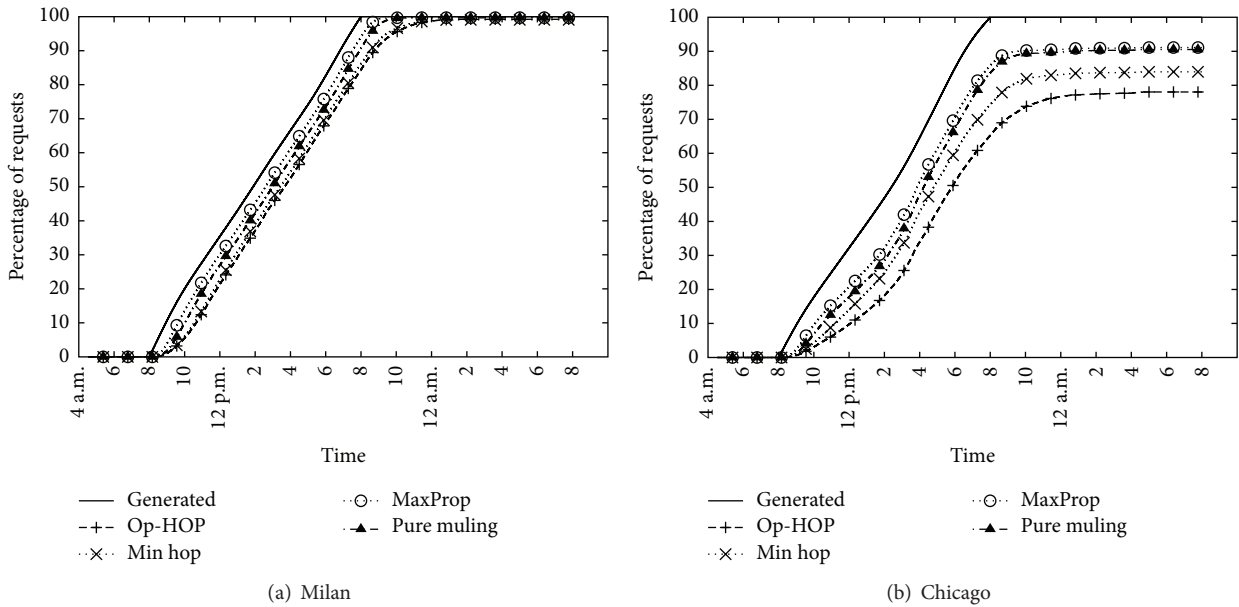

(a) Milan

(b) Chicago

FIGURE 9: Request delivery rates with the IAD distribution scheme.

IG before reaching the destination. Analyzing the charts, we can observe that, despite the lower delivery delay, MaxProp does not outperform Op-HOP in a sensible way. The only exception is Chicago with the IAD scheme (Figure 7(b)), but this is more likely due to a scalability problem of Op-HOP.

7.2. Delivery Rates. Figures 8 and 9 show the request delivery rates in the various cases. First of all, Min hop always generates the worst performance (sometimes to an unacceptable level such as in Figure 8(b)) unless the system is extremely connected (Figure 9(a)). This can be explained considering the extremely long delivery delay: packets will accumulate

during the evening while buses going out of service will make the network less connected, increasing the number of packet drops. Op-HOP is not suffering to the same degree because paths are built based on encounter probability: a smaller fraction of the total traffic will stay in the system for a much longer time and will be dropped at the end of the day, but a more considerable number of packets will be able to find its way to destination.

