



Article

# Evaluating Forwarding Protocols in Opportunistic Networks: Trends, Advances, Challenges and Best Practices

Vishnupriya Kuppusamy \*, Udaya Miriya Thantrige, Asanga Udugama  
and Anna Förster

Sustainable Communication Networks, University of Bremen, 28359 Bremen, Germany;  
mperer4@uwo.ca (U.M.T.); adu@comnets.uni-bremen.de (A.U.); anna.foerster@comnets.uni-bremen.de (A.F.)

\* Correspondence: vp@comnets.uni-bremen.de

Received: 29 March 2019; Accepted: 4 May 2019; Published: 11 May 2019



**Abstract:** A variety of applications and forwarding protocols have been proposed for opportunistic networks (OppNets) in the literature. However, the methodology of evaluation, testing and comparing these forwarding protocols are not standardized yet, which leads to large levels of ambiguity in performance evaluation studies. Performance results depend largely on the evaluation environment, and on the used parameters and models. More comparability in evaluation scenarios and methodologies would largely improve also the availability of protocols and the repeatability of studies, and thus would accelerate the development of this research topic. In this survey paper, we focus our attention on *how* various OppNets data forwarding protocols are evaluated rather than *what* they actually achieve. We explore the models, parameters and the evaluation environments and make observations about their scalability, realism and comparability. Finally, we deduce some best practices on how to achieve the largest impact of future evaluation studies of OppNets data dissemination/forwarding protocols.

**Keywords:** opportunistic networks; forwarding; data dissemination; evaluation methodology; simulation; OMNeT++; ONE; Adyton

## 1. Introduction

A class of networks where the devices connect and exchange data whenever they come in contact are called opportunistic networks (OppNets). As the name suggests, these networks get formed whenever the opportunity arises. Since an OppNet is formed by the availability of the devices in the connection range of each other, device-to-device communication technologies such as Bluetooth, ZigBee and WiFi-direct are used for communication in opportunistic networks.

Although it seems straight-forward to deploy OppNets, real testbeds to study and evaluate OppNets are quite uncommon. A typical testbed for OppNets consists of real users with their mobility patterns, traffic generation, buffer management in each device, a data dissemination protocol and post processing methods to obtain the necessary metrics. Although these modules can be developed and integrated in testbeds, the main challenge is in the repeatability of the experimental setup. Simply put, the dynamics of the network and, especially, the real users' mobility makes it impossible to repeat the experiment in the same manner again. Thus, simulators are the most common way of evaluating OppNets as they support the flexibility and ease of implementation.

One of the approaches in bridging the gap between real world and simulation-based evaluations is to attempt to precisely capture the properties of a real-world scenario and reproduce it in a simulator. This is often done by capturing various types of real world traces, e.g., mobility or application traces.

A different, but not contradicting approach is to perform evaluations with a large range of scenarios, so that the evaluation has statistical significance. Moreover, benchmarks for OppNets do not exist to the best of our knowledge and, hence, the responsibility lies with the researchers to identify suitable evaluation configurations and experimental setups for evaluating their OppNet protocols. In this paper, we study the aspects of OppNet performance evaluations in simulators and how they match with the real-world expectations for OppNets. Further, we briefly discuss the challenges in conducting OppNet evaluations, and present a gap analysis with respect to the evaluation of OppNets protocols. It is important to note that we do not evaluate the discussed protocols; rather, we explore their evaluation methodologies to derive best practices for future evaluation studies.

A detailed analysis of Mobile Ad-hoc Networks (MANET) evaluations was presented by Kurkowski et al. [1] and this survey attempts to provide a similar analysis for OppNet evaluations. A number of relevant surveys on OppNets exists in the literature, which focuses on the protocols themselves, their taxonomy and classifications based on specific aspects such as group dynamics or parameter optimizations. A general overview and analysis of protocols, models and tools for OppNets were discussed by Mota et al. [2]. The evolution of OppNets research and challenges were presented by Trifunovic et al. [3]. Neighbor discovery approaches were explored in detail by Pozza et al. [4]. Routing protocols and data dissemination protocols were summarized by Chakchouk et al. [5] and Sobin et al. [6], whereas the impact of mobility models was discussed by Batabyal et al. [7]. Simulating OppNets, in terms of the scalability, realistic models and metrics, were analyzed by Dede et al. [8] and Mota et al. [2]. Further, an evaluation methodology was proposed by Garcia et al. [9] pertaining to simulation studies for Vehicular Ad-hoc Networks (VANETs). This paper complements these existing surveys by extending the focus to evaluation procedures used in OppNet literature.

The main goal of the paper is three-fold: firstly, we analyze the assessment environments and methodologies of OppNet protocol evaluations adopted in current simulators; secondly, we discuss how they align with the realistic considerations for OppNets; and, thirdly, we propose a range of scenarios that these evaluations can span to show the potential of OppNets. In all of the presented figures and evaluations, we base the analysis on what we found in the references, but the determination of realisms of the explored parameters and scenarios is a view provided by us.

This paper is organized as follows: Section 2 presents the OppNet applications and the methodology of the survey is discussed in Section 3. The different models relevant to an OppNet evaluation are discussed in separate sections: Section 4 focuses on the simulation environments used for OppNets, while comparative studies between protocols in evaluations are discussed in Section 5. Scalability of experiments is analyzed in Section 6 followed by mobility in Section 7 and cache and traffic in Section 8. Section 9 gives an overview about the metrics used in OppNet evaluation environments and Section 10 summarizes our insights. Section 11 concludes the paper.

## 2. Applications of Opportunistic Networking Protocols

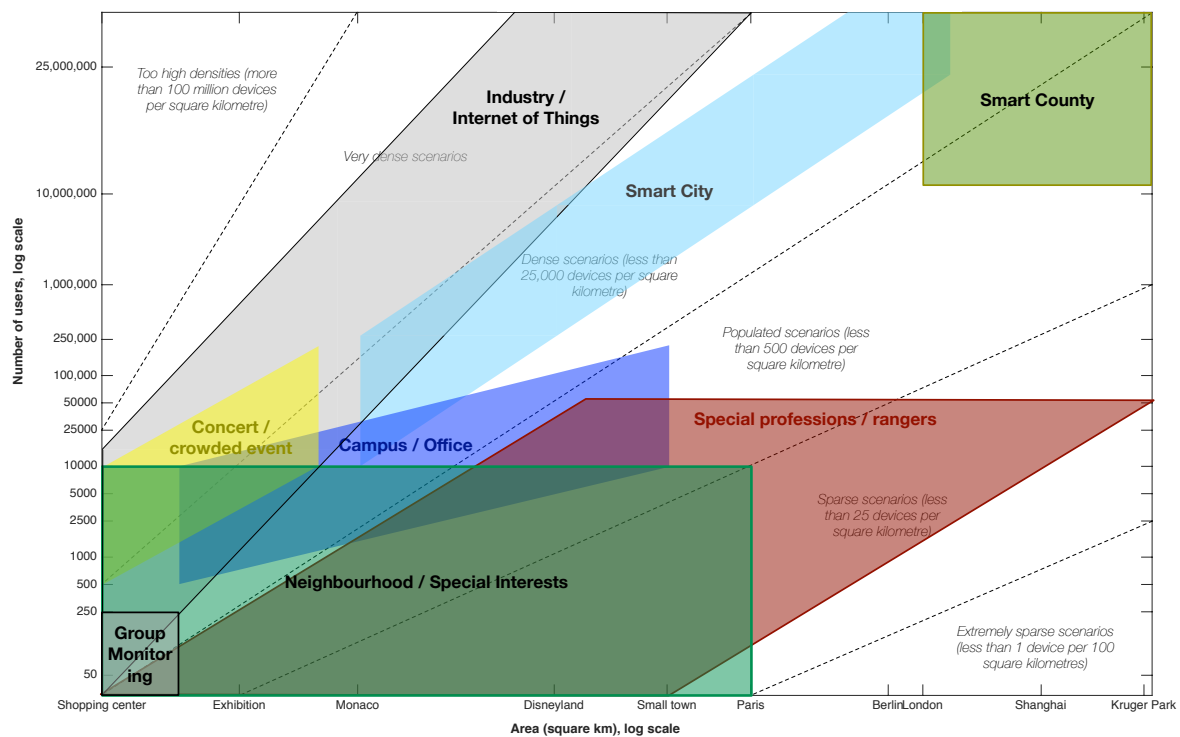
In this section, we first describe some of the notable application scenarios for OppNets. Then, we analyze them in terms of their node densities and size in general to extract relevant scenarios for evaluating forwarding protocols.

OppNets applications have already been studied several times, e.g., recently in [3]. There, several applications have been proposed and discussed, which are also included here. The applications have been used from the papers in this survey, however the papers do not mention specific parameters for the used applications. Our main contribution is that we identify their realistic density properties.

The following list of applications is not exhaustive. On the one hand, it cannot be exhaustive, because OppNets have not been leveraged broadly by real applications yet—who was able to predict applications such as Twitter or Facebook before cheap, affordable Internet connections emerged? On the other hand, we attempt to cover the most prominent and promising scenarios here. Along with the descriptions of the applications, we also discuss their estimated sizes, which are further summarized in Figure 1. We have considered the information in the publications to decide the density scale. All these

publications state the applications their protocols are relevant for, but do not necessarily identify the application specific parameters. We take this a step further by trying to identify the realistic density properties that can serve these application areas. Our judgment on realism is based on our observations from the publications/expectations in the respective application area.

**Group monitoring:** This is probably the simplest and most straight-forward application for OppNets. It refers to a group of students or tourists, who stroll around together through a city or museum. There are mainly two application goals: to identify if somebody is missing (not connected to the group any more) and to send information in real time to all members. This application is in fact a step between mobile ad-hoc networks and opportunistic networks. It was prototyped by Saloni et al. [10]. This application scenario can realistically scale up to 250 people at most and over a small geographical area. Even if the group moves, the relative area it covers remains small.



**Figure 1.** OppNets applications and their node densities. Six different node density areas have been identified and marked with dotted lines.

**Neighborhood/special interests:** This scenario unites two different applications. First, neighbors would like to exchange information about local events, things to share, problems, discussions, etc. In this case, a well defined group of people, living close to each other (but not always connected), exchange data. In the second version of this scenario, another group of well defined people, such as the members of a sporting club, exchange data with each other. In comparison to the neighbors, the members of these groups are more spread, but still in relative proximity of each other. Since the application assumes that people know each other at least coarsely, the maximum number of people even in a large city can reach no more than 10,000 people over an area of a city such as Paris. We assume that, if larger cities are targeted, the application would rather split into several geographically isolated areas.

**Special professions/rangers:** This application resembles the last one, but assumes that the people are brought together by their special profession or role, e.g., firefighters, policemen or rangers. The number of such people is not random and depends on the population or area. Thus, the node density tends to be more stable or inflexible. This can be clearly seen in Figure 1 (red area). Another typical property of this scenario is that it has rather sparse node densities.

Campus/office: In this scenario, people working together exchange information, e.g., in an office or campus environment. In addition, here we can observe some dependency between area and number of nodes, but the general node density is higher than for the rangers scenario.

