A Peer-to-Peer Overlay System for Message Delivery in Wide Intermittently-Connected Hybrid Networks

Armel Esnault, Nicolas Le Sommer, Frédéric Guidec

A Peer-to-Peer Overlay System for Message Delivery in Wide Intermittently-Connected Hybrid Networks

Armel Esnault, Nicolas Le Sommer and Frédéric Guidec

IRISA Laboratory, Université de Bretagne-Sud
Bâtiment ENSIBS, Campus de Tohannic
56017 Vannes Cedex, France
{Armel.Esnault, Nicolas.Le-Sommer, Frederic.Guidec}@univ-ubs.fr

Abstract. With the emergence of the Internet of Things, billions of new devices will be wirelessly-connected to the Internet in the next decade, thus yielding a growth of the data traffic, especially in the network infrastructures maintained by mobile operators. Intermittently-connected hybrid networks (ICHNs), which combine an infrastructure part and loosely-connected mobile ad hoc parts, offer interesting perspectives to cope with the growing data traffic.
This paper presents a decentralized unstructured peer-to-peer overlay system that aims at supporting communications in ICHNs deployed in wide geographical areas. It also presents the simulation results we obtained for our system for an ICHN composed of a few hundreds of fixed stations (with Internet access) and thousands of mobile devices used by people to exchange data while roaming a medium-size city.

Keywords: *opportunistic networking, intermittently-connected mobile ad hoc networks, peer-to-peer overlay*

1 Introduction

Every day, billions of people use their smartphones to access the Internet either through a 3G, a 4G or a Wi-Fi connection. With the evolution of the Internet towards an Internet of Things, the number of communicating devices in our daily environment will increase tremendously in the next decade, thus entailing a growth of the data traffic generated by these many devices in the future. This growth will undoubtedly be more important in the networks deployed by mobile operators, because people will want to access and to control their things from anywhere at anytime using their smartphones or their tablets. Yet the spectrum available for mobile data traffic is limited, and recent radio access technologies (such as LTE) are reaching the limits of Shannon's law. Progress made at the physical layer to increase the capacity of cellular networks may therefore prove insufficient in the future. One of the most relevant solution in terms of cost-effectiveness and deployment facilities is based on Wi-Fi mesh networks.

Several mesh networking projects, such as RoofNet [1], Serval [5], OpenGarden[1] and and Commotion[2] have shown that it is possible to provide nomadic users in cities

[1] http://opengarden.com/
[2] https://commotionwireless.net

with broadband multi-hop connectivity to the Internet, thanks to a set of Wi-Fi access points acting as mesh routers and forming a backbone infrastructure with a limited number of wired links to the Internet. Yet, these mesh networks have some limitations. When such networks involve mobile devices, and when these devices are distributed sparsely or irregularly in the environment, network partitions can occur. Maintaining end-to-end connectivity between all network nodes then becomes quite a challenge, as dynamic routing protocols designed especially for MANETs (Mobile Ad hoc NETworks), such as OLSR, B.A.T.M.A.N. or AODV, prove unable to ensure message forwarding in such conditions.

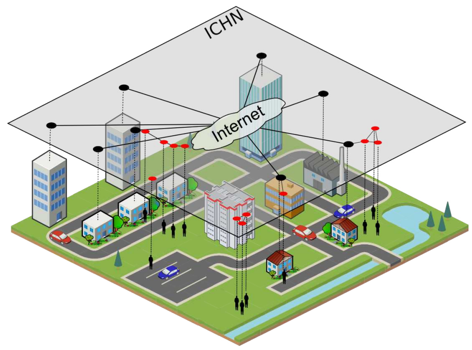


Fig. 1: Illustration of an intermittently-connected hybrid network (ICHN).

An interesting evolution of mesh networks is what we refer to as intermittently-connected hybrid networks (ICHNs), which unlike traditional mesh networks can tolerate network partitions. In an ICHN, communications rely on the "store, carry and forward" principle. This principle is the foundation of Opportunistic Networking [3]. The basic idea is to take advantage of radio contacts between devices to exchange messages, while exploiting the mobility of these devices to carry messages between different parts of the network. Two devices can thus communicate even if there never exists any temporaneous end-to-end path between them. Recent experiments conducted in real conditions have shown that applications that do not require end-to-end routes to work, such as voice-messaging, e-mail, or data sharing, can indeed perform quite satisfactorily in networks that rely on this principle [13,4,6,15,10].

