Intelligent Environments 2016
P. Novais and S. Konomi (Eds.)

Named Data for Mobile AdHoc Networks

Ana Filipa PEREIRA [a], Maria João NICOLAU [a,1], António COSTA [a],
Joaquim MACEDO [a], Alexandre SANTOS [a]
[a] *Centro Algoritmi, Universidade do Minho, Portugal*

Abstract. For the needs of future Internet, many researchers are proposing a paradigm shift towards a new information-centric approach. The Named-Data Networking (NDN) is one these proposals. Instead of addresses, packets should use only names, either expressing interest on specific information or naming the data content carried in. Applying the NDN model in a mobile ad-hoc network can greatly simplify the routing, since there is no need for global addresses and all nodes in the network can cache data packets.

In this paper we propose a new strategy, called MultiPoint Relay (MPR) Strategy, targeted to minimize redundancy in Interest and Data packet forwarding in Ad-Hoc NDN networks. Interest packets sent by each node are only retransmitted by a subset of selected relay neighbors, as in OLSR. A time delay is also used to detect and prevent duplicated transmissions. Data packets can either be forwarded using the reverse path or by using similar delay technique in case of topology changes. The proposed strategy was implemented in ndnSIM simulator and compared with related works. Obtained results show that the strategy is effective in improving interest satisfaction ratio with reduced network overhead.

Keywords. Ad-Hoc Networks, Named-Data Networks, Forwarding

1. Introduction

A Mobile AdHoc NETwork (MANET) [1] is a self-configuring network where the nodes have the ability to autonomously create a communications network between them without the assistance from a network infrastructure. Nodes in MANETs are wireless and mobile, which makes the MANET topology highly dynamic. Routing is a difficult task, that has to be performed by all nodes. The infrastructure independence of MANETs makes them suitable for a variety of applications like emergency situations, support in natural disasters or military conflicts, or simply in the spontaneous sharing of information in meetings or classes.

Named Data Networking (NDN) [2] is a new architecture designed to meet current and future needs of the Internet. NDN shifts communication paradigm from host-centric to data-centric. This new paradigm differs from the current Internet in some important aspects. First, all contents are identified following a naming scheme based on URIs. Second, the communication is driven by the consumer. When a user needs data, it shows interest in it by sending an interest

---

[1] Corresponding Author: E-mail:joao@dsi.uminho.pt

packet. When an interest packet reaches a node that has the desired content (the producer), a data packet is sent back. Third, storing packets on the network facilitates content delivery. NDN provides native support for mobility since there is no association between the identification and the location of information.

The NDN model has the potential of overcoming some of MANET problems caused by high mobility. Disruptions and route breaks are frequent and may result in network partition making node to node communication difficult to establish and maintain. Nodes can communicate based on the data they need, instead of determining a route to a specific node. This can greatly simplify the implementation of routing in these environments. In this paper we explore a new data-centric forwarding strategy for mobile ad-hoc network scenarios. The proposed strategy, called MPR (MultiPoint Relay) Strategy is based on OLSR, a well known protocol for AdHoc routing. However the MPR Strategy was completely adapted to the data-centric paradigm inheriting all its advantages. The proposed strategy was implemented and compared with others showing gains in terms of interest satisfaction.

The rest of the paper is structured as follows. Section 2 presents a brief overview of relevant related work. Section 3 discusses the proposed strategy in detail. Section 4 details the implementation efforts in *ndnSIM* [3]. Section 5 presents and discusses the results obtained in simulation. Finally the conclusions and future work are presented in Section 6.

2. Related Work

The original forwarding strategy proposed for Named Data Networks is presented in [4]. Interest packets are broadcasted through the network using flooding, while data packets follow the reverse path used by the interest packets that reached the producer. The advantage of this approach is its robustness under mobility and intermittent connectivity. However, it causes large overhead in MANETs due to redundant data propagation. NAIF (Neighborhood-aware forwarding interest) [5] was proposed as an improvement to [4]. Instead of broadcast, NAIF uses a cooperative approach to forward interest packets. A node gradually lowers its forwarding rate if it hears its neighbors answering with corresponding data packets. Conversely, when a node detects it has dropped too many interest packets, it increases its forwarding rate to compensate.