Concert/crowded event/disaster: Here, a crowded event such as a soccer game or concert is assumed. Such scenarios can be very dense, e.g., more than 50,000 people can fit into a large stadium. Of course, space is limited and some dependency between area and number of people/nodes inside can be observed. This scenario also includes the disaster case, as usually many people in small areas are affected and panic can cause further density.

Smart City: Here, people and devices in a city are connected together, e.g., street lights, traffic lights, parking lots, vehicles and pedestrians. These scenarios are relevant when the relative density is quite high to cover the complete city. From experience, smart city concepts are applied mostly to bigger cities with larger populations than to smaller towns.

Smart County: This scenario extends the Smart City application to even larger areas, such as San Francisco Bay Area or Berlin/Brandenburg Metropolitan Area. The relative density is similar or lower than for Smart Cities, as the county includes also less-populated areas. However, the absolute area and number of devices/people are greater.

Industry/Internet of Things: Another significant scenario for OppNets is the internet of things or Industry 4.0. Here, millions of devices are installed in relatively small areas (e.g., factories or warehouses) to fully control industrial processes. In this scenario, fewer people and more devices are present, which makes the relative density very high.

Disaster alert: This scenario is probably the one most referenced to in OppNets. Here, potentially many people over large areas need to be warned about an upcoming danger. The areas can be potentially as big as Kruger park (e.g., a volcano eruption, earthquake or big storm) and as small as a neighborhood (car accident). The number of people is also unlimited—from few people around an accident to the whole population of California. This application is not shown in Figure 1, since it would cover the complete area.

*Take-home message:* Similar to the conclusions in [3], we summarize that there are many different applications with great potential in OppNets, which cover different absolute areas and number of people/devices and relative densities. However, there are also some areas in Figure 1, which we consider non-realistic, i.e., very high densities (over 100 million devices per square kilometer) and very low densities (fewer than one device per square kilometer). While such scenarios are interesting from scientific point of view, experiments need to consider also the more realistic ones.

### 3. Methodology of the Survey

In this survey, we focus mainly on evaluation methodologies for OppNet protocols than on their details or taxonomies. We are interested in how they have been evaluated and compared to other protocols and whether there are some trends to be observed in the last five years.

The selection of the papers involved in this study are based on two factors: (i) The papers should be indexed in Scopus to ensure that they are peer-reviewed research works. (ii) The papers should describe pure opportunistic forwarding approaches. To be precise, we did not consider ad-hoc networks or protocols involving a combination of ad-hoc and infrastructure networks. The papers span from 2000 to 2018. Our list of papers is presented in Table 1 and the data from the papers used in this survey are presented in Appendix B. We do not claim to be exhaustive in our survey, simply because there are many works in this area and not all of them clearly position themselves in the opportunistic networks field.

The OppNets evaluation methodology includes typically the following components:

- Simulation environment or frameworks (Section 4)
- Comparative studies—which protocol has been compared to which others (Section 5)
- Network densities (Section 6)
- Mobility models (Section 7)

- Cache sizes and traffic (Section 8)

We consider these components as the most important ones, which on the one side dictate the quality and realism of the simulation studies and on the other side compare the protocols against each other. We do not consider theoretical studies or real-world studies, as those are rather rare in the area of OppNets.

**Table 1.** OppNet protocols involved in this study.

Protocol	Year of Publication	Concise Description
Epidemic Routing [11]	2000	Flooding of all the messages to all the neighbors.
Fresh [12]	2003	Nodes having the most recent encounters with destination are chosen as forwarders.
SEPR [13]	2003	Nodes with the shortest expected path length to the destination are selected as forwarders.
Seek and Focus [14]	2004	Randomized forwarding and Utility-based forwarding are used based on encounter rates.
Spray and Wait [15]	2005	Controlled-replication. Spraying $L$ copies of message $M$ to relays and holding the final copy until the destination node is met.
MobySpace [16]	2005	Nodes having similar mobility pattern to the destination node.
Maxprop [17]	2006	Estimating highest delivery likelihood of neighbors to the destination based on frequency of encounters with the destination.
Spray and Focus [18]	2007	Spraying message copies to the neighbors and utility forwarding
Prioritized ER [19]	2007	Estimation of routing cost to destination and prioritization of packet bundles to transmit based on routing cost.
HiBop [20]	2007	Contact history and stored context information about neighbors are used for selection of forwarders.
Geo-Opps [21]	2007	Using geographic location of the destination and prior known route schedules of destination and encountered neighbors.
Propicman [22]	2007	Estimating the probability of meeting the destination by sending probes to two hop neighbors.
Utility-based Spraying [23]	2007	Nodes choose between different utilities such as most mobile first, most social first, etc. for selection of relays.
Rapid [24]	2007	Adaptive replication of copies based on a chosen utility metric such as minimizing average delivery latency, minimizing maximum latency, etc.
ORWAR [25]	2008	Controlled-replication such as spray and wait routing combined with prioritization of messages by choosing required utility metrics.
EBR [26]	2009	Replication-based and selection of forwarders based on the number of encounters with the destination.
CAR [27]	2009	Evaluating and predicting context information such as mobility locally and by sending updates to neighborhood, using delivery predictability for relay selection.
SimBetTs [28]	2009	Relay selection based on social analysis on betweenness centrality, similarity index and strength of ties between nodes.
FairRoute [29]	2009	Tie strength and social status of nodes to assist in forwarding decisions.
Prophet+ [30]	2010	Enhancements to delivery predictability by considering node's buffer, power, location, popularity along with the delivery predictability obtained from prophet.

Table 1. Cont.

Protocol	Year of Publication	Concise Description
Prophet v2 [31]	2011	Fine-grained encounter rates by taking into account the unsuccessful intermittent connections, thus increasing the reliability of delivery predictability.
BubbleRap [32]	2011	Social and structural properties such as centrality and community metrics are used to select forwarders.
R3 [33]	2011	Estimates end-to-end delays of different paths to the destination and selects the path with best replication gain, uses adaptive replication.
3R [34]	2011	Estimates encounter probability by prediction of fine-grained regular encounters pertaining to a time window in a given day.
dLife [35]	2012	Daily routines of users are used to increase the accuracy of predicting future social contacts.
Sprint [36]	2013	Predicting the future contacts over a given time by analyzing the mobility patterns and using additional information on social contacts.
SGBR [37]	2013	Identifying social groups based on frequent and longer contact durations, social properties used to route packets between different communities.
Scorp [38]	2013	Prediction of probability of encounters having content-specific interests among the neighbors with similar social interests and daily routines.
HBPR [39]	2013	Exchange of location updates between nodes and predicting the direction of future movements to find shortest paths to the destination.
Onside [40]	2014	Community identification based on similar social interests and exchange of interests table between nodes to chose forwarders.
JDER [41]	2014	Targets high reachability by giving preference to selecting nodes connected to multiple communities (cut-nodes) as forwarders.
GAER [42]	2014	Genetic algorithm based next hop selection, distance between mean of home locations of neighbors to destination along with a fitness function is used to select forwarders.
PRoWait [43]	2015	Uses delivery predictability of Prophet to select forwarders combined with spray and wait routing.
Pathsampling [44]	2016	Learns network topology by using probes sent along with the beacons and selects forwarders based on estimated end-to-end delivery probability.
CGrAnt [45]	2016	Local information, situational information and domain information is used to select forwarders.
ABCON [46]	2016	Relay nodes are selected by the number of encountered neighbors.
EER [47]	2016	Calculates expected encounter value for every neighbor by using contact durations and frequent contacts which determines the forwarders and the number of copies.
RPC [48]	2016	Estimates reachability probability computed considering centrality measure and encounters rates, which is used in selecting forwarders.
EDR [49]	2016	Encounter rates with the destination and estimated distance to the destination are used to select forwarders.
GSAF [50]	2016	Destination dependent identifier is used to spray messages to forwarders. After the message is in the locality of the destination, it is flooded to all neighbors.
HPR [51]	2016	Delivery predictability of nodes with spraying is the forwarding strategy.
E2FA [52]	2016	Delivery predictability of nodes and buffer utility are used for choosing forwarders.

Table 1. Cont.

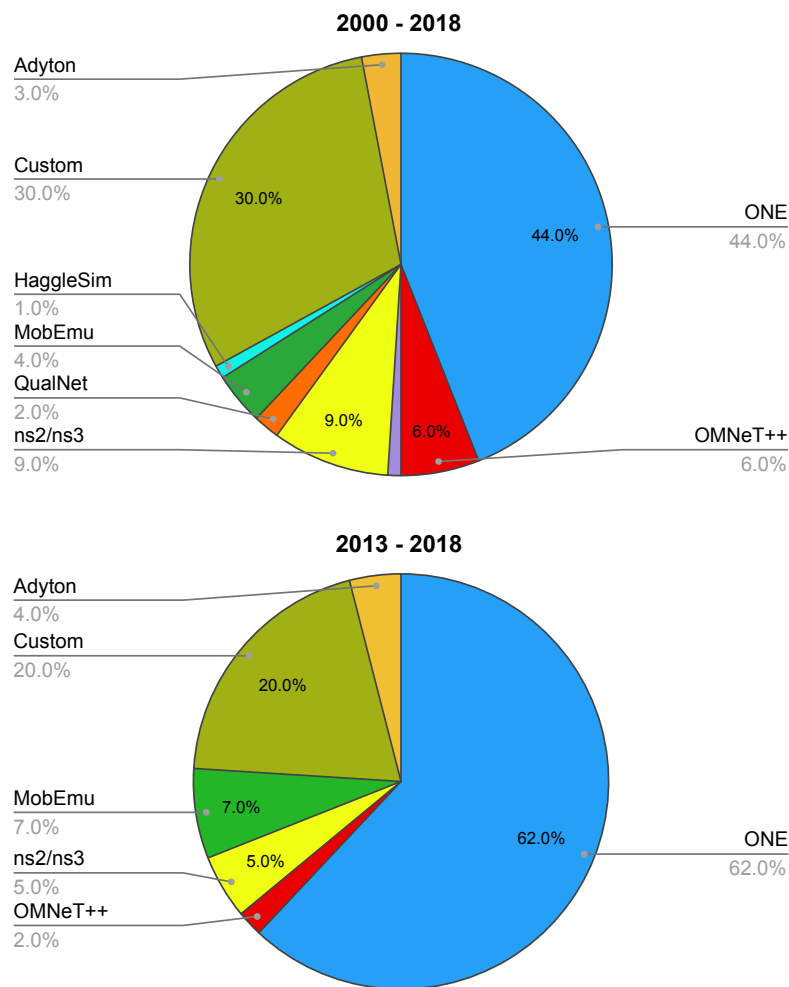
Protocol	Year of Publication	Concise Description
Multi-S&W Routing [53]	2017	Spray and wait routing with next hop selection based on weighted sum of betweenness centrality, friendship index and similarity index.
FGAR [54]	2017	Adaptive replication and forwarding based on contact prediction and success probability of delivery between contacts in a given time interval.
IBR [55]	2017	Effects of interactions between nodes in terms of popularity without detection of communities is the forwarding strategy.
FC-DFCM [56]	2017	Relationship strength of node pairs and contact durations are used in selection of forwarders.
EIMCT [57]	2017	Social network based contact durations are considered for forwarding decisions.
SAPR [58]	2017	Social characteristics of nodes and their mobility are the factors for making forwarding decisions.
CPR [59]	2017	Prediction-based routing decision considering statistical contact information, contact transitivity and instant contact information.
TCCB [60]	2017	Temporal social contact patterns and temporal centrality prediction are used to select forwarders.
FARS [61]	2018	Fairness based routing involving weighted factors of contact duration, residual buffer and historical amount of delivered data.
ELPFR-MC [62]	2018	Energy-aware routing based on node's residual energy, location prediction and delivery probability.
Predict and Forward [63]	2018	Next hop selection based on node profiles and attributes along with historical encounters to calculate delivery probability.
IoR [64]	2018	Average time since the message generation, average distance travelled in hops and delivery predictability of ProPhet are used to select forwarders.
CAOF [65]	2018	Relay selection is based on node's activeness to meet nodes in its own community and different communities within a bounded time.
CAF [66]	2018	Adaptive weighted combination of friendship index, similarity index, centrality, contact strength and trust to find the suitable relays.
CbR [67]	2018	Nodes belonging to clusters with good delivery capability are preferred than just the node's utility value.
RBES [68]	2018	Congestion level at a node and contact history of nodes are together used for forwarding strategy.
kROp [69]	2018	Partition of neighbors to clusters and next hop is determined by an evaluation function to find the cluster with optimal delivery capability.
PBQ [70]	2018	Delivery probability is calculated by Poisson based distribution along with consideration of node's daily routines and mobility patterns.
MLProph [71]	2018	Enhanced delivery predictability of ProPhet with machine learning by considering node's popularity, power consumption, speed, location and frequently encountered nodes.
EPSoc [72]	2018	Social-based epidemic routing where degree centrality of nodes is used for next hop selection.
CoSim [73]	2018	Cosine similarity of the data packets between nodes are used to find the similarity of the nodes which in turn are used to forward messages.
CGR [74]	2018	Replication based on scheduled contact patterns with predetermined network prediction.