We argue that ICHNs can be an interesting option for mobile operators to cope with the growth of the data traffic inherent in the emergence of the Internet of Things, even for Internet service providers or local authorities to provide nomadic people with ubiquitous services without resorting to any expensive infrastructure equipment. In this paper, we present Nephila, a decentralized and unstructured peer-to-peer overlay system we have designed to support communications in a wide ICHN, such as that illustrated in Figure 1. Nephila offers an homogeneous view of the network by hiding the connectivity disparities between mobile and fixed devices. It implements scalable mechanisms to perform neighbor discovery, and it provides a utility-based function to define message forwarding strategies. The "store, carry and forward" principle is used to deal with connectivity disruptions.

The remainder of the paper is organized as follows. Section 2 presents works pertaining to communication in intermittently-connected networks, and shows their limitations for wide ICHN. Section 3 gives an overview of Nephila, and details some of its salient features. Section 4 describes the neighbor discovery process implemented in Nephila. Section 5 shows how message forwarding paths are computed in Nephila. Section 6 presents evaluation results. Nephila is here compared with two other well-known opportunistic forwarding protocols: Spray-and-Wait [14] and PRoPHET [9]. Finally, Section 7 summarizes our contribution, and gives perspectives for future work.

2 Related work

Communications in intermittently-connected networks composed only of mobile devices or of both stationary and mobile devices have been studied in research works dealing with delay/disruption-tolerant networking and opportunistic networking [18]. These works implement the ”store, carry and forward” principle and devise several strategies and optimizations to control the replication of messages, and thus to avoid an epidemic dissemination of messages in the network [16]. The Spray-and-Wait routing protocol [14] provides a drastic way to reduce the number of message copies that are disseminated in the network. It relies on a two-phase message delivery mechanism. During the spray phase, the source node and first carriers of the message spread multiple copies of the same message over the network. Then, in the wait phase, each relay node maintains its copy in its cache and eventually delivers it to the final destination if it encounters this one. Nevertheless, this protocol relies on a random-based forwarding, and as such is not really suited to route messages efficiently in wide ICHNs. In order to improve message delivery in intermittently-connected networks, several research works have investigated deterministic approaches. For instance, the Message Ferrying approach [19] exploits the non-randomness mobility behavior of specific nodes (the message ferries) to carry messages between the different partitions of the network with the aim of increasing the delivery rate and of reducing the message propagation delays.

Some works go beyond the Message Ferrying approach by considering the recurrent social mobility patterns of users and by investigating solutions relying on probabilistic and prediction techniques in order to choose among a set of nodes the best carrier(s) for each message that must be forwarded. The probabilities and the predictions of future contacts are computed using context properties and history of contacts. PRoPHET [9] takes it routing decisions according to the delivery probabilities it computes based on the frequency of encounters. When a node wants to forward a message to another node, it looks for the neighbor node that has the highest frequency of direct or transitive contacts with the destination. Where PRoPHET relies on the history of encounters, the Context-Aware Routing protocol (CAR) [11] exploits some contextual properties using utility functions and Kalman filters to compute the message delivery estimation. HiBOp [2] and Propicman [12] investigate a social and history-based approach in order to predict the future movements of the nodes and to route the messages according to this prediction. HiBOp and Propicman can be considered as more general than CAR, because unlike the latter, they do not suppose the existence of an underlying routing protocol, and they can process context information provided by the message emitters in order to take their routing decisions.

History-based algorithms can offer a good performances in terms of delivery ratio and delivery delay, but have the major drawback of assuming that a node is able to store locally a large amount of contact information. Such an assumption is somewhat optimistic since the majority of the devices forming ICHN are small portable devices with limited capabilities and capacities. Processing contact histories and predicting movement patterns is also a tricky problem, especially in environments composed of numerous mobile devices that move following irregular patterns, such as those carried by pedestrians in a city. Moreover, unlike the RoofNet, Commotion, Serval and OpenGarden projects, the above-mentioned works focus only on communication between mobile devices, and they do not exploit the communication capacities offered by a mesh networking infrastructure. This point can be considered as a limitation to support communication between mobile devices in wide ICHNs. However it must be noticed that the disruption-tolerant communication solutions proposed in these works are more efficient than those implemented in projects Commotion and Serval (RoofNet and OpenGarden projects do not offer such communication means). Indeed, in Serval and Commotion, the text messaging and file sharing applications are developed over a file distribution system that implements an epidemic file dissemination model based on the ”store, carry and forward” principle in order to tolerate the communication disruptions. In Commotion and Serval, no precautions are taken to limit the number of messages that are exchanged and forwarded by the devices. Messages or files are thus disseminated in the network and replicated on every encountered node. Consequently in in high-density regions, some network congestions can happen.

In the next section, we present a peer-to-peer overlay system that provides efficient communications between devices in wide ICHNs by tolerating communication disruptions and network partitions, and by exploiting mesh network infrastructures.

