HAL
open science

Neighborhood Discovery with Activity Monitoring in Multichannel Mesh Networks

Dorra Abdelali, Fabrice Theoleyre, Abdelmalik Bachir, Andrzej Duda

To cite this version:

Dorra Abdelali, Fabrice Theoleyre, Abdelmalik Bachir, Andrzej Duda. Neighborhood Discovery with Activity Monitoring in Multichannel Mesh Networks. WCNC 2010 - IEEE Wireless Communications and Networking Conference, 2010, Sydney, Australia. pp.1-6, 10.1109/WCNC.2010.5506146 . hal-01073413v2

HAL Id: hal-01073413

https://hal.science/hal-01073413v2

Submitted on 15 Apr 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Neighbor Discovery with Activity Monitoring in Multichannel Wireless Mesh Networks

Dorra Abdelali† Fabrice Theoleyre† Abdelmalik Bachir§, Andrzej Duda†

†Grenoble Informatics Laboratory, Grenoble, France
Email: dorra.abdelali@imag.fr, fabrice.theoleyre@imag.fr, andrzej.duda@imag.fr

§Imperial College, London, UK
Email: a.bachir@imperial.ac.uk

***Abstract*—One way of improving performance of wireless mesh networks is to use multiple non-overlapping channels. At the same time, the mesh network must continuously self-adapt to varying radio conditions and topology changes. Thus, we propose a neighbor discovery protocol that fits the requirements of multichannel networks while dealing at the same time with the deafness problem. The proposed scheme also takes advantage of neighbor discovery to continually monitor channel activity. We provide a theoretical analysis of the average discovery time. Our simulations show that the solution integrated within Molecular MAC, a multichannel MAC, results in efficient discovery and channel assignment.**

***Index Terms*—multichannel mesh networks, neighbor discovery, channel assignment**

I. Introduction

We consider *wireless mesh networks* with routers based on the IEEE 802.11 technology. When mesh routers use a single network interface, the performance of packet forwarding quickly degrades with the number of hops due to channel contention and spatial problems such as hidden, exposed, masked, and blocked nodes [1]. We can observe that the capacity of a wireless mesh network strongly depends on the ability of nearby mesh routers to communicate in parallel, which is only possible if neighbor routers that may interfere, use different channels.

One way of improving performance is to use multiple non-overlapping channels so that stations can transmit in parallel and without collisions. When a mesh router has several radio interfaces, it can tune them to different channels and use them simultaneously. Choosing the best channels while maintaining the network globally connected is an optimization problem that can be solved in a centralized [2] or a distributed way [3].

Taking advantage of multiple channels is also possible when mesh routers only have single network interfaces—they can dynamically switch channels to use the best channel at a given instant for data transmission. Obviously, to operate in this way, the network must implement a cooperation mechanism so that the pair of communicating nodes knows the channel chosen for transmission. If it is not the case, the so-called *deafness* problem arises: when the receiver is tuned to another channel, the sender cannot send its packet. Several solutions exist to avoid the deafness problem: nodes can set up periodical rendezvous [4], [5], organize transmissions in a pseudo-random manner [6], or structure the network in a specific way [7].

A wireless mesh network may evolve during its operation when some mesh routers join or leave the network. Moreover, the radio environment may change because of obstacles so some links may appear or disappear. It is thus vital to cope with topology changes, even if they are not so frequent as in the MANET type of networks. Autonomic operation of wireless mesh network calls for a distributed channel allocation protocol that strongly depends on the ability of detecting current neighboring nodes. If we consider environments without a common control channel or periodical rendezvous, mesh routers will require a specific multichannel neighbor discovery protocol.

Surprisingly, to the best of our knowledge, the neighbor discovery problem in multichannel networks has never been studied. However, this kind of networks presents specific challenges compared to classical mesh networks. In particular, a single `hello` packet is insufficient to discover all the neighbors because of deafness: some nodes may listen to different channels. Thus, we need to adapt and extend classical neighbor discovery to the multichannel case.

Our contribution is fourfold:

1) we present a neighbor discovery protocol that addresses the deafness problem and fits the requirements of multichannel networks,
2) we analyze the protocol to derive the average discovery delay and tune the parameters of the protocol,
3) to support channel allocation, we augment the protocol with a simple way of measuring channel activity,
4) we apply the protocol to Molecular MAC [7] and evaluate its performance when it uses a simple channel allocation algorithm inspired by a previous work [8].

Although we consider the proposed protocol in the context of Molecular MAC [7], we can easily extend it to other schemes as we will explain in the rest of the paper.

II. Problem statement

A multihop wireless network requires a neighbor discovery protocol: a node must learn its neighbors and evaluate the

quality of radio links. In classical mesh networks (single-channel case), each node usually broadcasts periodical `hello` packets so that the receiver can add the source of the packet to its neighbor table. This scheme does not require any synchronization and is self-healing: a node that stops receiving `hellos` from a neighbor, just removes it from its table. Moreover, a node can distinguish between bidirectional and unidirectional links by including the list of nodes from which it receives `hellos` [9]. A node can also estimate the link quality by monitoring the number of received `hellos` and computing the packet delivery rate [10].

Neighbor discovery is much more complex in multichannel mesh networks. In particular, when a node transmits a `hello` packet, there is no guarantee that all its neighbors receive it, since they may listen to other channels. One solution is to replicate `hello` and send one `hello` packet per channel. However, during this operation, the transmitter is deaf to data transmissions on its main channel. Thus, this scheme needs to minimize the discovery interval to limit deafness while allowing for accurate discovery of new neighbors.