#### 4. Simulator Environments

The first step in evaluating the performance of an OppNet protocol is the selection of the simulation platform to use. In general, there are two main options: custom built simulators and

standard simulators. The standard simulators usually used for OppNets are ONE [75], OMNeT++ [76], ns-2/ns-3 [77], MobEmu [78], and Adyton [79]. For both custom-built and standard simulators, there is also the option of providing the original code for reproducibility purposes or not.

Figure 2 shows which network simulators were used in all explored studies and in the recent ones since 2013. A tendency can be observed that recent studies use more standard simulators, especially the ONE, and fewer custom built ones. This is surely a positive trend, as it helps towards re-usage of models and reproducibility.



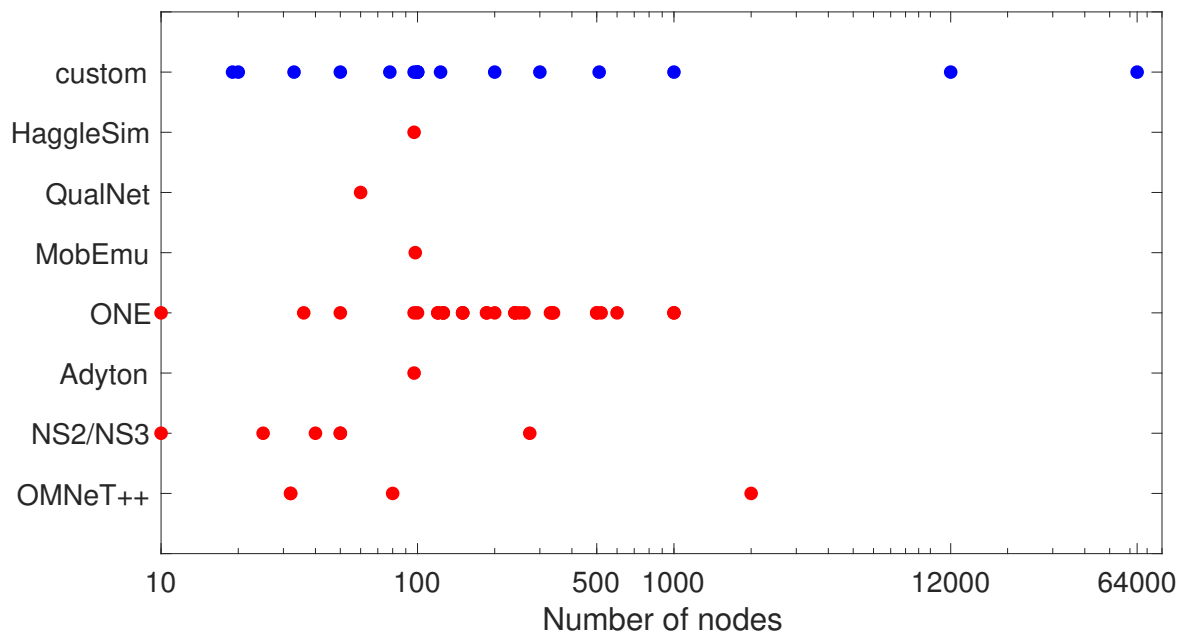
**Figure 2.** Used evaluation environments: for all surveyed publications (top); and for all recent studies since 2013 (bottom).

There are various reasons one might argue against and for custom-built simulators. One possible reason is speed and optimization. Standard simulators are typically of general use and tend to be slower. This potentially limits the scale of the OppNet simulation. Figure 3 shows the correlation between number of nodes (simulation scale) and whether a standard or a custom simulator has been used. This figure shows that in fact the largest simulations with 12,000 and 64,000 nodes have been conducted on custom simulators. However, these are rather outliers and most of the custom-built simulators have been used for much smaller network sizes. In addition, standard simulators have been used for studies with 1000 and 2000 nodes.

Another argument for implementing custom-built simulators has been the learning experience and the deep understanding of the built system. However, self-built solutions tend to be simpler and ignore many important properties. Thus, an argument we find in various references [80] and



from speaking directly to the authors is that, while a deep understanding and mastering of the used environment is surely of great importance, learning from well-designed and carefully tested already existing simulators is probably the better choice.



**Figure 3.** Simulation scale for different simulation environments, for all papers in our surveys which declare the number of nodes.

An important question is which simulator to choose. We compared the scalability and availability of simulation models in an earlier work in [8], where we concluded that the ONE has the largest variety of simulation models, while C++ based simulators like OMNeT++ or Adyton are much faster. Our current work is partially focused on further improving the scalability of our own OMNeT++ based OppNets framework OPS [81] and increasing the number of available models there. Finally, it does not matter which simulator is selected, as long as the original code is made available to the research community and the evaluation methodology is carefully designed.

Unfortunately, only one of the studies point to where their source code can be found: CGR [74]. Surely, some of them have found the way and became part of the ONE or other simulators (e.g., MaxProp, Epidemic, BubbleRap, SimBet, and ProPhet), but this is rather the exception. Reproducibility is highly important for fostering research in general and providing the used source code would be an important first step.

*Take home message: Standard simulation environments are to be preferred against self-built ones and source code should be made available to the research community.*

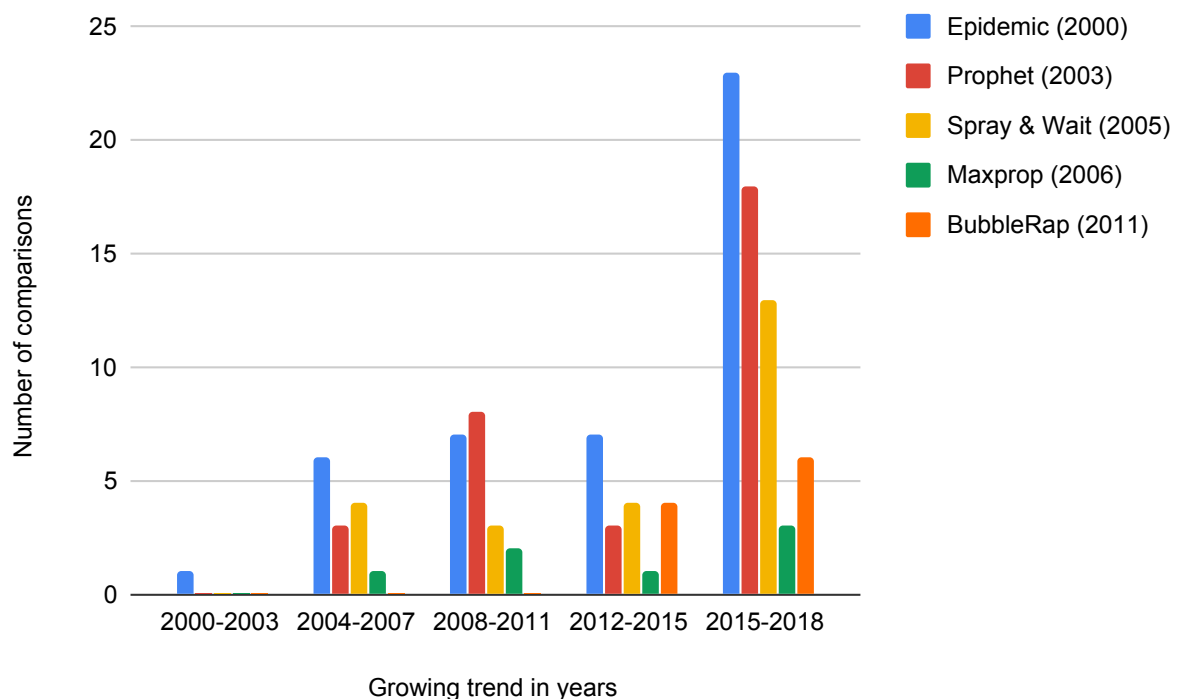
## 5. Comparative Studies

One of the most important decisions when designing an evaluation study is whether to compare the new protocol against some existing ones. The researcher potentially weakens her protocol and its chances to be published if no such comparison is provided. However, comparing against many different protocols also does not make a lot of sense and costs a lot of efforts, since most of the protocols are not readily available.

Thus, researchers often opt for a tradeoff, where they compare their new protocols against available and well explored ones. In the case of OppNets, these protocols are typically Epidemic, ProPhet, Spray & Wait, MaxProp and BubbleRap, as shown in Figure 4. It can be seen clearly in Figure 4 that the most widely used protocols for comparison are Epidemic, Prophet, Spray & Wait, Maxprop and BubbleRap and this trend stays the same even in the recent studies during 2015–2018.

We also present these results extensively in Figure A1 in Appendix A, where we have marked all protocols which have been used for comparison from other protocols. For example, the SEPR protocol from 2003 (Line 3 in the figure) has been compared against Epidemic. It can be clearly seen that freely available implementations of protocols (marked blue in the figure) are preferred options to compare against. Furthermore, the matrix in Figure A1 gets “thinner” towards the right bottom corner, while it should be actually filled under its diagonal.