Second, we have to observe that not even the PM distribution scheme is able to guarantee a 100% delivery of packets. This is particularly true for Chicago (Figures 8(b) and 9(b)) and depends on the PTS being scarcely connected: buses are
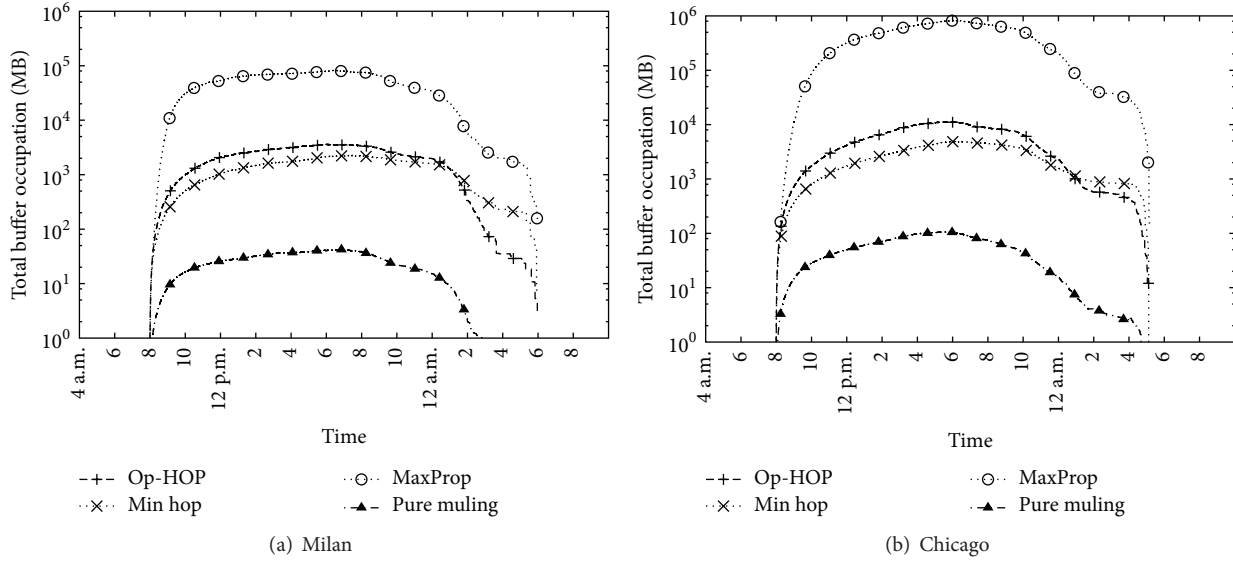
(a) Milan

(b) Chicago

FIGURE 10: Buffers usage in the ILD distribution scenario.
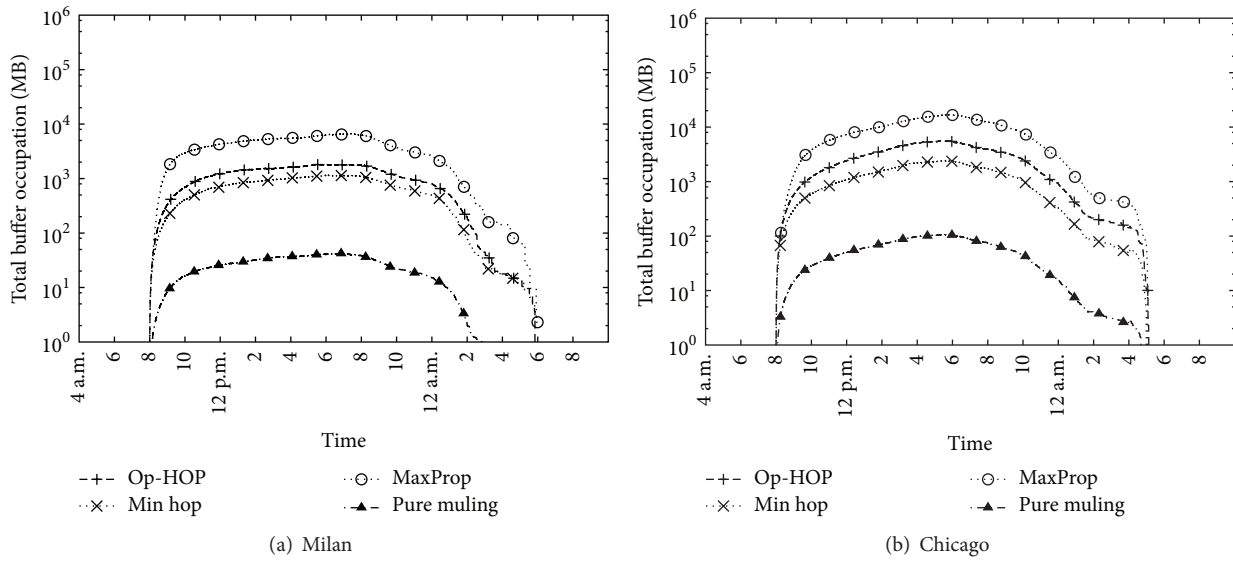

(a) Milan

(b) Chicago

FIGURE 11: Buffers usage in the IAD distribution scenario.

less likely to be able to spool packets from the internal buffer when going out of service. Nevertheless, MaxProp is able to slightly outperform the PM scheme but it needs assistance from the IGs.