3 Overview of Nephila

An ICHN can be complex, can cover a wide area (e.g., a city), and can be composed of numerous fixed and mobile devices (see Figure 1). Mobile devices can be carried and used by people, or can be embedded in vehicles. Fixed devices can be connected to the Internet or to corporate intranets through wired or wireless links. Our approach aims at proposing a versatile system. We do not make any assumption about the management and the location of fixed devices, about the global number of devices in the network, or about how mobile devices move in the network. Mobile devices can therefore be distributed irregularly in the environment. Fixed devices can be operated by mobile operators, by local authorities, or they can be simply managed by private companies or citizens (e.g., residential gateways). Mobile and fixed devices can appear in, and disappear from, the network at any time.

Nephila is a decentralized and unstructured peer-to-peer overlay system that offers an homogeneous view of wide ICHNs (see Figure 2), and that is meant to support the communications between devices in these networks. It implements a set of mechanisms and provides several functions that allows to devise various event-based message

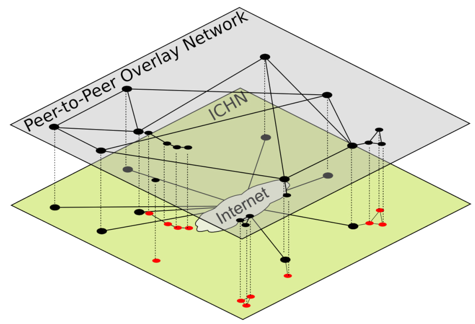


Fig. 2: A peer-to-peer overlay system for an ICHN.

forwarding strategies in a simple way. The main mechanisms and functions are the following:

Neighbor discovery Nephila performs a proactive discovery of fixed and mobile nodes. It can provide each node with a list of its neighbors, and can notify these nodes about any change in their neighborhood. This discovery process is an original feature of Nephila, and consequently it is detailed afterward in a specific section.

Backbone management Nephila creates and maintains logical links between devices that act as gateways between the infrastructure part and loosely-connected mobile ad hoc parts of the IHCN in order to form a backbone (see Figure 2). This backbone helps cover a wide area and support communications between numerous mobile devices. The scalability of our system mainly resides in the existence of this backbone.

Disruption tolerant networking The messages received from the local applications or from the neighbor nodes are stored locally in a cache until they expire, or until they are delivered directly to their destination. When a mobile node encounters another one, they can exchange the messages they have in their local cache, thus implementing the ”store, carry and forward” principle, and supporting the connectivity disruptions and the network partitions.

Computation of message forwarding paths In order to devise efficient message forwarding algorithms, Nephila provides a list of so-called ”trail values” (TV). The trail value of a node reflects its capacity to reach a given destination, either directly or through intermediate nodes. The computation and the dissemination scheme of trail values are key features of our proposition, and consequently they are described more precisely later in Section 5.

Message forwarding In Nephila, the message forwarding can be conditioned by a preliminary exchange of summary vectors. A vector includes the ID of messages a node has locally, and for which it considers the neighbor with which it performs an exchange as a best forwarder than itself based on the trail values of this one. When a node receives such a summary vector, it compares the ID of the messages it has locally with those contained in the summary vector, and request the message(s) that it does not have yet.

Event-based message forwarding algorithms Several events are generated in Nephila, such as the arrival of a new neighbor, the disappearance of a neighbor, the reception of trail values from neighbors, the reception of summary vectors and the reception of requests for a message (or a list of messages). Based on these events, several message forwarding strategies can be easily devised in Nephila. Currently, Nephila implements a message forwarding algorithm that exploits all the above-described features and implements the ”store, carry and forward” principle. This algorithm is called BTSA (for Best Trail Selection Algorithm). With BTSA, each mobile node receiving a message is expected to forward a copy of this message to one or several of its neighbors. When no forwarding opportunity exists (e.g., no neighbor is available, or all current neighbors are considered as being unsuitable as carriers for this message) the node stores the message and waits for future contacts with other devices. A message can thus be delivered to a destination node, even if the source and destination never get close to each other, and even if they are never present simultaneously in the network. In addition with BTSA, each node takes a local decision based on its own trail values and on those sent by its neighbors. When a node receives a message from a neighbor node, or from a local application, it forwards this message to the neighbor that has the greatest trail value for the destination, provided that this value is greater than its own value. Sometimes it might be risky to forward message copies exclusively to good carriers, especially in sparse network where the contact with a good carrier is uncertain. To address this issue BTSA has a stock of a few number of copies dedicated to bad neighbors. This number of copies is limited in order to avoid network overload and resource consumption on mobile devices.