This comparison strategy has one important disadvantage: it does not allow the research community to evaluate newer protocols against each other. For example, both Predict & Forward from 2018 and PathSampling from 2016 were compared against Epidemic and Spray & Wait. However, which is better or to be preferred in which situations? They are obviously close enough in their application scenarios to each other, since both selected the same set of comparative protocols.



**Figure 4.** Comparative studies of the surveyed OppNet protocols with the most compared OppNet protocols and their trends.

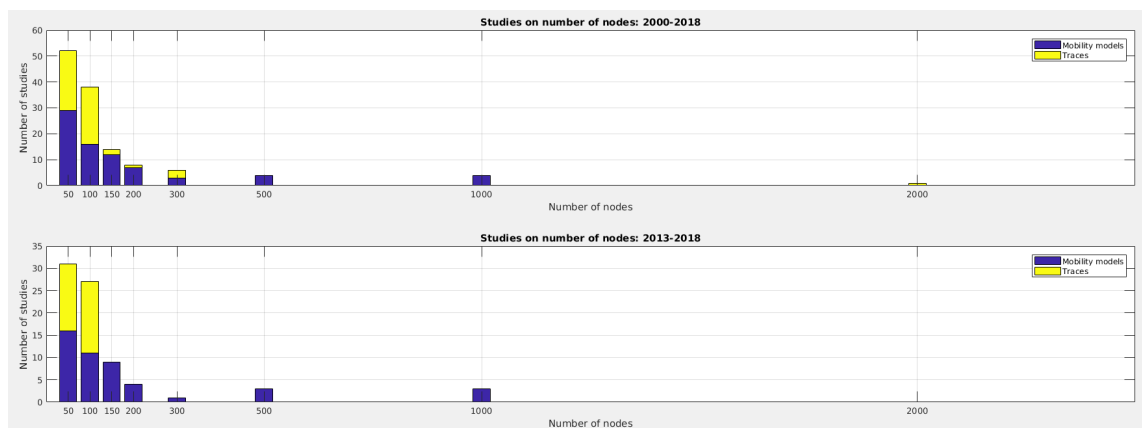
The observations from this section lead us to only one possible way out of this problem: new protocols need to be implemented in standard simulators and their code needs to be published. This is the only possibility to enable future studies to use the most recent advances in the area. This would also have the positive side effect of researchers reproducing and confirming the results of each other and thus pushing the state of the art forward faster. Comparative studies against traditional protocols such as Epidemic can still be very valuable to explain better the new properties or to set the new protocol in context.

**Take home message:** *New protocols need to be compared against recent advances and their code needs to be freely available.*

## 6. Scalability

Scalability in OppNets is directly related to the network size, i.e., the number of nodes in the network. The network size considered in the studied protocols is represented in Figures 3 and 5. Most of the studies have been evaluated for a network size of fewer than 100 nodes. Very few evaluations have considered more than 200 nodes and the trend declines rapidly for 500 nodes, 1000 nodes and above. In one of the studies (Fresh [12]), the network size ranged 1000–64,000 nodes. In studies where

mobility traces were used (see also the discussion in the next Section 7), the network size was limited by the number of nodes in the traces.



**Figure 5.** Frequently used network size in the studies. Very large outliers are not shown for better visibility.

We discuss realistic network densities and number of nodes in Section 2 above. Table 2 presents an overview of the here discussed protocols and the densities they have been tested for (Figure 1). Not all protocols provide enough details to evaluate the network density; some do not even provide the network size. Many of the papers refer to some of the application scenarios in Section 2, but the used parameters settings are rather small and thus not very realistic.

However, it needs to be noted again that many simulators (refer to Section 4, Figure 3 and our previous work in [8]) do not cope well with large simulations. Nevertheless, simulations with approximately 500–1000 nodes can be considered realistic and are well supported by all simulators. There are also good mobility traces at this network scale (refer also to Section 7).

**Take-home message:** Large simulations with thousands of nodes are still hard to achieve nowadays. This is one of the most important challenges and goals for OppNet modeling. New studies should consider the targeted application scenarios (Figure 1) and target at least 500–1000 nodes. Some special application scenarios, e.g., group monitoring applications, might require a custom scale.

**Table 2.** Node densities for studied protocols (missing protocols do not provide enough information).

Protocol	Extremely Sparse: 1 Device per km <sup>2</sup>	Sparse: 1–25 Devices per km <sup>2</sup>	Populated: 26–500 Devices per km <sup>2</sup>	Dense: 501–25,000 Devices per km <sup>2</sup>	Very Dense: 25,001–1 Million Devices per km <sup>2</sup>	Extremely Dense: 100 Million Devices or More per km <sup>2</sup>
Epidemic		x	x	x		
Seek and Focus		x		x		
Spray and Wait	x	x	x	x		
Maxprop	x	x				
Spray and Focus		x	x			
Prioritized Epidemic		x				
CAR		x	x			
HiBop		x				
GeoOpps		x				
Propicman			x			
Utility Based		x	x			
Rapid	x					
ORWAR		x				
EBR		x				
SimBeTs			x			
Prophet	x	x	x			
BubbleRap		x	x			
dLife		x				
Sprint			x			
Scorp		x				
HBPR		x				
GAER		x				

Table 2. Cont.

Protocol	Extremely Sparse: 1 Device per km <sup>2</sup>	Sparse: 1–25 Devices per km <sup>2</sup>	Populated: 26–500 Devices per km <sup>2</sup>	Dense: 501–25,000 Devices per km <sup>2</sup>	Very Dense: 25,001–1 Million Devices per km <sup>2</sup>	Extremely Dense: 100 Million Devices or More per km <sup>2</sup>
CPR		x				
Prowait		x				
CGrAnt		x				
GSAF		x	x			
PathSampling			x			
ABCON			x			
EER		x				
RPC		x				
EDR		x				
HPR		x				
FC-DFCM		x				
EIMST		x	x			
SAPR			x			
CAF			x			
CoSim		x				
ELPFR-MC		x				
FARS			x			
IoR		x				
kRoP		x				
MLProph		x				
PBQ Routing		x				
PnF			x			

## 7. Mobility

Mobility is probably the most important simulation model for evaluating OppNets. It is the main driver of OppNets and how messages get diffused in the network. Mobility (or its absence) controls which nodes are meeting each other and for how long.

We explored the different mobility models for OppNets in our earlier survey [8]. Generally speaking, we identified three groups of mobility models:

- **Random** mobility traces use analytical models to describe the mobility of devices/people in OppNets. They are simple, fast, but very unrealistic. Examples include Random Waypoint, Random Direction, etc.
- **Real trace** mobility models gather real GPS or other location data from real users and replay them in simulation. They are very realistic, easy to implement, but slow. Furthermore, gathering the traces is a very tedious task and there is no way to increase the number of nodes later. There exist a well-known database with such traces, called *Crowdad* (<https://crowdad.org>).
- **Hybrid** models combine both worlds by extracting statistical data or observations from real traces and then implementing a randomized model based on those. They are faster than real traces and more realistic than random models. However, they also quickly become very complex to understand and implement. It is also very hard to grasp all behavioral observations in one model. Examples for such models include SWIM (Small Worlds in Motion) [82], HCMM [83] or TRAILS [84].

The properties we are looking at for mobility models are:

- **Scalability:** How many nodes can a model produce/simulate? Random and hybrid mobility models are not really restricted in their scalability: as many nodes as needed can be simulated. However, traces are limited to the maximum number of nodes they have been collected for.
- **Realism:** How realistic is the behavior of the moving nodes? Real traces are clearly real. Random models are least realistic, while hybrid models tend to have more realistic properties.
- **Generalization:** How general can the results be considered? A single real trace is a snapshot and thus not representative. Analytical (random and hybrid), when used for a large number of scenarios and parameters, can be considered representative studies with statistical significance.

The properties of the three families of mobility models are also summarized in Figure 6 (left). Figure 6 shows how different mobility models vary in degrees of their relevance in terms of generalization, realism and scalability. For instance, a single trace is highly realistic, although neither highly scalable nor relevant in terms of generalization, as it adheres to a specific scenario. Until recently, the models have been used exclusively, i.e., only one model was applied in a particular simulation. However, other approaches are also possible: running simulations with several mobility models separately and together. The first idea is quite straightforward: a simulation is run first with mobility Model A, then with Model B, Model C, etc. All results together are used to derive the performance of the explored protocol.

The second idea is more complex and has not been applied yet. Here, individual traces are not run separately from each other, but on top of each other. The coordinate systems of the individual traces need to be converted to match. In this way, a much more scalable and dense simulation is possible, which makes the performance evaluation scalable and general at the same time. The effect of these modifications is also shown in Figure 6 (right). The only remaining challenge is performance of the model itself, as trace-based models are known for their slow performance.

The idea of using several mobility models at the same time is not exclusive for traces. It is also very useful with hybrid models, where for example some nodes move similar to vehicles and others similar to pedestrians. This has been for example applied in GSAF [50], kRop [69] and MLProph [71] where bus movement is used for vehicles and pedestrians use shortest path map based movement.

Figure 7 summarizes the used mobility models in the here explored studies. There is a slight trend towards using more than two traces or at least one trace and one analytical model. This is a very positive trend, as it makes the studies more representative.

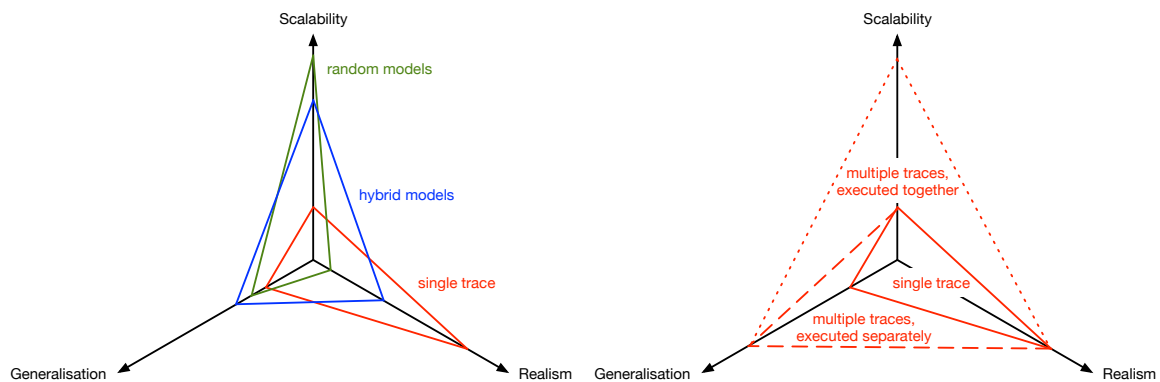


Figure 6. Properties of standard mobility models (left); and extension opportunities for trace-based models (right).