7.3. Resource Usage. So far, MaxProp seems to be the best option for MDTN: lower delivery delay and acceptable delivery rate. Nevertheless, we have to keep in mind that many of these advantages come due to its multicopy approach. More likely, it is going to use more resources than the other routing policies. To this aim, in the following subsections we analyze the usage of system resources under the various routing conditions.

7.3.1. Buffers Usage. The first metric we are going to analyze is the global buffer usage. Figures 10 and 11 depict the evolution of the total amount of traffic in transit in the network for different distribution schemes.

As we can see, MaxProp is always more demanding than the Min hop routing policy. The effect is sensibly reduced only when transmission between IGs is used. Interestingly enough, these are the cases where performance differences are less prominent when compared to the other policies. The difference between the profiles of Min hop and Op-HOP is due to different delivery delay: the added request from Op-HOP is the space required by packets taking a longer route to reach destination.
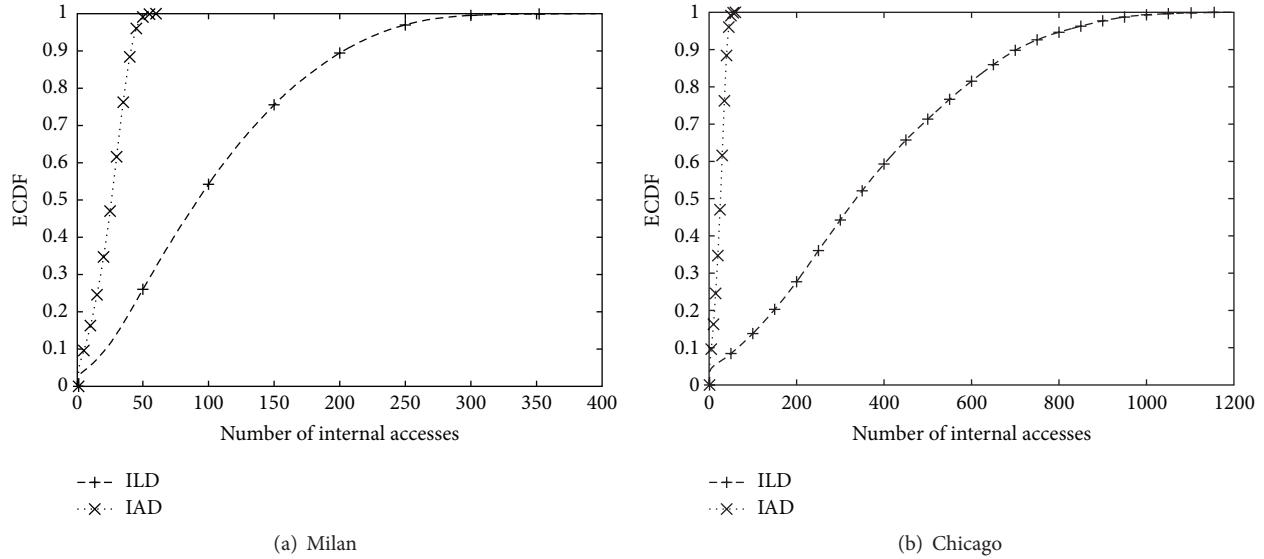
(a) Milan

(b) Chicago

FIGURE 12: ECDF of the number of unrequired Internet accesses for MaxProp.

7.3.2. Wasted Internet Access. Another resource for MDTN is the Internet access: an IG, upon receiving a packet, must use its access network to fulfill the request and produce a response. In the case of multicopy routing (MaxProp), multiple copies of the same request will require multiple accesses to the Internet with a consequent waste of resources. Figure 12 shows the ECDF to have a given number of Internet accesses for a given request.

As we can see from the curves, with the ILD scheme, only 50% of requests will generate less than 100 Internet accesses in Milan and less than 350 in Chicago. These values drop considerably when the IAD scheme is adopted, lowering down to 25 in both cases.