In [6] LFBL (Listen First, Broadcast Later) is presented. LFBL forwarding strategy is based on a distance table maintained in each node. Only nodes closer to a packet's destination than the previous sender are eligible to be forwarders. In addition, a node listen to the channel, waiting to see if other node just send the same packet first. Like LFBL, E-Chanet [7] proposes a distance based forwarding strategy. However, unlike LFBL, E-Chanet is targeted to NDN and presupposes the existence of NDN tables (Forwarding Information Base (FIB), Pending Interest Table (PIT) and Content Store (CS), adding a new Distance Table (DT) to them.

Finally, in [8], an evaluation of NDN forwarding strategies for MANETs is presented. In order to achieve the evaluation, the authors divide the various forwarding strategies into two classes: blind and aware forwarding. Blind forwarding, include the simpler strategies, whose aim is to avoid or minimize the broadcast storm problem. The forwarding strategies that use additional information (like distance tables in E-Chanet) were placed in the second class. Then a representative strategy of each class was implemented in ndnSIM [3]. Based on the evaluation, a set of guidelines for the design of forwarding strategies for named data MANETs is given.

3. MPR Strategy Description

In ad-hoc environments, strategies based on flooding, generate a large number of duplicated packets that have a major impact on communication performance, because the increased traffic leads to packets collisions and losses. The proposed strategy, called *MPR Strategy*, aims to contribute to overcome or at least minimize some of these problems. It is based on the following main principles: (i) Flooding can be improved by explicitly selecting the subset of relay neighbors to use as forwarders; (ii) When using broadcast channels, it is possible to overhear packets in flight and abort the transmission if duplicated packets are listen.

MPR Strategy is based on OLSR, Optimized Link State Routing Protocol [9], a proactive link-state routing protocol for MANETs. The key idea behind OLSR is the use of MultiPoint Relays (MPRs). Instead of using all neighbors to disseminate topology information, each node must select a subset of them to be responsible for forwarding the link state advertisements (LSA). The nodes contained in this subset are called MPRs. In this way a significant flooding reduction may be achieved. MPRs are chosen from the set of neighbors using Hello messages. Hello messages are used for each node to discover its 2-hop neighbors. Nodes should select MPRs such that there exists a path to each of its 2-hop neighbors via a node selected as an MPR. These MPR nodes then source and forward link state advertisements. The other neighbors, who were not chosen as MPR, simply discard them. The list of MPRs chosen is included in the link state advertisement in order to allow a node who receive such a message find out what to do with it. If he finds its ID in the MPRs list, the node forwards the packet, otherwise it discards it. Fig. 1 illustrates the MPR mechanism.

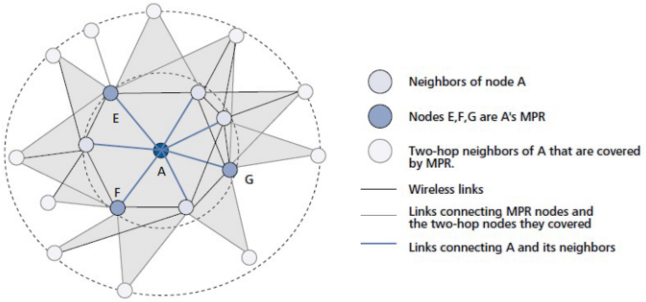


Figure 1. MPR Mechanism (figure adapted from [10])