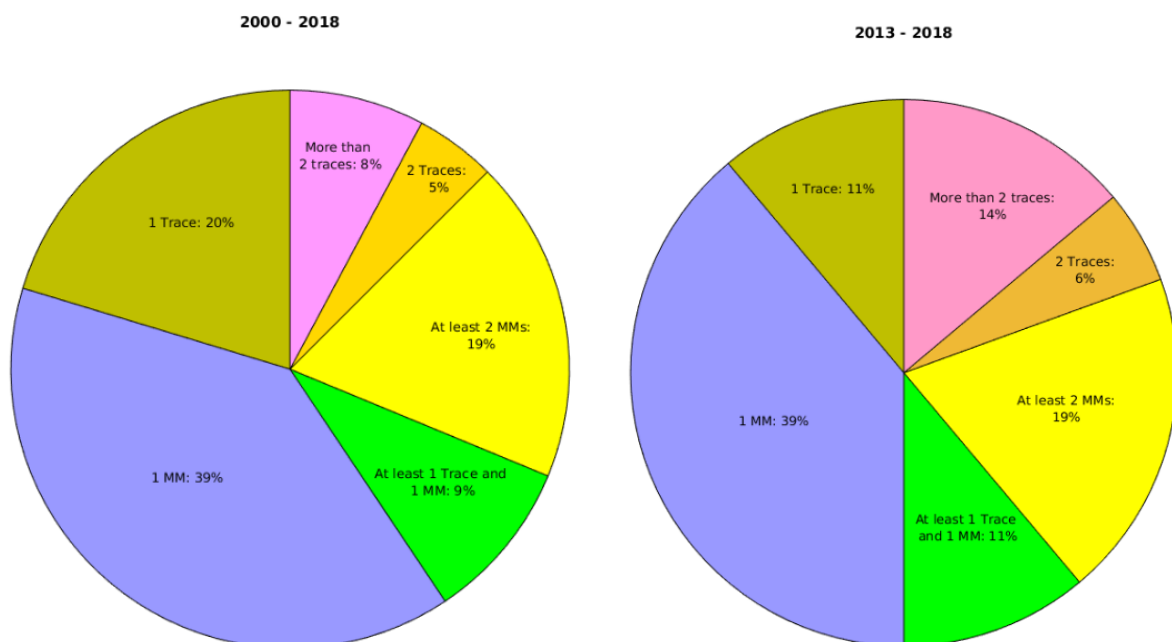


Figure 7. Used mobility models in the here explored studies. There is a slight trend towards using more than two traces or at least one trace and one mobility model.

**Take home message:** Using realistic and scalable mobility models is crucial for performance evaluation. A good option is a sophisticated hybrid mobility model, with a large variety of scenarios and parameters settings. Using a benchmark-like evaluation with several traces and hybrid mobility models is an even better choice.

### 8. Cache Size and Traffic

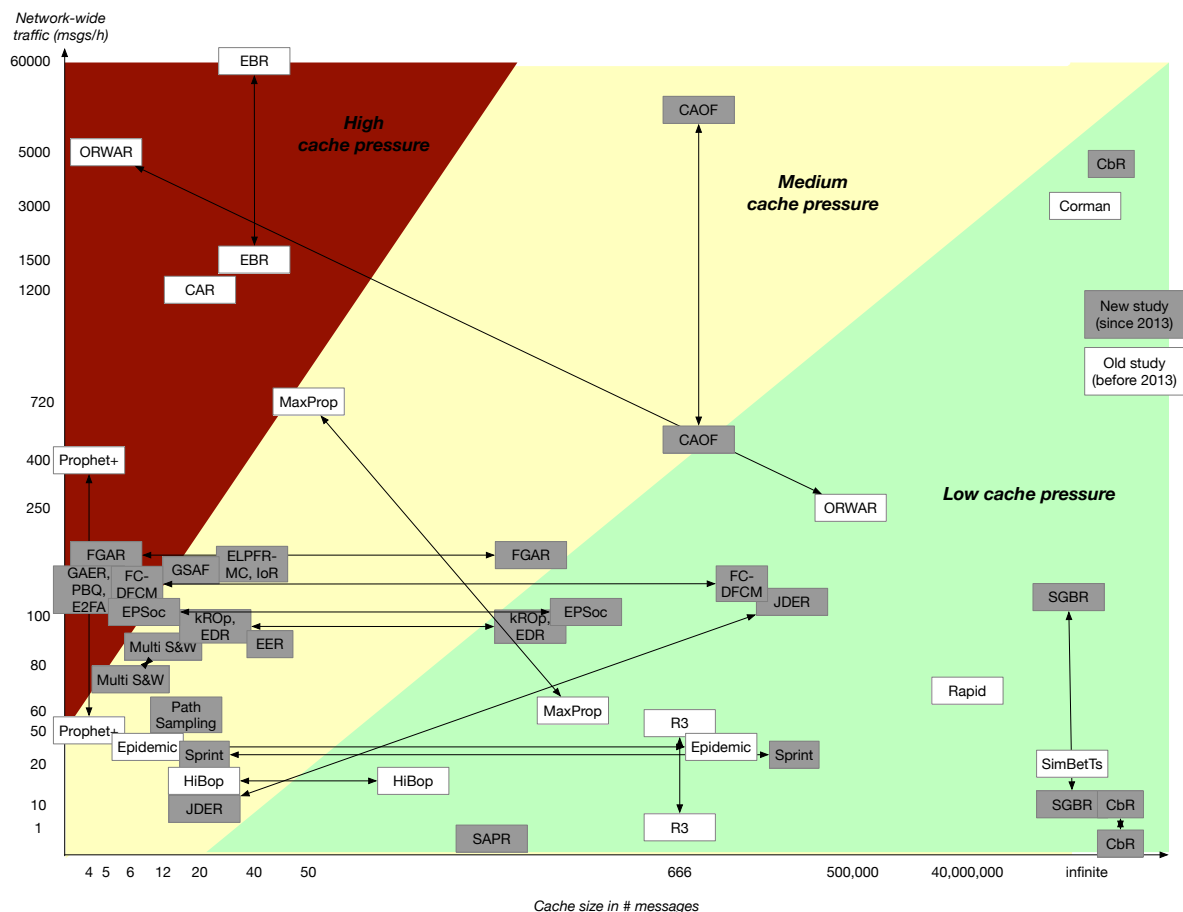
The second most important parameter of the performance evaluation of OppNets, after mobility, is the amount of messages in the network. Usually, this parameter is driven by four independent other parameters, i.e., cache size, traffic generation, time-to-live of messages and simulation time. However, the explored studies do not homogeneously explore those parameters in the same way, thus an analysis is not possible. Instead, we analyze the network-wide traffic in number of messages per hour against the cache size (in number of messages). The results are provided in Figure 8. Not all studies provide

enough details, thus only a subset are depicted in the graph. In the same graph, we have marked three different areas: high, medium and low cache pressure.

In low cache pressure scenarios, the generated traffic will never exceed the cache size and thus no buffer drops will be observed (unless time-to-live of messages is used). In medium cache pressure scenarios, buffer drops will occur, but rarely and will likely not affect the work of the protocol significantly. In high cache pressure scenarios, however, buffer drops will dominate.

Ideally, any protocol should be evaluated in all three scenarios to cover all possibilities. The protocols which get close to this goal are ORWAR and MaxProp. In Figure 8, we have also marked newer protocols (since 2013) in grey. We can observe that many of these newer studies have considered high to medium cache pressure scenarios. Older protocols often do not provide enough details about their traffic generation patterns, which is the reason why they are missing from this graph (few white boxes).

The most often used traffic generation frequency is by far 25–35 s. This comes from the standard settings of the ONE simulator. Thus, it is highly important how simulators are configured by default, since many researchers use these parameters under the assumption that they are somehow more “correct”. We consider this risky, as often such default settings are chosen either at random or are historically motivated (old settings from other scenarios). The traffic pattern and frequency depends highly on the scenario used, ranging from few messages per day for human-generated messages to hundreds of messages per second for sensor-generated ones.



**Figure 8.** Cache sizes against network-wide traffic. Not all studies provide sufficient details, thus only a subset is depicted.

**Take home message:** Cache sizes and traffic have not been considered very seriously in existing studies, taking default values. However, the parameter settings need to be well adjusted towards the application scenario taken into account.



## 9. Metrics

In this section, we briefly discuss the mostly commonly used metrics for all the protocols in the surveyed papers and the potential to identify new metrics specific to an OppNet.

*Delivery ratio* is the ratio of total number of packets delivered to the respective destinations to the total number of generated packets in the network. *Delivery delay* is the time difference between generation of packets and the reception of the packets at the destination. *Delivery cost* mainly attempts to capture the impact of replication on the performance of the OppNet. Hence, the delivery cost is usually measured in terms of the overhead such as number of copies or number of transmissions.

Specific analysis with respect to the metrics are not presented here owing to the fact that all the protocols in Table 1 utilize all three of the above mentioned metrics. The enhancements in the protocols always involve a trade-off in the metadata exchanged between nodes prior to the actual message transfer. The metadata comprises all the data stored in each node for choosing relays such as rate of encounters, mobility patterns, profiles of nodes for predicting future contacts and daily schedules. As this certainly involves the necessary information exchange between nodes upon contact, it can be considered as pre-processing overhead and should be accounted for in evaluations. Moreover, the pre-processing overhead depends on the operation of the protocols themselves. Hence, it is essential for the authors to identify all such possible overhead in evaluating their respective OppNet protocols.

## 10. Holistic Guide to OppNets Evaluations

In each of the previous sections, we have identified the most important insights in take-home messages. Here, we summarize our findings from the trends observed in the OppNet evaluations and further identify few more general properties and challenges in evaluating OppNet protocols.

First, and most importantly, performance evaluations have been improving over the last years significantly, with larger and more realistic scenarios being evaluated, and more sophisticated models being applied. Nevertheless, there is room for further improvement.

The best practices can be summarized as:

- Select a standard simulator. We recommend ONE or OMNeT++, as those are the most well documented and actively developed ones.
- Select OppNet protocols to compare against. They should be close in their general application scenario (destination-less or destination-oriented, etc.) and should be recent, e.g., from the last five years. Additionally, compare against optimal solutions. This combination ensures the correct positioning of the new protocols into the context of existing ones and how to progress the state of the art.
- Design a good application scenario with realistic number of nodes, traffic, cache sizes and simulation time.
- Select a good mobility model, able to cater for the application scenario and its scale. Traces (see also the discussion below) or recent hybrid mobility models are a good option. If using traces, use at least 3–5 different traces. Best, use several hybrid models and several traces.
- Explore the relevant protocol specific metrics in terms of overhead.
- Explore the parameter space of your scenario from minimum to maximum possible values. Report on confidence intervals.

Additionally, we would like to summarize our findings in the form of future work suggestions.

**Suggestion 1: Open source code.** The most important and significant improvement is the easiest and the most difficult to enforce at the same time: open source code. If all researchers would publish their code and make it available for other researchers to use and validate, the whole community will profit and the development will speed up significantly. It sounds easy, but there are also counter arguments. Publishing the code means more work to tidy it up, check for errors, comment and make sure it compiles correctly out of your own machine. This is a step which many researchers skip to save

time. Companies are particularly reluctant to share their code, as they believe it would give too many details into possibly patented or copyrighted products, although the probability of actually patenting or bringing the code to market is very low. It is the task of publishers and editors to enforce this development and many communities have started requesting code and other supplementary materials as part of their publication process.