Considering the cost of memory, we can sustain a certain degree of overallocation; Internet access may be expensive and increasing usage by 20 times may have a severe impact on the system yet. This situation calls for a carefully planned content delivery network tailored for metropolitan services accessed via PTS.

7.4. Field Trial with MDTN. At this point, we provide evidence of proof-of-concept implementation of our service architecture and the experience gathered from a field trial which has taken place in Padova, Italy.

The current implementation of MDTN is a Java based application built for Android capable devices and tested for Android OS 5.1. It consists of a package composed by a client and a server both sharing the same core module handling a DTN-like communication. The client consists of a tabbed interface allowing the user to manage and easily interact with the MDTN services. The current implementation contemplates two basic services which are (i) a delay tolerant e-mail service and (ii) a file download which for now is limited to the download of a web resource.

In the following, we show by means of a field experiment a concrete usage of MDTN. The map in Figure 13 shows the
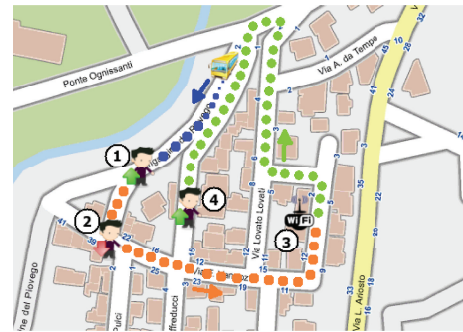

FIGURE 13: Demo map showing the interactions along the route between carrier and user. The blue dots represent the car route with no pending tasks. The orange dots represent the carrier route where there are pending tasks, issued by the user at ②. Along the green dots path the carrier has satisfied user requests (at ③) and is ready to forward the output when user gets on board at ④.

interactions taking place between the carrier and the user, respectively, operating MDTN server and client. In this demo the carrier is a car with a predetermined route and an on-board WiFi AP while the user is mobile and operating MDTN client on an Android enabled device.

The enumerated points in the map represent the following actions:

(1) The user gets on the car and successively forwards MDTN server a task.

(2) The user gets off the car and continues his journey.

(3) The carrier is nearby a wireless AP connected to the Internet (e.g., bus station), satisfies users pending tasks, and stores the output locally.

(4) The user gets on the car and once connected to the MDTN server retrieves the required content.

(a)

(b)

(c)

(d)

(e)

Figure 14: MDTN field trial.

Through Figure 14 the aforementioned interactions taking place for each designated point of the map are shown. In Figure 14(a) the vehicle (carrier) leaves the station equipped with Internet connection and travels along the predetermined route shown in Figure 13. In Figure 14(b) the user gets on board the vehicle, establishes a connection to the MDTN server, and forwards some requests (Figure 14(c)). Once user has arrived at destination, he gets off the vehicle, disconnecting from the server; meanwhile the carrier follows its route toward the station (Figure 14(d)). Leaving the station the carrier has satisfied user requests and stored their output locally. Figure 14(e) shows the user getting on board the vehicle, connecting to the MDTN server in order to retrieve the requested contents.

8. Conclusion and Future Work

In this paper we studied the performance of MDTN: a delay tolerant application platform built on top of a PTS and able to provide opportunistic connectivity. We showed that MDTN could be a viable solution elastic, nonreal-time data retrieval.

Nevertheless, performance indexes of the considered routing policies have shown that there is no golden rule for routing. Infrastructure aided delivery will actually make a difference in the service provisioning. On the other hand, MaxProp and multicopy routing approaches have to be preferred in loosely connected environments despite the fact that we still need to understand if the added resource usage is worth the performance improvement.

In the future we are planning to extend this work to find a dynamic solution for the tradeoff between single- and multiple-copy routing algorithms based on PTS density and city topology. This new routing policy should be able to switch between the two forwarding modes in order to improve performances while preserving resource usage.

Competing Interests

The authors declare that they have no competing interests.

References

[1] S. Basagni and A. Capone, "Recent research directions in wireless ad hoc networking," *Ad Hoc Networks*, vol. 5, no. 8, pp. 1205–1207, 2007.

[2] A. Kaiser, N. Achir, and K. Boussetta, "Multiplayer games over wireless ad hoc networks: energy and delay analysis," in *Proceedings of the International Conference on Ultra Modern Telecommunications and Workshops*, IEEE, St. Petersburg, Russia, October 2009.