4 Neighbor discovery

In an ICHN, it is difficult to forward the messages towards their destinations based on the IP address of the devices, especially in the ad hoc parts of the network due to the mobility of the nodes. In order to address this issue, Nephila assigns to each node a unique ID at the installation time, and forwards the messages according to these IDs. The neighbor discovery mechanisms maintain a mapping between the ID of nodes and their IPv6 address. It must be noticed that the IPv6 address of a node is only known by the neighbors of this one.

Nephila implements two distinct neighbor node discovery mechanisms: one for the mobile ad hoc parts of an ICHN, and another one for the backbone part of this ICHN. The neighbor discovery of mobile devices is achieved by a standard beaconing system, while in backbone the neighbors of nodes are discovered using a peer sampling service based on Cyclon [17]. With Cyclon, the nodes forming the backbone exchange each other lists of neighbors periodically. In order to initiate the discovery we assume, in the current implementation of this service, the existence of a limited set of fixed nodes. This set is a parameter of Nephila. It is used by Cyclon in order to discover the nodes that are always present in the backbone. Cyclon allows to define the maximum size of the list of neighbors exchanged periodically by the nodes. Thus, each fixed node only knows a limited number of the nodes forming the backbone, thus reducing the computation time needed to evaluate the next suitable message forwarder(s). Cyclon provides a fast dissemination of routing tables thanks to gossiping techniques.

5 Computation of forwarding paths

The trail values computed by Nephila are meant to be used by a forwarding algorithm in order to select, for each message, the most relevant next forwarder(s). The nodes share their respective trail values with their neighbors through the discovery process. The values are included in beacon messages, and are provided by the Cyclon-based service we have developed. Disseminating such pieces of information can be costly in terms of traffic because the nodes can have trail values for a large number of devices. In order to address this issue, we use a modified version of the Exponential Decay Bloom Filter (EDBF [8]). EDBF is an extension of the traditional Bloom filter. EDBF allows to encode probabilistic forwarding tables in a highly compressed manner. It was originally designed to store and propagate in unstructured peer-to-peer networks information about content hosted in the neighborhood of a node. EDBF has been coupled with a query routing mechanism in order to locate content in such networks. The modified version of the EDBF we designed allows to store and disseminate efficiently the trail values of each node in the overlay system. This version of EDBF will be called TBF (for Trail Bloom Filter) in the remainder of this paper. Based on the values included in TBFs, a message can reach its destination in a small number of hops and with a high probability.

A TBF is defined as an array of m floating point elements noted $[tbf_0,...,tbf_{m-1}]$. It uses k hash functions (assumed to be independent) noted $h_0,h_i,...,h_{k-1}$. Hash functions are used to insert trail values of nodes into a TBF. These function are also used to retrieve a trail value from a TBF. When inserting a given node x in a TBF, all k functions $h_0,h_i,...,h_{k-1}$ are evaluated simultaneously over x and the floating point elements $tbf_0,...,tbf_{m-1}$ in the TBF indexed by $h_i(x), with\, i \in [0,m-1]$ are set to the trail value of x. A query for node y in a TBF looks at the floating elements in the TBF indexed by $h_i(y), with\, i \in [0,m-1]$ and returns the trail value for node y.

In order to take into account the loss of confidence in the routing information (i.e., in trail values) over time and hops, a function *decay* is applied by a node on its TBF periodically. When a node detects the disappearance of one of its neighbors, it deletes the TBF it received from this neighbor previously, and it applies the *decay* function on its local TBF in order to reflect the loss of capacity to forward messages to this node. When a node detects another node in its neighborhood, it calls the *reinforce* function on its local TBF. This function sets to the maximum trail value (i.e., to 1), the value of all floating point elements $tbf_0,...,tbf_{m-1}$ returned by the hash functions for this new neighbor node. When a node receives the TBF of a neighbor, it stores this TBF locally and it merges a decayed version of this TBF with its own TBF. The periodic decaying applied on the local TBF is compensated for each neighbor by the frequent invocation of the *reinforce* function triggered by the discovery process. Function *merge* keeps the maximum of each element of two TBF tbf and $rtbf$, where tbf is the local TBF structure and where $rtbf$ is a TBF sent by a neighbor node.