The same idea of using MPRs may be applied to interest forwarding in order to reduce the interest flooding overhead. In NDNs each node maintains three structures: Forwarding Information Base (FIB), Pending Interest Table (PIT) and Content Store (CS), to process interest and data messages. A node (called consumer) sends out an interest message with a name to retrieve desired data. After receiving an interest message, nodes check local CS. If desired data is found, this interest is satisfied and a data message containing name and data is generated and is transmitted back, along interest message's reverse path to consumer. Otherwise interest message is added in PIT and is forwarded until desired data is found or interest message TTL expires. We applied the MPR mechanism to the interest forwarding strategy in order to reduce redundant interest message and improve the performance of our protocol. In this way, before forwarding an interest packet, each node must select its MPRs and include them in the interest message. Then when one of its neighbors receive the interest message, it looks to the list of MPRs contained in the interest message to see if it appears there. If so, it concludes that it is a selected MPR, and must forward the interest packet. Otherwise it drops the packet.

In addition to the MPR mechanism, the proposed strategy also uses a timer-based message suppression technique. The basic idea is simple. Instead of forwarding immediately an Interest or Data message, the node delays the transmission for a random time interval. During that time interval the node listens to the radio channel to see if it can hear the same packet again, being forwarded by some other neighbor node. If so, the message is dropped and the transmission aborted, instead of being forwarded. This type of flooding control, without any need of state information, may bring several advantages in ad-hoc environments.

3.1. Hello Protocol

Each node in the network needs to know all its 2-hop neighbors in order to select the MPRs. To discover its 2-hop neighbors, the proposed strategy relies on an Hello protocol. Each node has to send Hello messages periodically to all its neighbors. Hello messages should contain the node identifier (ID) of the sender, and the IDs of all its known neighbors. The first time, it will contain only the node ID that is sending the message, since it does not have knowledge of its neighbors. Thus this algorithm requires two iterations to converge. In order to support the Hello Protocol, all nodes must maintain a new table, called Neighbors Table (NT), containing the IDs of all its neighbors and also the number and the corresponding IDs of the 2-hop neighbors (neighbors of neighbors).

To fit in the context of named data networks, the Hello messages should be transmitted through interests and data messages. For this, all nodes should be consumers and simultaneously producers of the "*/hello*" prefix. As a consumer, each node periodically sends interests in the "*/hello*" prefix with the time interval of 5 seconds. Since all its neighbors that receive "*/hello* interest message are also producers of the same prefix, they don't forward it further. Instead, they read its own Neighbors Table and answer the request with a data message. The data message contains the node ID and the ID of alls its neighbors registered in the Neighbors Table (NT). Upon receiving the data message, the requester uses the received information to update its own neighbors table.

In order to avoid collisions, nodes must delay the transmission of data messages of the Hello Protocol for a time interval T_{Hello} calculated randomly in the interval $[0..5ms]$. The maximum value for this timer ($5ms$) is the expected RTT. In addition, a lifetime is set to each neighbor entry in the neighbors table. A timer called $T_{Neighbor}$ is started when an entry is created or updated. If the timer expires, the corresponding information also expires. The value of this timer is based on equation $T_{Neighbor} = Hello_{interval} + T_{Hello} + RTT$, where $Hello_{interval}$ is the time interval between Hello messages ($5s$), T_{Hello} is the forced transmission delay ($5ms$, max value), and finally RTT is the expected round trip time ($5ms$).

3.2. MPRs Selection

From the Neighbors Table (NT), updated by the Hello Protocol, each node must choose a subset of neighbors which will be used as MPRs. The MPRs Selection algorithm uses some additional data structures: (i) a vector with the 2-hop neighbors, initialized to empty and which should contain in the end all 2-hop neighbors; (ii) a vector with the MPRs, also initialized to empty and which should contain in the end the MPRs that will be used; and (iii) an auxiliary table that will be used in case there is a draw in the selection of MPRs. This auxiliary table will include an entry for each neighbor with (1) the neighbor ID, (2) the number of neighbors of this neighbor, (3) a list with the neighbors of this neighbor that are not already in the vector with the 2-hop neighbors attainable by the already selected MPRs.