**Suggestion 2: Documentation and default parameter values.** Another important insight is the usage of default parameter values and available models in standard simulators. It is obvious from our discussions that researchers tend to use readily available models and pre-defined values, instead of designing them from scratch and with only quality in mind. This is due to two reasons: First, time restriction is again an issue. Implementing and testing a new model is a time-consuming task and the goals and advantages of doing this are not always obvious. This problem can be easily resolved with Suggestion 1. Second, the complexity of standard simulators and their models can hinder especially less experienced researchers in how to setup them correctly. Here, only a combination of the following can help: rigorous documentation; well-designed default values; design of benchmarking suites; and a lively, helpful community, ready for open discussions.

**Suggestion 3: Benchmarks.** Following from Suggestion 2, the development of benchmarks is highly advisable for the further development of the community. Many other communities have well designed and adopted benchmarks, which significantly simplify the development and testing of new algorithms and protocols on one side, but also in better judging their quality and true novelty, on the other side.

**Suggestion 4: Improved mobility models.** One of the most challenging parts of OppNet simulation is the mobility model. Existing models suffer from various problems and no real solution is in sight. One possibility is the above described benchmarking process: e.g., combining many different traces into one benchmark. In Section 7, we have also described an idea of how to put different traces “on top of each other” to run simultaneously.

**Suggestion 5: Faster simulations.** Research is also desperately needed to speed up simulations. Currently, it takes up to weeks to run a single large scenario. This is due to the level of detail in which simulations are currently executed, typically at the packet level. Some new approaches already exist, e.g., in FALCON [85], but require a lot of manual adaptation. It is desirable to develop a highly scalable and fast simulation environment, where protocols are still implemented in the same way as in the real world.

These suggestions build also the basis for our own future work in the area of OppNets, even if not all of them will be addressed soon and at the same time and the community is welcome to support us.

## 11. Conclusions

In this paper, we have focused on performance evaluation of opportunistic networking data dissemination/forwarding protocols. Differently from other surveys, we have discussed *how* the protocols were evaluated instead of *what* they actually do or whether they perform well. This study has led us to two main outcomes: a best practice evaluation process and a list of suggestions on further improving the process for the whole community.

**Author Contributions:** Conceptualization, V.K. and A.F.; methodology, V.K., U.M.T., and A.F.; software, V.K., and A.F.; validation, V.K., A.U. and A.F.; formal analysis, V.K. and A.F.; investigation, V.K. and U.M.T.; resources, A.F.; data curation, V.K. and U.M.T.; writing—original draft preparation, V.K., A.U. and A.F.; writing—review and editing, V.K., U.M.T., A.U. and A.F.; visualization, V.K. and A.F.; supervision, A.F.; project administration, A.F.; funding acquisition, V.K. and A.F.

**Funding:** This research was funded by German Academic Exchange Service (Deutscher Akademischer Austauschdienst, DAAD) Research Grants, Germany.

**Conflicts of Interest:** The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.



Protocol	Year of Publication	Number of nodes	Simulators used	Mobility models / Traces used	Protocols compared with
Epidemic routing	2000	50	ns-2	Random Walk	NA
Fresh	2003	1000 - 64000	custom	Random waypoint & Random walk	NA
SEPR	2003	-	custom	City section mobility model	Epidemic with drop oldest and drop least encountered buffer
Seek & focus	2004	20; 50	custom discrete event driven	Random walk, Random waypoint	Random, direct, utility-based
Spray & wait	2005	100, 200	custom discrete event driven	Random walk, Random waypoint	epidemic, random flood, utility flood, seek and focus
MobySpace	2005	300	custom discrete event driven	moility pattern from dartmouth trace	epidemic, random flood, spray and wait, direct delivery
Maxprop	2006	30, 40	testbed	UmassDieselNet - 30 nodes	Random, Oracle, Drop least encountered
Spray and focus	2007	100	custom	Random waypoint mobility model, Random walk	Epidemic, utility flooding, spray and wait
Prioritized Epidemic	2007	25	ns-2	Random waypoint mobility model	Epidemic
HiBop	2007	16 - 32	OMNet++	Community based Mobility Model	Epidemic and Prophet
Geo-Opps	2007	2000	OMNet++	Traces	Greedy and Move
Propicman	2007	30, 50	custom simulator	No but a community based scenario	Epidemic and Prophet
Utility-based	2007	100	custom simulator	community-based mobility model	s&W
Rapid	2007	avg 19 nodes per day	custom	UmassDieselnet - 40 nodes	maxprop, spray and wait, prophet, random
ORWAR	2008	126	ONE	City section Mobility model	maxprop, spray and wait, prophet, epidemic, direct
EBR	2009	Maxprop: 26, 51, 101, 151, rest: 25, 50, 100, 150, 200, 250	ONE	Random waypoint, vehicle map based, event driven	Epidemic, spray and wait, spray and focus, prophet, maxprop
CAR	2009	16, 24, 32	custom	social group based mobility model	Epidemic and Prophet
SimBetTs	2009	123, 223, 264	Omnnet++	traces - hagggle	Epidemic and CAR
Fair Route	2009	96	custom	MIT trace	Prophet, SimBet, Epidemic
Prophet+	2010	274	DTNSim /Ns	Infocomm 05 trace	Prophet
Prophet v2	2011	18, 500	ONE	traces (18 nodes), WDM - 500 nodes	Prophet, epidemic, spray and wait
BubbleRap	2011	41, 37, 54, 98, 97	HagggleSim	traces	prophet, simbet
R3	2011	16, 6, 60, 20	Qualnet Simulator	Mesh nw, Hagggle, Dieselnet traces, Dieselnet DTN	Rapid, Random
3R	2011	123	custom	University of illinois trace	prophet, epidemic
dLife	2012	150, 36	ONE	Working day movement, Cambridge traces	BubbleRap
Sprint	2013	33	custom	content traces, st.andrews traces, UPB, Infocom, HCMM	Bubblerrap, epidemic
SGBR	2013	10, 50	one	traces, map-based MM	Epidemic and Prophet, spray and wait, optimal maxprop
Scorp	2013	150 (MM), 36 (traces)	ONE	WDM, bus movement, shortest map based, traces	spray and wait, bubble rap, dLife
HBPR	2013	150	ONE	Custom Human Mobility model	Epidemic
Onside	2014	98, 76, 66	Mobemu	traces	epidemic, ML-SOR
JDER	2014	22, 66, 27, 9, 19, 54, 50, 98	custom	UPB 2011, UPB 2012, ST. Andrews, Intel, Cambridge, Content, Infocom, Infocom 2006	BubbleRap, SPRINT, Epidemic
GAER	2014	80	ONE	Custom - skewed towards community-based	Epidemic, S&W, Prophet
Prowait	2015	124	ONE	Shortest path map based movement model	Epidemic, S&W, Prophet
RPC	2016	41, 97	ONE	Infocomm 05, MIT Reality	Prophet, Epidemic

Figure A2. Cont.

PathSampling	2016	10	NS3	Randomwaypoint	epidemic, spray and wait
CGrAnt	2016	339, 140	ONE	Working day, POI movement model	Epidemic, Prophet, dLife
ABCON	2016	40	ns-2	HCMM	Epidemic, SimBet, Prophet
GSAF	2016	195	ONE	Shortest path map based movement model, bus movement	GeoEpidemic
EER	2016	40 - 240	ONE	vehicle-based map-driven model	CAR, EBR, MaxProp, S&W, S&F, Prophet
E2FA	2016	40 - 120	ONE	Shortest path map based movement	Epidemic, MaxProp, S&W, Prophet
EDR	2016	96	ONE	Shortest path map based movement	HBPR, ProWait
HPR	2016	80, 40, 6	ONE	Shortest path - based movement, map route movement	Epidemic, Prophet, S&W
Multi-spray & wait based	2017	100	ONE	SLAW, shortest map-based	epidemic, prophet, spray and wait
IBR	2017	49, 97, 78	Adyton	Milan (PMTR), MIT Reality, Infocomm 2006	Prophet, BubbleRap, Epidemic
TCCB	2017	96, 78	custom	MIT reality trace, Infocomm 06 trace	Epidemic, BubbleRap, Prophet
FC - DFCM	2017	120, 78	ONE	Community model, Infocomm 06	Prophet, BubbleRap
FGAR	2017	30 - 36	custom	traces - cambridge (200), MIT reality (20,000) and Dieselnet (30)	Epidemic, prophet, 3R and spray and wait
EIMCT	2017	1000	ONE	Shortest Path map-based movement	S&W, Epidemic
SAPR	2017	100, 200	ONE	-	S&W, Epidemic, Prophet
CPR	2017	20 - 100	ONE	RWP, Infocomm trace	Epidemic, S & W, Prophet, direct routing
FARS	2018	41, 100	custom	Infocomm trace, TVCM mobility model	SimBet, Epidemic, FairRoute
ELPFR - MC	2018	96, 336	ONE	Shortest path map based movement model	Prophet, ProWait, EDR, DEEP
Predict and Forward	2018	1000	ONE	Health capability maturity model	EIMST, S&W, Epidemic
IoR	2018	126	ONE	Shortest path map based movement model	RPC, Prophet
CAOF	2018	76	MobEmu	Sigcomm trace	BubbleRap, SPRINT, Epidemic, JDER
CAF	2018	30 (HCMM)	MobEmu	UPB 2012, Sigcomm 2009 trace, HCMM	BubbleRap, Epidemic
CbR	2018	76, 96, 44	Adyton	Infocomm 05, Sigcomm 06, MIT Reality, Milano PMTR	DF, COORD and Prophet
RBES	2018	100 - 500	ONE	Map Route Movement	MaxProp
kROp	2018	96, 66 - 186	ONE	Shortest path map based movement model, Route based map movement model	HBPR, ProWait, Prophet
PBQ	2018	100 - 200	ONE	Custom Human Mobility model	Epidemic, Prophet, HBPR
ML Proph	2018	150, 50, 40, 25, 14, 50, 40	ONE	Bus movement, Working day movement	Prophet+
EPSoc	2018	36	ONE	Cambridge trace	Epidemic and BubbleRap
CoSim	2018	50 - 600	ONE	-	S&W, Epidemic

Figure A2. Evaluation environment analysis from surveyed papers.