In order to retrieve the trail value for a given node from a TBF, we have defined a function called *query*. By applying this function on its own TBF and on the TBFs received from its neighbors, and by comparing the values returned by this function, a node can determine if it is a better forwarder than its neighbors, or otherwise can select the best one among them. This function *query* can be assimilated to the utility-based functions introduced by Musolesi and Mascolo, in [11]. Functions *decay*, *reinforce, merge* and *query* rely on the hash functions associated with the TBF. These functions are defined as follows:

Parameters:
df is the decaying factor
n is the identity of a node
tbf TBF of the local node or a remote node
$rtbf$ TBF of a remote node
mv is a minimum trail value
Functions:
$decay(tbf, df, mv) : tbf_i \leftarrow max(mv, tbf_i * (1 - df)); \forall i, 0 \leq i \leq m-1$
$reinforce(tbf, n) : tbf_{h_i(n)} \leftarrow 1; \forall i, 0 \leq i \leq k-1$
$merge(tbf, rtbf) : tbf_i \leftarrow max(tbf_i, rtbf_i); \forall i, 0 \leq i \leq m-1$
$query(tbf, n) : min(tbf[h_0(N_n)], tbf[h_1(N_n)], ..., tbf[h_{m-1}(N_n)])$

TBFs allow to propagate trail values transitively in the network. The trail values decrease proportionally with the number of hops and the time. The trail value for a given destination reachable in n hops is defined as $(1-df)^n * (1-df)^{\left\lfloor \frac{e-a}{p} \right\rfloor}$, where e and a are respectively the emission date of a TBF sent by a node located at $n-1$ hops and the reception date of a TBF sent by a neighbor node, and p the decaying period.

By assigning to the devices forming the backbone a default trail value greater than that assigned to mobile devices, one can favor the forwarding of messages via the backbone. Thus, the forwarding of a message whose destination is unknown at the emission time could all the same be triggered. The message will indeed be forwarded towards a device of the backbone, and will be stored in the latter until discovering the destination. It is expected that the nodes of the backbone will be more able to deliver the message than the initial emitter itself, especially when the destination is not in the close neighborhood of the emitter. Since this message will be stored, carried and forwarded by intermediate nodes, it can obviously be delivered directly by one of them if it encounters the destination.

6 Simulation and results

In order to evaluate the overlay system described in the previous sections, we have developed a prototype in Java, and we performed simulations using the ONE [7], a discrete event simulator written in Java. In this section, we present the scenario, the simulation setup we used, and the results we obtained.

6.1 Scenario and simulation setup

The scenario we consider in the simulations is meant to be as realistic as possible. It involves a number of pedestrians carrying smartphones who walk freely in the streets of the city of Vannes, a French city covering about 25 km^2. We assume that pedestrians move at a speed that varies between 1 m/s and 1.6 m/s, and that 100 pedestrians use their smartphone to communicate periodically. They send several messages, and then stop to use their smartphones. After a few minutes they repeat the same process. For the sake of comparison, the number of people using their smartphone is kept constant, while the total number of pedestrians increases. The smartphones of the other pedestrians only act as relays. We performed simulations with 100, 500, 1000 and 2000 pedestrians. The environment is additionally populated by either 0, 100 or 200 fixed access points. These access points are interconnected through the Internet, and form the backbone in Nephila. They were chosen among residential gateways deployed in the city of Vannes on the basis of geographic and random criteria. An illustration of this environment is given in Figure 3.

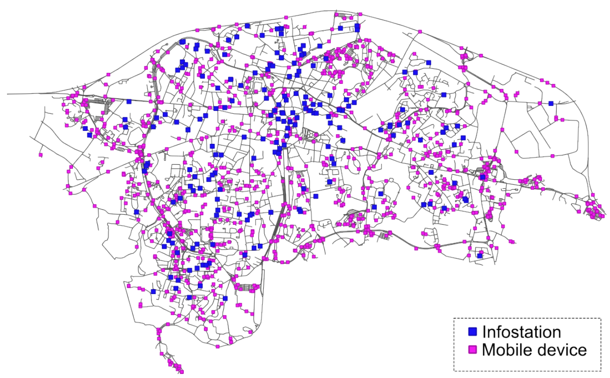


Fig. 3: Illustration of the simulation environment.

By considering an environment devoid of fixed access points, an environment populated with 100 fixed access points, and another one populated with 200 fixed access points, we want to evaluate the performances of Nephila (with BTSA), and the impact of the number of access points on the message delivery (i.e., the impact of the size of the backbone). In the simulation results presented hereafter, these three configurations will be referred as BTSA 0, BTSA 100 and BTSA 200 respectively.

The simulation duration is 1 hour and 20 minutes. During the first 10 minutes, both fixed and mobile devices perform a discovery of their neighbors, and compute and disseminate Trail Bloom Filters (TBFs). After this warm-up period, mobile devices start to send messages. The emission of messages is stopped 10 minutes before the end of the simulation. The other simulation parameters are defined in Tables 1 and 2. The Nephila's parameters were determined empirically through extensive experiments.

Parameter	Value
Size of the cache of messages of mobile devices	10 MBytes
Size of the cache size messages of access points	40 MBytes
Bitrate of wired links	20 Mbit/s
Bitrate of wireless interfaces	10 Mbit/s
Communication range of the wireless interface	50 meters
Number of hops	30
Maximum size of messages	200 kBytes

Table 1: Simulation parameters.