First, the node will have to sort the Neighbors Table in descending order of number of neighbors, placing on the top of the list the neighbors with most neighbors. Then, the algorithm analyses the first neighbor in the ordered Neighbors Table and verify if there are more neighbors with the same number of neighbors. If so, the algorithm moves all of them to the auxiliary table filling the corresponding fields according to actual content of the vector with the 2-hop neighbors already attainable. If not, that is, there is one unique neighbor with most neighbors, then if it contains neighbors that do not belong to the vector of 2-hop neighbors already attainable, it must be an MPR, and its neighbors should be placed in that vector. Otherwise the algorithm proceeds. The second step will deal with the auxiliary table, and look for the neighbor in this table with more not attainable neighbors. One of them must be chosen for MPR and its not attainable neighbors should be placed in the 2-hop neighbors vector. That neighbor must be removed from the auxiliary table and the remaining entries updated. The neighbors that were inserted in the 2-hop neighbors vector should be removed from the corresponding entries in the auxiliary table and if, in sequence of this, the list with the neighbors that are not already in the 2-hop neighbors already attainable vector by the already selected MPRs became empty, the corresponding neighbor must be removed. This process continues until the auxiliary table became empty. In the same way the neighbors in the Neighbors Table should be all analyzed in order to select all that may become MPRs.

3.3. Interests Forwarding

When a node has an interest to send, it needs to place the set of MPRs selected in a new field of the header of the interest packet, referred to as "*MPR field*". Besides the "*MPR field*" and the other standard fields, we propose another "*Route field*" that carries the ID of the nodes that are part of the route used by the interest message. Each node that receives the interest packet adds its own ID to the route filed. This field is used in the data message forwarding process as it will be explained in the next section.

When a node receives an interest message and it concludes that it should forward it (for example, because it does not have the corresponding content in its Content Store), it must find out if its ID is included in the "*MPR field*" of the interest message. If so, the node is a MPR and it should forward the interest message. Otherwise, the node should drop the interest message. If the node is a MPR and should forward the message, it does not forward it immediately. Instead, the node waits for a while, listening to see if any other node is also transmitting the same interest message, in which case the node gives up that transmission. The time interval that a node waits is calculated by the equation $T_{Interest} = (DW + random[0, DW]) * DeferSlotTime$. According to [8], a study about the impact of these parameters in the forwarding strategies, the values for DW and $DeferSlotWindow$, in various scenarios, should be $127\mu s$ and $28\mu s$ respectively.

During the period of $T_{Interest}$, the interest message remains in a queue of delayed messages, while the node listens to the channel. For each interest messages it receives, a search is made in this queue, to see if there are any interests on hold with the same prefix. If so, the state of validity of these interest will be changed to *false*, so that the forwarding of these interests will be aborted when $T_{Interest}$ expires. The cancellation of the interest messages forwarding could be performed at this time, however, if the node received another equal interest, then it would not be possible to detect that it was a duplicate interest. So it was decided that a node always wait until $T_{Interest}$ expires to make its cancellation, in order to detect any duplicate interests, who arrived in the time interval. When $T_{Interest}$ expires, the corresponding validity field is checked. If it is false the interest message is dropped otherwise it is forward after updating MPR and Route fields.

3.4. Data Forwarding

When an interest message arrives to a producer, or to a node that has the corresponding data in its *Content Store*, it will have to copy the route traversed by the interest message, which will be stored in the "Route" header field of the interest message, and put it in the same field in the data message header before sending it. All nodes that receive the data message will have to verify if they belong to the route. If so, they will forward the message immediately, without any delay. Otherwise, a timer is started with a delay time defined by $T_{Data} = (DW + random[0, DW]) * DeferSlotTime$. The procedure is similar to the one previously described to forward interest messages. T_{Data} is less than the time delay $T_{Interest}$, to give a higher priority to the transmission of data packets.

The main objective of this new functionality is to make the data message use the same path as traveled by the interest message in order to save time and reduce the RTT, whenever possible. However, because of nodes mobility, it is possible that the route has been broken. For this reason it was decided that a node that received a data packet, and did not belong to the route, could even so forward it. In this case, the node will have to wait a period equal to T_{data} and the decision on forward that data message will be positive only if during this time, the node does not receive an equal data message. Therefore, it will be possible to deliver a data message to the consumer even if the route used by the interest message has become invalid.