## References

1. Kurkowski, S.; Camp, T.; Colagrosso, M. MANET simulation studies: The incredibles. *ACM SIGMOBILE Mob. Comput. Commun. Rev.* **2005**, *9*, 50–61. [[CrossRef](#)]
2. Mota, V.F.; Cunha, F.D.; Macedo, D.F.; Nogueira, J.M.; Loureiro, A.A. Protocols, mobility models and tools in opportunistic networks: A survey. *Comput. Commun.* **2014**, *48*, 5–19. [[CrossRef](#)]
3. Trifunovic, S.; Kouyoumdjieva, S.T.; Distl, B.; Pajevic, L.; Karlsson, G.; Plattner, B. A decade of research in opportunistic networks: Challenges, relevance, and future directions. *IEEE Commun. Mag.* **2017**, *55*, 168–173. [[CrossRef](#)]
4. Pozza, R.; Nati, M.; Georgoulas, S.; Moessner, K.; Gluhak, A. Neighbor discovery for opportunistic networking in internet of things scenarios: A survey. *IEEE Access* **2015**, *3*, 1101–1131. [[CrossRef](#)]
5. Chakchouk, N. A Survey on Opportunistic Routing in Wireless Communication Networks. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 2214–2241. [[CrossRef](#)]
6. Sobin, C.; Raychoudhury, V.; Marfia, G.; Singla, A. A survey of routing and data dissemination in delay tolerant networks. *J. Netw. Comput. Appl.* **2016**, *67*, 128–146.

7. Batabyal, S.; Bhaumik, P. Mobility models, traces and impact of mobility on opportunistic routing algorithms: A survey. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 1679–1707. [CrossRef]
8. Dede, J.; Förster, A.; Hernández-Orallo, E.; Herrera-Tapia, J.; Kuladinithi, K.; Kuppusamy, V.; Manzoni, P.; bin Muslim, A.; Udugama, A.; Vatandas, Z. Simulating opportunistic networks: Survey and future directions. *IEEE Commun. Surv. Tutor.* **2018**, *20*, 1547–1573. [CrossRef]
9. García-Campos, J.M.; Sánchez-García, J.; Reina, D.; Toral, S.; Barrero, F. An evaluation methodology for reliable simulation based studies of routing protocols in VANETs. *Simul. Model. Pract. Theory* **2016**, *66*, 139–165. [CrossRef]
10. Saloni, M.; Julien, C.; Murphy, A.L.; Picco, G.P. Lasso: A device-to-device group monitoring service for smart cities. In Proceedings of the 2017 International Smart Cities Conference (ISC2), Wuxi, China, 14–17 September 2017; pp. 1–6.
11. Vahdat, A.; Becker, D. Epidemic Routing for Partially Connected Ad Hoc Netw. 2000. *Technical Report CS 200006 2000*, Department of Computer Science, Duke University, NC. Available online: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.34.6151> (accessed on 11 May 2019)
12. Dubois-Ferriere, H.; Grossglauser, M.; Vetterli, M. Age matters: Efficient route discovery in mobile ad hoc networks using encounter ages. In Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking & Computing, Annapolis, MD, USA, 1–3 June 2003; pp. 257–266.
13. Tan, K.; Zhang, Q.; Zhu, W. Shortest path routing in partially connected ad hoc networks. In Proceedings of the Global Telecommunications Conference, San Francisco, CA, USA, 1–5 December 2003; Volume 2, pp. 1038–1042.
14. Spyropoulos, T.; Psounis, K.; Raghavendra, C.S. Efficient routing in intermittently connected mobile networks: The single-copy case. *IEEE/ACM Trans. Netw.* **2008**, *16*, 63–76. [CrossRef]
15. Spyropoulos, T.; Psounis, K.; Raghavendra, C.S. Spray and wait: An efficient routing scheme for intermittently connected mobile networks. In Proceedings of the 2005 ACM SIGCOMM Workshop on Delay-Tolerant Networking, Philadelphia, PA, USA, 22–26 August 2005; pp. 252–259.
16. Leguay, J.; Friedman, T.; Conan, V. Evaluating mobility pattern space routing for DTNs. *arXiv* **2005**, arXiv:cs/0511102.
17. Burgess, J.; Gallagher, B.; Jensen, D.; Levine, B.N. Maxprop: Routing for vehicle-based disruption-tolerant networks. In Proceedings of the 25th IEEE International Conference on Computer Communications, Barcelona, Spain, 23–29 April 2006; pp. 1–11.
18. Spyropoulos, T.; Psounis, K.; Raghavendra, C.S. Spray and focus: Efficient mobility-assisted routing for heterogeneous and correlated mobility. In Proceedings of the Fifth Annual IEEE International Conference on Pervasive Computing and Communications Workshops, White Plains, NY, USA, 19–23 March 2007; pp. 79–85.
19. Ramanathan, R.; Hansen, R.; Basu, P.; Rosales-Hain, R.; Krishnan, R. Prioritized epidemic routing for opportunistic networks. In Proceedings of the 1st International MobiSys Workshop on Mobile Opportunistic Networking, San Juan, Puerto Rico, 11 June 2007; pp. 62–66.
20. Boldrini, C.; Conti, M.; Jacopini, J.; Passarella, A. Hibop: A history based routing protocol for opportunistic networks. In Proceedings of the 2007 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, Espoo, Finland, 18–21 June 2007; pp. 1–12.
21. Leontiadis, I.; Mascolo, C. GeOpps: Geographical opportunistic routing for vehicular networks. In Proceedings of the 2007 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, Espoo, Finland, 18–21 June 2007; pp. 1–6.
22. Nguyen, H.A.; Giordano, S.; Puiatti, A. Probabilistic routing protocol for intermittently connected mobile ad hoc network (propicman). In Proceedings of the 2007 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, Espoo, Finland, 18–21 June 2007; pp. 1–6.
23. Spyropoulos, T.; Turletti, T.; Obraczka, K. Utility-based message replication for intermittently connected heterogeneous networks. In Proceedings of the 2007 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, Espoo, Finland, 18–21 June 2007; pp. 1–6.
24. Balasubramanian, A.; Levine, B.; Venkataramani, A. DTN routing as a resource allocation problem. *ACM SIGCOMM Comput. Commun. Rev.* **2007**, *37*, 373–384. [CrossRef]

25. Sandulescu, G.; Nadjm-Tehrani, S. Opportunistic DTN routing with window-aware adaptive replication. In Proceedings of the 4th Asian Conference on Internet Engineering, Pratumam, Bangkok, Thailand, 18–20 November 2008; pp. 103–112.
26. Nelson, S.C.; Bakht, M.; Kravets, R. Encounter-based routing in DTNs. In Proceedings of the INFOCOM 2009, Rio de Janeiro, Brazil, 19–25 April 2009; pp. 846–854.
27. Musolesi, M.; Mascolo, C. Car: Context-aware adaptive routing for delay-tolerant mobile networks. *IEEE Trans. Mob. Comput.* **2009**, *8*, 246–260. [[CrossRef](#)]
28. Daly, E.M.; Haahr, M. Social network analysis for information flow in disconnected delay-tolerant MANETs. *IEEE Trans. Mob. Comput.* **2008**, 606–621. [[CrossRef](#)]
29. Pujol, J.M.; Toledo, A.L.; Rodriguez, P. Fair routing in delay tolerant networks. In Proceedings of the INFOCOM 2009, Rio de Janeiro, Brazil, 19–25 April 2009; pp. 837–845.
30. Huang, T.K.; Lee, C.K.; Chen, L.J. Prophet+: An adaptive prophet-based routing protocol for opportunistic network. In Proceedings of the 2010 24th IEEE International Conference on Advanced Information Networking and Applications, Perth, Australia, 20–23 April 2010; pp. 112–119.
31. Grasic, S.; Davies, E.; Lindgren, A.; Doria, A. The evolution of a DTN routing protocol-PROPHETv2. In Proceedings of the 6th ACM Workshop on Challenged Networks, Las Vegas, NV, USA, 19–23 September 2011; pp. 27–30.
32. Hui, P.; Crowcroft, J.; Yoneki, E. Bubble rap: Social-based forwarding in delay-tolerant networks. *IEEE Trans. Mob. Comput.* **2011**, *10*, 1576–1589. [[CrossRef](#)]
33. Tie, X.; Venkataramani, A.; Balasubramanian, A. R3: Robust replication routing in wireless networks with diverse connectivity characteristics. In Proceedings of the 17th Annual International Conference on Mobile Computing and Networking, Las Vegas, NV, USA, 19–23 September 2011; pp. 181–192.
34. Vu, L.; Do, Q.; Nahrstedt, K. 3r: Fine-grained encounter-based routing in delay tolerant networks. In Proceedings of the 2011 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, Lucca, Italy, 20–24 June 2011; pp. 1–6.
35. Moreira, W.; Mendes, P.; Sargento, S. Opportunistic routing based on daily routines. In Proceedings of the 2012 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), San Francisco, CA, USA, 25–28 June 2012; pp. 1–6.
36. Ciobanu, R.I.; Dobre, C.; Cristea, V. SPRINT: Social prediction-based opportunistic routing. In Proceedings of the 2013 IEEE 14th International Symposium on “A World of Wireless, Mobile and Multimedia Networks” (WoWMoM), Madrid, Spain, 4–7 June 2013; pp. 1–7.
37. Abdelkader, T.; Naik, K.; Nayak, A.; Goel, N.; Srivastava, V. SGBR: A routing protocol for delay tolerant networks using social grouping. *IEEE Trans. Parallel Distrib. Syst.* **2013**, *24*, 2472–2481. [[CrossRef](#)]
38. Moreira, W.; Mendes, P.; Sargento, S. Social-aware opportunistic routing protocol based on user’s interactions and interests. In Proceedings of the International Conference on Ad Hoc Netw., Wroclaw, Poland, 8–10 July 2013; pp. 100–115.
39. Dhurandher, S.K.; Sharma, D.K.; Woungang, I.; Bhati, S. HBPR: History based prediction for routing in infrastructure-less opportunistic networks. In Proceedings of the 2013 IEEE 27th International Conference on Advanced Information Networking and Applications (AINA), Barcelona, Spain, 25–28 March 2013; pp. 931–936.
40. Ciobanu, R.I.; Marin, R.C.; Dobre, C.; Cristea, V.; Mavromoustakis, C.X. ONSIDE: Socially-aware and interest-based dissemination in opportunistic networks. In Proceedings of the 2014 IEEE Network Operations and Management Symposium (NOMS), Krakow, Poland, 5–9 May 2014; pp. 1–6.
41. Ciobanu, R.I.; Reina, D.; Dobre, C.; Toral, S.; Johnson, P. JDER: A history-based forwarding scheme for delay tolerant networks using Jaccard distance and encountered ration. *J. Netw. Comput. Appl.* **2014**, *40*, 279–291. [[CrossRef](#)]
42. Dhurandher, S.K.; Sharma, D.K.; Woungang, I.; Gupta, R.; Garg, S. GAER: Genetic algorithm-based energy-efficient routing protocol for infrastructure-less opportunistic networks. *J. Supercomput.* **2014**, *69*, 1183–1214. [[CrossRef](#)]
43. Dhurandher, S.K.; Borah, S.J.; Obaidat, M.S.; Sharma, D.K.; Gupta, S.; Baruah, B. Probability-based controlled flooding in opportunistic networks. In Proceedings of the 2015 12th International Joint Conference on e-Business and Telecommunications (ICETE), Colmar, France, 20–22 July 2015; Volume 6, pp. 3–8.