[3] J. Dedrick, V. Gurbaxani, and K. L. Kraemer, "Information technology and economic performance: a critical review of the empirical evidence," *ACM Computing Surveys*, vol. 35, no. 1, pp. 1–28, 2003.

[4] A. Amoroso, G. Marfia, and M. Roccetti, "Going realistic and optimal: a distributed multi-hop broadcast algorithm for vehicular safety," *Computer Networks*, vol. 55, no. 10, pp. 2504–2519, 2011.

[5] C. Caini, R. Firrincieli, D. Lacamera, and M. Livini, "Virtualization technologies for DTN testbeds," in *Proceedings of the 2nd International ICST Confernce on Personal Satellite Services (PSATS '10), Rome, Italy, February 2010*, pp. 276–287, Springer, 2010.

[6] Y. Ghamri-Doudane, "Towards urban vehicular networking," in *Advanced Infocomm Technology: 5th IEEE International Conference, ICAIT 2012, Paris, France, July 25–27, 2012. Revised Papers*, vol. 7593 of *Lecture Notes in Computer Science*, pp. 1–4, Springer, Berlin, Germany, 2013.

[7] C. E. Palazzi and A. Bujari, "A delay/disruption tolerant solution for mobile-to-mobile file sharing," in *Proceedings of the IFIP Wireless Days (WD '10)*, pp. 1–5, Venice, Italy, October 2010.

[8] C. E. Palazzi, F. Pezzoni, and P. M. Ruiz, "Delay-bounded data gathering in urban vehicular sensor networks," *Pervasive and Mobile Computing*, vol. 8, no. 2, pp. 180–193, 2012.

[9] M. Furini, "Mobile games: what to expect in the near future," in *Proceedings of the 8th International Conference on Intelligent Games and Simulation (GAME-ON '07)*, pp. 93–95, November 2007.

[10] N. Yu and Q. Han, "Context-aware community construction in proximity-based mobile networks," *Mobile Information Systems*, vol. 2015, Article ID 402705, 18 pages, 2015.

[11] A. Bujari, C. E. Palazzi, D. Maggiorini, C. Quadri, and G. P. Rossi, "A solution for mobile DTN in a real urban scenario," in *Proceedings of the IEEE Wireless Communications and Networking Conference Workshops (WCNCW '12)*, pp. 344–349, Paris, France, April 2012.

[12] L. Pelusi, A. Passarella, and M. Conti, "Opportunistic networking: data forwarding in disconnected mobile ad hoc networks," *IEEE Communications Magazine*, vol. 44, no. 11, pp. 134–141, 2006.

[13] S. Burleigh, A. Hooke, L. Torgerson et al., "Delay-tolerant networking: an approach to interplanetary internet," *IEEE Communications Magazine*, vol. 41, no. 6, pp. 128–136, 2003.

[14] D. Maggiorini, L. A. Ripamonti, A. Bujari, and C. E. Palazzi, "Evaluating design constraints for proximity-based games in a real urban topology," in *Proceedings of the IEEE International Conference on Multimedia and Expo Workshops (ICMEW '13)*, pp. 1–6, San Jose, Calif, USA, July 2013.

[15] D. Maggiorini, C. Quadri, and L. A. Ripamonti, "On the feasibility of opportunistic collaborative mixed reality games in a real urban scenario," in *Proceedings of the 21st International Conference on Computer Communications and Networks (ICCCN '12)*, pp. 1–5, August 2012.

[16] D. Maggiorini, C. Quadri, and L. A. Ripamonti, "Opportunistic mobile games using public transportation systems: a deployability study," *Multimedia Systems*, vol. 20, no. 5, pp. 545–562, 2014.

[17] N. Uchida, N. Kawamura, G. Sato, and Y. Shibata, "Delay tolerant networking with data triage method based on emergent user policies for disaster information network system," *Mobile Information Systems*, vol. 10, no. 4, pp. 347–359, 2014.

[18] U. G. Acer, P. Giaccone, D. Hay, G. Neglia, and S. Tarapiah, "Timely data delivery in a realistic bus network," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 3, pp. 1251–1265, 2012.