4. Implementation

The proposed MPR Strategy was implemented on ndnSIM [3], an open source NDN simulator, based on NS-3 [11]. So far, most studies that used ndnSIM have focused on wired networks an not on Mobile Wireless Networks. Perhaps for this reason, multi-hop wireless communication is not supported by the official current version of ndnSIM. In this simulator, a node can forward a packet of interest/data received through all network interfaces except through the interface used by the arrived packet. As a result, a packet received by the 802.11 interface would not be sent directly over the same interface. Thus, the first task we had was to adapt the simulator in order to work around this issue and support to relay traffic through the 802.11 radio interfaces.

Then we decided to implement the proposed strategy in an incremental way in order to evaluate the impact of each of its features in the Forwarding Strategy performance. We begin to implement BF (Blind Forwarding) Strategy as it was proposed by [8]. The BF Strategy is a broadcasting scheme which delays Interest and Data transmissions in order to limit the collision probability and packet redundancy, using packet overhearing. With this strategy we can better evaluate the impact of using time delay $T_{Interest}$ in the interest packets forwarding and T_{Data} in the data packet forwarding.

Then we modified the Data Forwarding Scheme in order to make the data message use the same path that was traversed by the interest message and we called this strategy the DSR Strategy, because it was inspired by Dynamic Source Routing (DSR) [12] Protocol used in AdHoc Networks. With this new strategy we can further evaluate the impact of making the data packets use the reverse interest path, as a complement of the simple delay approach of BF.

And finally we implemented the Hello Protocol and MPRs Selection Algorithm to achieve the Interest Forwarding Scheme of MPR Strategy. MPR Strategy also combines the features of the previous implemented forwarding strategies.

5. Performance Evaluation

Performance evaluation was done by simulation using ndnSIM [3]. The *MPR Strategy* was compared with *BF Strategy*, *DSR Strategy* and *Broadcast*. The

Broadcast Strategy is a simple baseline, based on pure broadcast, without any mechanism to avoid redundant transmissions.

Simulation parameters used on all simulations are presented in Table 1. Distinct simulation scenarios were tested, changing both the number of producers P and the number if consumers C in the same way, with $C, P \in \{2, 7, 12, 17, 22\}$. Consumers generate Interest request at a constant rate of one per second. Consumers only start at time 6s in order for MPR tables to converge. The simulation ends after 204 seconds. A total number of 198 interest are therefore originated by each consumer during simulation. The number of packets in simulation increases with the number of producers and consumers. This is used to verify the behavior of each strategy when the number of redundant transmissions can potentially increase.

Table 1. Simulation parameters

Parameter	Value used
Area	600m x 1500m
Number of Nodes	80
Mobility Model	Random Waypoint
Transmission Range	160m
Traffic Type	*Constant Bit Rate (CBR)*
Packet size	64$bytes$
Number of Flows	15
Transmission Rate	2$kbps$
WiFi Specifications	IEEE 802.11b, freq. 2.4GHz. Transfer rate up to 2$Mbps$
Simulation time (s)	204s

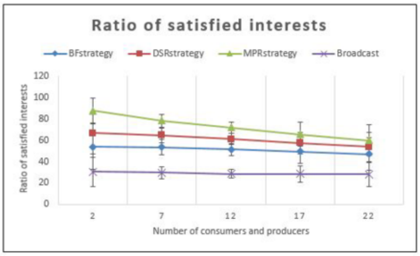
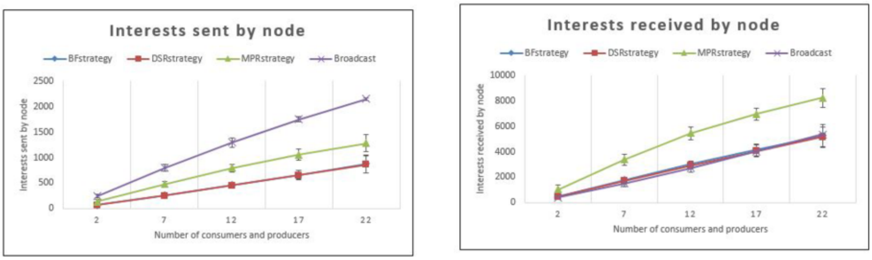