44. Visca, J.; Baliosian, J.; Fuentes, R.A.; Cavalli, A.R. Path sampling, a robust alternative to gossiping for opportunistic network routing. In Proceedings of the 2016 IEEE 12th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), New York, NY, USA, 17–19 October 2016; pp. 1–8.
45. Vendramin, A.C.K.; Munaretto, A.; Delgado, M.R.; Fonseca, M.; Viana, A.C. A social-aware routing protocol for opportunistic networks. *Expert Syst. Appl.* **2016**, *54*, 351–363. [[CrossRef](#)]
46. Park, J.; Lee, J.; Kim, S.K.; Jang, K.; Yang, S.B. A forwarding scheme based on swarm intelligence and percolation centrality in opportunistic networks. *Wirel. Netw.* **2016**, *22*, 2511–2521. [[CrossRef](#)]
47. Chen, H.; Lou, W. Contact expectation based routing for delay tolerant networks. *Ad Hoc Netw.* **2016**, *36*, 244–257. [[CrossRef](#)]
48. Wu, J.; Wang, J.; Liu, L.; Tanha, M.; Pan, J. A data forwarding scheme with reachable probability centrality in DTNs. In Proceedings of the 2016 IEEE Wireless Communications and Networking Conference, Doha, Qatar, 3–6 April 2016; pp. 1–6.
49. Dhurandher, S.K.; Borah, S.; Woungang, I.; Sharma, D.K.; Arora, K.; Agarwal, D. EDR: An encounter and distance based routing protocol for opportunistic networks. In Proceedings of the 2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA), Crans-Montana, Switzerland, 23–25 March 2016; pp. 297–302.
50. Rajaei, A.; Chalmers, D.; Wakeman, I.; Parisi, G. GSAF: Efficient and flexible geocasting for opportunistic networks. In Proceedings of the 2016 IEEE 17th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM), Coimbra, Portugal, 21–24 June 2016; pp. 1–9.
51. Yu, C.; Tu, Z.; Yao, D.; Lu, F.; Jin, H. Probabilistic routing algorithm based on contact duration and message redundancy in delay tolerant network. *Int. J. Commun. Syst.* **2016**, *29*, 2416–2426. [[CrossRef](#)]
52. Zhang, F.; Wang, X.; Jiang, L.; Zhang, L. Energy efficient forwarding algorithm in opportunistic networks. *Chin. J. Electron.* **2016**, *25*, 957–964. [[CrossRef](#)]
53. Guan, J.; Chu, Q.; You, I. The social relationship based adaptive multi-spray-and-wait routing algorithm for disruption tolerant network. *Mob. Inf. Syst.* **2017**, *2017*, 1819495. [[CrossRef](#)]
54. Zhang, H.; Wan, L.; Chen, Y.; Yang, L.T.; Peng, L. Adaptive Message Routing and Replication in Mobile Opportunistic Networks for Connected Communities. *ACM Trans. Internet Technol. (TOIT)* **2017**, *18*, 2. [[CrossRef](#)]
55. Li, Y.; Bartos, R. Interaction based routing algorithm for opportunistic mobile social networks. In Proceedings of the 2017 14th IEEE Annual Consumer Communications & Networking Conference (CCNC), Las Vegas, NV, USA, 8–11 January 2017; pp. 492–497.
56. Yu, X.; Liu, Z. Message forwarding strategy with fuzzy mechanism in opportunistic network. In Proceedings of the 2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), Chennai, India, 22–24 March 2017; pp. 309–314.
57. Jia, W.; Chen, Z.; Zhao, M. Effective information transmission based on socialization nodes in opportunistic networks. *Comput. Netw.* **2017**, *129*, 297–305.
58. Zhao, R.; Wang, X.; Zhang, L.; Lin, Y. A social-aware probabilistic routing approach for mobile opportunistic social networks. *Trans. Emerg. Telecommun. Technol.* **2017**, *28*, e3230. [[CrossRef](#)]
59. Zhang, L.; Wang, X.; Lu, J.; Ren, M.; Duan, Z.; Cai, Z. A novel contact prediction-based routing scheme for DTNs. *Trans. Emerg. Telecommun. Technol.* **2017**, *28*, e2889. [[CrossRef](#)]
60. Zhou, H.; Leung, V.C.; Zhu, C.; Xu, S.; Fan, J. Predicting temporal social contact patterns for data forwarding in opportunistic mobile networks. *IEEE Trans. Veh. Technol.* **2017**, *66*, 10372–10383. [[CrossRef](#)]
61. Ma, H.; Wu, H.; Zheng, G.; Ji, B.; Li, J. FARS: A Fairness-aware Routing Strategy for Mobile Opportunistic Networks. *KSII Trans. Internet Inf. Syst.* **2018**, *12*, 1992–2008. [[CrossRef](#)]
62. Borah, S.J.; Dhurandher, S.K.; Woungang, I.; Kandhoul, N.; Rodrigues, J.J. An Energy-Efficient Location Prediction-Based Forwarding Scheme for Opportunistic Networks. In Proceedings of the 2018 IEEE International Conference on Communications (ICC), Kansas City, MO, USA, 20–24 May 2018; pp. 1–6.
63. Liu, K.; Chen, Z.; Wu, J.; Xiao, Y.; Zhang, H. Predict and Forward: An Efficient Routing-Delivery Scheme Based on Node Profile in Opportunistic Networks. *Future Internet* **2018**, *10*, 74. [[CrossRef](#)]
64. Baek, K.; Seo, D.; Chung, Y. An Improved Opportunistic Routing Protocol Based on Context Information of Mobile Nodes. *Appl. Sci.* **2018**, *8*, 1344. [[CrossRef](#)]



65. Tao, J.; Wu, H.; Shi, S.; Hu, J.; Gao, Y. Contacts-aware opportunistic forwarding in mobile social networks: A community perspective. In Proceedings of the 2018 IEEE Wireless Communications and Networking Conference (WCNC), Barcelona, Spain, 15–18 April 2018; pp. 1–6.
66. Ciobanu, R.I.; Reina, D.G.; Dobre, C.; Toral, S.L. Context-adaptive forwarding in mobile opportunistic networks. *Ann. Telecommun.* **2018**, *73*, 559–575. [[CrossRef](#)]
67. Papapetrou, E.; Likas, A. Cluster-based Replication: A Forwarding Strategy for Mobile Opportunistic Networks. In Proceedings of the 2018 IEEE 19th International Symposium on “A World of Wireless, Mobile and Multimedia Networks” (WoWMoM), Chania, Greece, 12–15 June 2018.
68. Sajid, A.; Hussain, K. Rule Based (Forward Chaining/Data Driven) Expert System for Node Level Congestion Handling in Opportunistic Network. *Mob. Netw. Appl.* **2018**, *23*, 446–455. [[CrossRef](#)]
69. Sharma, D.K.; Dhurandher, S.K.; Agarwal, D.; Arora, K. kROp: K-Means clustering based routing protocol for opportunistic networks. *J. Ambient. Intell. Humaniz. Comput.* **2019**, *10*, 1289–1306. [[CrossRef](#)]
70. Sharma, D.K.; Kukreja, D.; Aggarwal, P.; Kaur, M.; Sachan, A. Poisson’s probability-based Q-Routing techniques for message forwarding in opportunistic networks. *Int. J. Commun. Syst.* **2018**, *31*, e3593. [[CrossRef](#)]
71. Sharma, D.K.; Dhurandher, S.K.; Woungang, I.; Srivastava, R.K.; Mohanane, A.; Rodrigues, J.J. A machine learning-based protocol for efficient routing in opportunistic networks. *IEEE Syst. J.* **2018**, *12*, 2207–2213. [[CrossRef](#)]
72. Lenando, H.; Alrfaay, M. EpSoc: Social-Based Epidemic-Based Routing Protocol in Opportunistic Mobile Social Network. *Mob. Inf. Syst.* **2018**, *2018*, 6462826. [[CrossRef](#)]
73. Lin, Y.; Chen, Z.; Wu, J.; Wang, L. An Opportunistic Network Routing Algorithm Based on Cosine Similarity of Data Packets between Nodes. *Algorithms* **2018**, *11*, 119. [[CrossRef](#)]
74. Madoery, P.G.; Raverta, F.D.; Fraire, J.A.; Finochietto, J.M. Routing in Space Delay Tolerant Networks under Uncertain Contact Plans. In Proceedings of the 2018 IEEE International Conference on Communications (ICC), Kansas City, MO, USA, 20–24 May 2018; pp. 1–6.
75. Keränen, A.; Ott, J.; Kärkkäinen, T. The ONE Simulator for DTN Protocol Evaluation. In Proceedings of the 2nd International Conference on Simulation Tools and Techniques, Rome, Italy, 2–6 March 2009; pp. 55:1–55:10.
76. Varga, A. The OMNET++ discrete event simulation system. In *Proceedings of the European Simulation Multiconference; SCS—European Publishing House: Prague, Czech Republic, 2001*; pp. 319–324.
77. Riley, G.F.; Henderson, T.R. The ns-3 network simulator. In *Modeling and Tools for Network Simulation*; Springer: Berlin, Germany, 2010; pp. 15–34.
78. Ciobanu, R.I.; Marin, R.C.; Dobre, C. MobEmu: A Framework to Support Decentralized Ad-Hoc Networking. In *Modeling and Simulation in HPC and Cloud Systems*; Kołodziej, J., Pop, F., Dobre, C., Eds.; Springer International Publishing: Cham, Switzerland, 2018; pp. 87–119. [[CrossRef](#)]
79. Papanikos, N.; Akestoridis, D.G.; Papapetrou, E. Adyton: A Network Simulator for Opportunistic Networks. 2015. Available online: <https://github.com/npapanik/Adyton> (accessed on 24 December 2018).
80. Breslau, L.; Estrin, D.; Yu, H.; Fall, K.; Floyd, S.; Heidemann, J.; Helmy, A.; Huang, P.; McCanne, S.; Varadhan, K.; et al. Advances in network simulation. *Computer* **2000**, 59–67. [[CrossRef](#)]
81. Udugama, A.; Förster, A.; Dede, J.; Kuppusamy, V.; Bin Muslim, A. OPS—An Opportunistic Networking Protocol Simulator for OMNeT++. In Proceedings of the International OMNeT++ Summit, Bremen, Germany, 7–8 September 2017.
82. Stefa, J.; Mei, A. Swim: A simple model to generate small mobile worlds. In Proceedings of IEEE INFOCOM Conference, San Diego, CA, USA, 24–31 July 2009.
83. Boldrini, C.; Passarella, A. HCMM: Modelling Spatial and Temporal Properties of Human Mobility Driven by Users’ Social Relationships. *Comput. Commun.* **2010**, *33*, 1056–1074. [[CrossRef](#)]

84. Förster, A.; Bin Muslim, A.; Udugama, A. TRAILS—A Trace-Based Probabilistic Mobility Model. In Proceedings of the 21st ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWIM), Montreal, QC, Canada, 28 October–2 November 2018.
85. Hernández-Orallo, E.; Cano, J.C.; Calafate, C.T.; Manzoni, P. FALCON: A new approach for the evaluation of opportunistic networks. *Ad Hoc Netw.* **2018**, *81*, 109–121. [[CrossRef](#)]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).