[19] M. Demmer and K. Fall, "DTLSR: delay tolerant routing for developing regions," in *Proceedings of the 1st ACM SIGCOMM Workshop on Networked Systems for Developing Regions (NSDR '07)*, pp. 1–6, Kyoto, Japan, August 2007.

[20] C. E. Palazzi and A. Bujari, "Social-aware delay tolerant networking for mobile-to-mobile file sharing," *International Journal of Communication Systems*, vol. 25, no. 10, pp. 1281–1299, 2012.

[21] M. Di Felice, L. Bedogni, and L. Bononi, "Dynamic backbone for fast information delivery in vehicular ad hoc networks: an evaluation study," in *Proceedings of the 8th ACM Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks (PE-WASUN '11)*, pp. 1–8, ACM, November 2011.

[22] G. Marfia, M. Roccetti, A. Amoroso, and G. Pau, "Safe driving in LA: report from the greatest intervehicular accident detection test ever," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 2, pp. 522–535, 2013.

[23] P. Hui, J. Crowcroft, and E. Yoneki, "BUBBLE rap: Social-based forwarding in delay tolerant networks," in *Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '08)*, pp. 241–250, ACM, Hong Kong, May 2008.

[24] A. Vahdat and D. Becker, "Epidemic routing for partially-connected ad hoc networks," Tech. Rep., 2000.

[25] A. Lindgren, A. Doria, and O. Schelén, "Probabilistic routing in intermittently connected networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 7, no. 3, pp. 19–20, 2003.

[26] K. Fall, "A delay-tolerant network architecture for challenged internets," in *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM '03)*, pp. 27–34, Karlsruhe, Germany, August 2003.

[27] A. Pentland, R. Fletcher, and A. Hasson, "DakNet: rethinking connectivity in developing nations," *Computer*, vol. 37, no. 1, pp. 78–83, 2004.

[28] C. T. De Oliveira, R. B. Braga, D. M. Taveira, N. C. Fern, and O. C. M. B. Duarte, "A predicted-contact routing scheme for Brazilian rural networks," in *Electraical Engineering Program*, COPPE/UFRJ, 2008.

[29] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, "MaxProp: routing for vehicle-based disruption-tolerant networks," in *Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM '06)*, pp. 1–11, April 2006.

[30] X. Zhang, J. K. Kurose, B. N. Levine, D. Towsley, and H. Zhang, "Study of a bus-based disruption-tolerant network: mobility modeling and impact on routing," in *Proceedings of the 13th Annual ACM International Conference on Mobile Computing and Networking (MobiCom '07)*, pp. 195–206, Montreal, Canada, September 2007.

[31] N. Banerjee, M. D. Corner, and B. N. Levine, "Design and field experimentation of an energy-efficient architecture for DTN throwboxes," *IEEE/ACM Transactions on Networking*, vol. 18, no. 2, pp. 554–567, 2010.

[32] J. G. Jetcheva, Y.-C. Hu, S. PalChaudhuri, A. K. Saha, and D. B. Johnson, "Design and evaluation of a metropolitan area multitier wireless ad hoc network architecture," in *Proceedings of the 5th IEEE Workshop on Mobile Computing Systems and Applications (WMCSA '03)*, pp. 32–43, Monterey, Calif, USA, October 2003.

[33] M. Sede, X. Li, D. Li, M.-Y. Wu, M. Li, and W. Shu, "Routing in large-scale buses ad hoe networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '08)*, pp. 2711–2716, April 2008.

[34] S. Ahmed and S. S. Kanhere, "Cluster-based forwarding in delay tolerant public transport networks," in *Proceedings of the 32nd IEEE Conference on Local Computer Networks (LCN '07)*, pp. 625–634, Dublin, Ireland, October 2007.

[35] S. Gaito, D. Maggiorini, G. P. Rossi, and A. Sala, "Bus switched networks: an ad hoc mobile platform enabling urban-wide communications," *Ad Hoc Networks*, vol. 10, no. 6, pp. 931–945, 2012.

[36] Google Transit, http://transit.google.com/.

[37] C. Quadri, D. Maggiorini, S. Gaito, and G. P. Rossi, "On the scalability of delay-tolerant routing protocols in urban environment," in *Proceedings of the IFIP Wireless Days (WD '11)*, pp. 1–6, IEEE, Niagara Falls, Canada, October 2011.