6.2 Simulation results

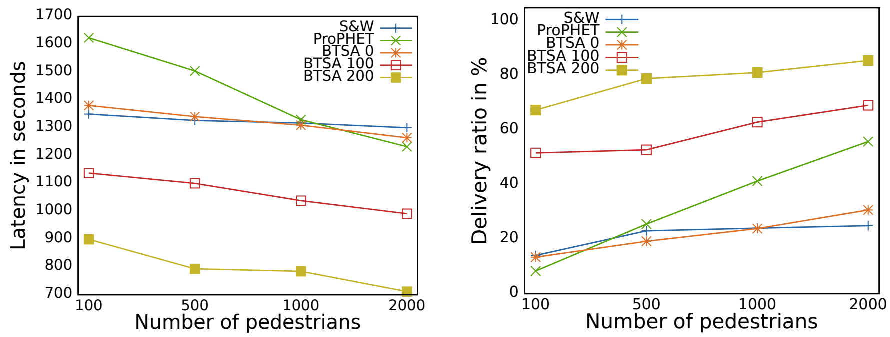
The objectives of these simulations is to know how our proposition works when only opportunistic communications are feasible, and to study the impact of the backbone and of the size of this one on the message delivery. For comparison purposes, we run in an environment devoid of fixed access points, the PRoPHET and the Spray-and-Wait protocols. A maximum of 10 copies of a given message can be replicated by Spray-and-Wait in our simulations. PRoPHET has been configured with its default parameters [9].

Parameter		Value
TBF	Decay period	20 s
	Decay factor	0.2
	Strengthen period	20 s
	Broadcast period	30 s
	Pedestrian min trail value	0.1
	Infostation min trail value	0.3
	Number of elements	600
	Number of hash functions	2
BTSA	Best threshold	0.1
	Number of copies	10
Cyclon	Neighborhood size	13
	Fanout	3
	Shuffling period	120

Table 2: Nephila's parameters.

The performances of Nephila (with BTSA) are compared with those of PRoPHET and Spray-and-Wait following three metrics, namely the average delay and the ratio of message delivery, and the network load. The delivery ratio is the number of messages delivered over the total number of messages sent by the 100 emitters. It reflects the ability of the protocol to eventually forward messages to their destination before they expired. It must be noticed that during a single simulation run, approximatively 2600 messages have been generated by the 100 pedestrians. The network load is the overall number of messages exchanged between nodes during the simulation. On the wireless parts of the ICHN, only messages carrying data related to the protocols were counted, because the beaconing process is not implemented in the versions of PRoPHET and Spray-and-Wait in the ONE simulator.

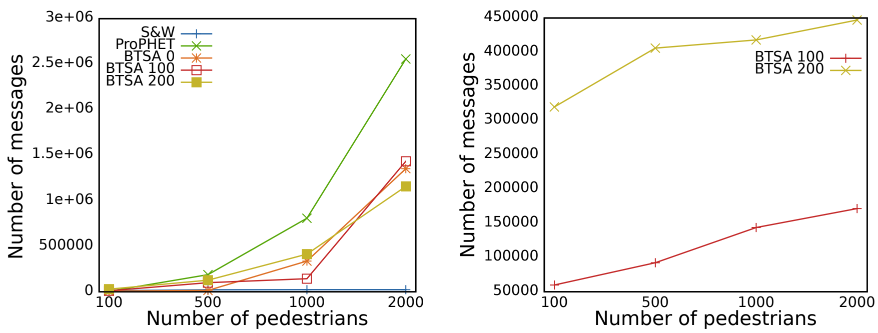
Figures 4a and 4b show, according to the number of pedestrians involved in the simulation, the average delay and the percentage of messages that have been delivered successfully. The average number of hops needed to forward a message to its destination



(a) Message delivery delay.

(b) Message delivery ratio.

Fig. 4: Message delivery performances.



(a) Number of messages forwarded in the wireless parts of the network.

(b) Number of messages forwarded in the backbone.

Fig. 5: Number of messages forwarded in wireless and wired parts of the network.

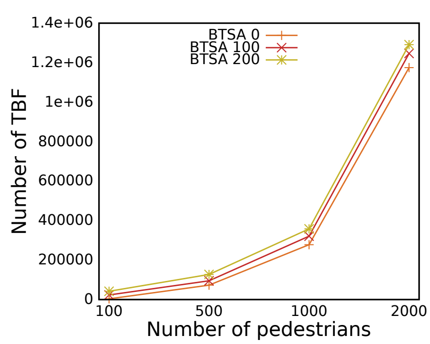


Fig. 6: Amount of TBF exchanged in the wireless parts of the network.