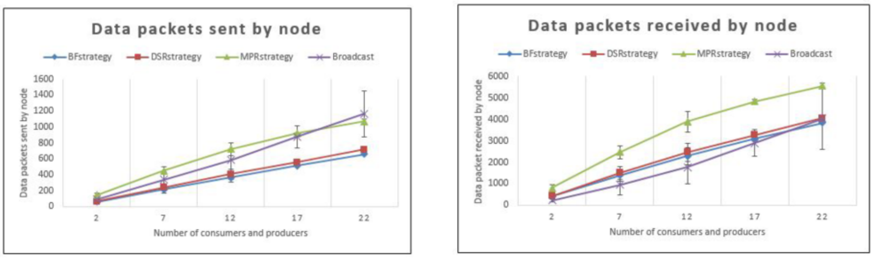
Figure 2. Interests satisfied (in percentage)

Fig. 3a and Fig. 3b show respectively the average number of interest packets sent and received per node. As we can see, both *BF Strategy* and *DSR Strategy* have similar results in these metrics because they use exactly the same forwarding strategy for Interest packets. *Broadcast Strategy* is the one that sends a greater number of interest packets, because it does not use any mechanism at all. All packets are retransmitted without any restrictions. Despite that fact, the *Broadcast Strategy* has the same average number of received interests per node, than the other two strategies: *BF Strategy* and *DSR Strategy*, due to the number of collisions that it originates. The *MPR Strategy* has the higher value regarding interest packets received, while it significantly reduces the number of Interest packets sent. This shows that it achieves the goal of reducing the number of unnecessary transmissions, and thus more distinct packets can be transmitted.

Fig. 3c and Fig. 3d show respectively the average number of data packets sent and received per node during the simulations. The strategy with less data packets received is *Broadcast Strategy*, because it has also the worst interest satisfied rate (Fig. 2). Most Interest Packets do not reach the destination, because network is overloaded, and less data packets are sent back. The *MPR Strategy* is the one with higher number of data packets received, since it is also the one that achieves the higher interest satisfied rate. Regarding the percentage of interest that is satisfied (Fig. 2, we can see that when the number of producers and consumers increases, the performance of all strategies decreases. That's an expected behavior since traffic increases and the number of duplicate packets also increases. The *DSR Strategy* has better results, regarding this metric, when compared to *BF Strategy*,



(a) Avg No of Interest Packets Sent (b) Avg No of Interest Packets Received



(c) Avg No of Data Packets Sent (d) Avg No of Data packets Received

Figure 3. Av number of Interest and Data Packets sent and received per node

because the later floods the data packets to the network when forwarding them, while the former tries to follow the reverse path back to the consumer. The *MPR Strategy* is the one that presents better results, because of the relay selection mechanism. On the other hand, the *Broadcast Strategy* presents obviously the worst results.

6. Conclusions

In this work we address the problem of routing and forwarding of packets in Mobile AdHoc networks using Named Data Network paradigm. A new MPR strategy was proposed, adapted from OLSR in order to better fit the NDN model. Each node that receives an Interest packet and does not have a matching content in the content store, first selects a minimum subset of neighbors, called multi-point relays, that can reach all its 2-hop neighbors. Only those nodes further forward the Interest packets. This results in a more efficient usage of the communication medium. An additional delay mechanism was added to further avoid unnecessary transmissions. Nodes wait and listen for identical packets before transmitting. Without topology changes, Data packets are forwarded trough the reverse path with no further delays. If there are topology changes, data packets are forwarded in a way similar to the one used to forward the Interest packets. This results in faster recovery from path failures. The proposed strategy was implemented in ndnSIM [3] simulator and evaluated in several simulation scenarios. The size of the network and the number of producers and consumers were the parameters considered in the evaluation. Results show an increase in the interest satisfied ratio. The number of interests satisfied, in percentage, was in the range of 60% to 80%.