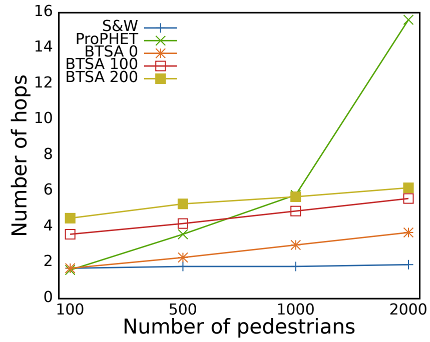


Fig. 7: Number of hops needed to reach the destination.

is given in Figure 7. Figures 5a, and 5b show the number of messages disseminated in the wireless and wired parts of the network respectively.

BTSA 0 (BTSA without access points) and Spray-and-Wait have very close performances in terms of delivery delay and ratio. However, where the number of copies remains constant with Spray-and-Wait, it increases with BTSA 0 according to the number of nodes in the network (see Figure 5a). Indeed, as we do not have assigned a time to live to messages, in order to evaluate only the behavior of our forwarding algorithm, the messages are replicated in the network even after the deliverance of the first copy to the destination. This number of messages can be reduced by assigning a time to live to messages and by implementing a healing mechanism in BTSA. Both BTSA 0 and Spray-and-Wait perform better than PRoPHET from 100 up to 500 pedestrians. The main reason lies in the fact that in sparse networks, the radio contacts are not frequent, and therefore the delivery probabilities estimated by PRoPHET on the basis of the frequency of encounters are not useful. With PRoPHET many messages are actually delivered to their destination by the initial emitters themselves. BTSA 0 and Spray-and-Wait do not have this limitation, as they can forward several copies of a message to "potentially bad" carriers. These carriers can in turn forward a copy of the message to other nodes, thus increasing the probability of encountering good carriers later. When the number of pedestrians increases, the number of radio contacts increases as well, and the probabilities of finding good carrier(s) increase accordingly. PRoPHET provides a better delivery ratio than BTSA 0 and Spray-and-Wait from 1000 up to 2000 pedestrians, but with the major drawback of increasing the number of copies disseminated in the network. Spray-and-Wait does not benefit as much as PRoPHET of the increase of the number of pedestrians because the number of copies that can be disseminated remains limited. The delivery ratio remains relatively constant due to the limited number of copies disseminated in the network (see Figure 5a).

Figure 6 shows the overhead induced by the TBF exchanges in the wireless parts of the network. This overhead depends of the number of contacts and of their duration. The overhead is thus more important in a dense network than in a sparse one. In the simulation we made for 2000 pedestrians, the average of the additional amount of data generated by each device of the wireless part of the ICHN during the simulation is about 2Kbits/s per nodes. In the wired part, the amount of data related to the TBF exchanges depends of the number of access points. It is about $3.2 * 10^8$ bytes for BTSA 100 and about $6.6 * 10^8$ bytes for BTSA 200. It represents on average 770bits/s per fixed node.

Figures 4a and 4b show the benefit provided by the backbone. This one allows to drastically reduce the message delivery delay and to strongly increase the delivery ratio without increasing the number of messages disseminated in the wireless parts of the network. A greater number of fixed nodes allows to cover a wider geographical area, and therefore to increase the probability of delivering a message to a remote destination, as well as to reduce the delivery delay. When the number of radio contacts increases, this effect is attenuated gradually. Figure 5b shows that an important amount of messages is transmitted through the backbone.

7 Conclusion and future work

In this paper we have presented Nephila, a system that can support opportunistic message forwarding in wide intermittently-connected hybrid networks (ICHN). Nephila builds and maintains a peer-to-peer overlay structure that interconnects fixed nodes at reasonable cost, while relying on the ”store, carry and forward” principle to tolerate connectivity disruptions on mobile nodes. It implements a scalable neighbor node discovery mechanism, and defines a utility-based function that is meant to help select the next message forwarders based on so-called trail values. These values reflect the capacity of a node to reach a given destination directly, or via intermediate nodes. In this paper, we have also compared the performance of our system with those of protocols Spray-and-Wait and Prophet. Simulation results confirm that communications between mobile devices in wide ICHN can indeed be achieved with good performances, using a limited number of fixed interconnected access points.

In the future, we plan to implement Nephila on Android-based smartphones and tablets, and on Raspberry Pi-based access points. These devices will be used to run experiments in real conditions. We also plan to devise new forwarding algorithms that can take into account contextual properties such as the power budget, location, speed, and trajectories of the mobile devices.

References

1. Aguayo, D., Bicket, J., Biswas, S., Judd, G., Morris, R.: Link-level measurements from an 802.11b mesh network. In: Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications. pp. 121–132. SIGCOMM ’04, ACM, New York, NY, USA (2004)
2. Boldrini, C., Conti, M., Jacopini, J., Passarella, A.: HiBOp: a History Based Routing Protocol for Opportunistic Networks. In: Conti, M. (ed.) International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM 2007). pp. 1–12. IEEE CS, Helsinky, Finland (2007)
3. Conti, M., Giordano, S.: Multihop Ad Hoc Networking: The Reality. IEEE Communications Magazine 45(4), 88–95 (Apr 2007)

4. Farrell, S., McMahon, A., Weber, S., Hartnett, K., Lynch, A., Meehan, E.: Report on DTN Applications During Arctic Summer 2010 Trial. In: 1st International Workshop on Opportunistic and Delay/Disruption-Tolerant Networking (Oct 2011)
5. Gardner-Stephen, P., Palaniswamy, S.: Serval mesh software-wifi multi model management. In: Proceedings of the 1st International Conference on Wireless Technologies for Humanitarian Relief. pp. 71–77. ACM (2011)
6. Islam, M.T., Turkulainen, A., Kärkkäinen, T., Pitkänen, M., Ott, J.: Practical Voice Communications in Challenged Networks. In: 1st Extreme Workshop on Communications (Aug 2009)
7. Keränen, A., Ott, J., Kärkkäinen, T.: The ONE Simulator for DTN Protocol Evaluation. In: SIMUTools '09: Proceedings of the 2nd International Conference on Simulation Tools and Techniques. ICST, New York, NY, USA (2009)
8. Kumar, A., Xu, J., Zegura, E.: Efficient and scalable query routing for unstructured peer-to-peer networks. In: INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE. vol. 2, pp. 1162–1173. IEEE (2005)
9. Lindgren, A., Doria, A., Schelén, O.: Probabilistic Routing in Intermittently Connected Networks. In: Proceedings of the The First International Workshop on Service Assurance with Partial and Intermittent Resources (SAPIR 2004). Fortaleza, Brazil (August 2004)
10. Mahéo, Y., Le Sommer, N., Launay, P., Guidec, F., Dragone, M.: Beyond Opportunistic Networking Protocols: a Disruption-Tolerant Application Suite for Disconnected MANETs. In: 4th Extreme Conference on Communication (ExtremeCom'12). pp. 1–6. ACM, Zürich, Switzerland (March 2012)
11. Musolesi, M., Mascolo, C.: CAR: Context-Aware Adaptive Routing for Delay Tolerant Mobile Networks. IEEE Transactions on Mobile Computing 8(2), 246–260 (2009)
12. Nguyen, H.A., Giordano, S., Puiatti, A.: Probabilistic Routing Protocol for Intermittently Connected Mobile Ad hoc Network (PROPICMAN). In: International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM 2007). pp. 1–6. IEEE CS, Helsinky, Finland (June 2007)
13. Ntareme, H., Zennaro, M., Pehrson, B.: Delay Tolerant Network on Smartphones: Applications for Communication Challenged Areas. In: 3rd Extreme Workshop on Communications (Sep 2011)
14. Spyropoulos, T., Psounis, K., Raghavendra, C.S.: Spray and Wait: an Efficient Routing Scheme for Intermittently Connected Mobile Networks. In: 2005 ACM SIGCOMM workshop on Delay-tolerant networking (WDTN'05). pp. 252–259. ACM, Philadelphia, PA, USA (2005)
15. Tziouvas, C., Lambrinos, L., Chrysostomou, C.: A Delay Tolerant Platform for Voice Message Delivery. In: 1st International Workshop on Opportunistic and Delay/Disruption-Tolerant Networking. pp. 1–5 (2011)
16. Vahdat, A., Becker, D.: Epidemic Routing for Partially Connected Ad Hoc Networks. Tech. rep., Duke University (April 2000)
17. Voulgaris, S., Gavidia, D., Van Steen, M.: Cyclon: Inexpensive Membership Management for Unstructured P2P Overlays. Journal of Network and Systems Management 13(2), 197–217 (2005)
18. Zhang, Z.: Routing in Intermittently Connected Mobile Ad Hoc Networks and Delay Tolerant Networks: Overview and Challenges. IEEE Communications Surveys and Tutorials 8(1), 24–37 (January 2006)
19. Zhao, W., Ammar, M., Zegura, E.: A Message Ferrying Approach for Data Delivery in Sparse Mobile Ad Hoc Networks. In: Proceedings of ACM Mobihoc 2004. Tokyo Japan (May 2004)