An interesting experiment planned for a near future is to compare this NDN approach with the host-centric traditional models. Another planned action is to address Quality of Service issues. In previous work [13] we proposed QMRS, a QoS routing protocol that uses a on-demand route discovery mechanism to find up to three node-disjoint paths that meet the QoS requirements. The goal is to be able to compare the current approach with this one. Another future task is to modify the probabilistic routing strategy that we have proposed in [14] for delay tolerant ad-hoc NDNs, in order to include this MPR Strategy.

Acknowledgments

This work has been supported by COMPETE: POCI-01-0145-FEDER-007043 and FCT – Fundação para a Ciência e Tecnologia within the Project Scope: UID/CEC/00319/2013.

References

[1] Imrich Chlamtac, Marco Conti, and Jennifer J.-N. Liu. Mobile ad hoc networking: imperatives and challenges. *Ad Hoc Networks*, 1(1):13–64, 2003.

[2] L. Zhang, A. Afanasyev, J. Burke, V. Jacobson, k. claffy, P. Crowley, C. Papadopoulos, L. Wang, and B. Zhang. Named Data Networking. *ACM SIGCOMM Computer Communication Review*, 44(3):66–73, Jul 2014.

[3] S. Mastorakis, A. Afanasyev, I. Moiseenko, and L. Zhang. ndnSIM 2.0: A new version of the NDN simulator for NS-3. Technical Report NDN-0028, NDN, January 2015.

[4] V. Jacobson, D. K. Smetters, J. D. Thornton, M. Plass, N. Briggs, and R. Braynard. Networking named content. *Commun. ACM*, 55(1):117–124, January 2012.

[5] Yu-Ting Yu, R B Dilmaghani, S Calo, M Y Sanadidi, and M Gerla. Interest propagation in named data manets. In *Computing, Networking and Communications (ICNC), 2013 International Conference on*, pages 1118–1122, January 2013.

[6] Michael Meisel, Vasileios Pappas, and Lixia Zhang. Listen first, broadcast later: Topology-agnostic forwarding under high dynamics. In *Annual conference of international technology alliance in network and information science*, page 8, 2010.

[7] M. Amadeo, Molinaro A., and G. Ruggeri. E-chanet: Routing, forwarding and transport in information-centric multihop wireless networks. *Computer Communications*, 36(7):792–803, 2013.

[8] M. Amadeo, C. Campolo, and A. Molinaro. Forwarding strategies in named data wireless ad hoc networks: Design and evaluation. *Journal of Network and Computer Applications*, 50:148–158, 2015.

[9] T. Clausen and P. Jacquet. Optimized link state routing protocol (olsr). *RFC3626*, 2003.

[10] Xiaoyan Hong, Kaixin Xu, and Mario Gerla. Scalable routing protocols for mobile ad hoc networks. *Network, IEEE*, 16(4):11–21, 2002.

[11] ns-3 project. *ns-3 Manual*. ns-3 project, June 2012.

[12] David B. Johnson, David A. Maltz, and Josh Broch. Dsr: The dynamic source routing protocol for multi-hop wireless ad hoc networks. In *In Ad Hoc Networking, edited by Charles E. Perkins, Chapter 5*, pages 139–172. Addison-Wesley, 2001.

[13] Tiago Coelho, António Costa, Joaquim Macedo, and Maria João Nicolau. Improving stability in qos routing for ad-hoc networks. *LNICST*, 140:121–133, 2014.

[14] P. Duarte, J. Macedo, A.D. Costa, M.J. Nicolau, and A. Santos. A probabilistic interest forwarding protocol for named data delay tolerant networks. *LNICST*, 155:94–107, 2015.