A. Molecular MAC

We start with shortly presenting Molecular MAC [7] since we base one part of our study on this multichannel MAC protocol. Molecular MAC follows a per-packet channel switching approach and can operate when nodes have only one network interface. The network is divided into atoms containing one nucleus and several electrons (Fig. 1). All communications in an atom pass through a nucleus that acts a sort of an access point for the atom. A nucleus chooses a fixed channel for all communications in an atom. All nodes in an atom contend for the channel according to the standard IEEE 802.11 DCF access method. To avoid interference, neighbor atoms use orthogonal channels. An electron may belong to several atoms and forwards frames between different nuclei: it continuously switches its channel to tune to the channels of corresponding nuclei.

Molecular MAC solves the deafness problem without requiring any synchronization scheme. An electron can send its frame to the nucleus, because it knows the channel used by the nucleus. Communications in the opposite direction require notification from a nucleus about pending frames. To notify electrons, a nucleus maintains a list of destinations for which it has at least one packet and piggybacks it onto all its `data` packets. Electrons have to periodically listen to the channel of each neighboring nucleus to receive piggybacked notifications. The notified electrons send a `Pull` frame that results in receiving a pending frame from a nucleus. To avoid starvation, an electron probabilistically switches its channel for sending data packets or receiving notifications.

Simulations showed that Molecular MAC outperforms existing multichannel single-interface MAC schemes with respect to the offered throughput and delay [7]. Further work proposed a method for assigning roles (nuclei and electrons) to nodes and creating a proper network structure for efficient packet forwarding [11].

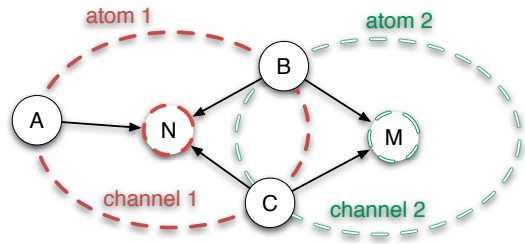

Fig. 1. Molecular architecture

III. Neighbor Discovery in a Multichannel Mesh Network

We present below the proposed neighbor discovery for multichannel mesh networks by adapting a classical scheme. We will also explain how it integrates activity measurement. To illustrate the scheme, we take Molecular MAC as an example of a multichannel MAC.

A. Neighbor Discovery

The main principle of the discovery scheme is the following: a node that wants to discover its neighbors switches to a channel for interval T_{disc}, broadcasts a `hello` packet, and waits for a reply. The `hello` packet includes information about the sender (id, role—nucleus or electron, atom state—channel chosen by the nucleus) and the list of neighbors to discover unidirectional links. The node sequentially changes channels to discover all possible neighbors. As there may be several nodes that receives the `hello` packet, they reply after a randomly distributed time interval to avoid simultaneous transmissions. Nodes activate discovery each T_{cycle} interval, so a node scans each channel at least once during this period.

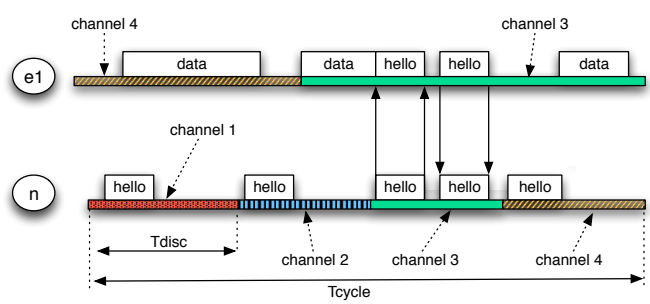

Fig. 2. Example of discovery by node n that tries to join the network.

Let us consider the case of a node that wants to join the network. As it needs to first integrate the network structure before participating in packet forwarding, it will continuously perform the discovery scans until a positive response from a node, either a nucleus or an electron. `hello` packets contend for the channel as all other packets. Figure 2 illustrates this process: node n tries to join the network and scans four channels. Electron e_1 operates in the network by forwarding packets on channels 3 and 4, and when it receives the `hello` packet on channel 3, it replies with a corresponding `hello` that allows node n to learn about electron e_1, its neighbors, and channels. The role that will further adopt node n depends on the discovery of other nodes—if it discovers the nucleus

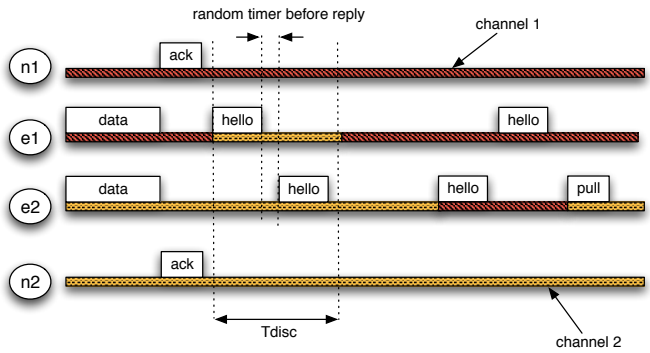

Fig. 3. Example of neighbor discovery done by electrons (n_1 and n_2 are nuclei while e_1 and e_2 are electrons)

operating on channel 3, it may become another electron, or it can create another atom by becoming a nucleus.

Let us consider the case of neighbor discovery during the operation of the network. Nodes need to periodically discover neighbors, because of topology changes—some links may disappear and others appear. In the case of Molecular MAC, we restrict neighbor discovery to electrons, because a nucleus would become deaf if it switches its channel to discover neighbors. Nevertheless, nuclei may benefit from the information gathered by electrons, even if they do not participate in discovery, since they have often a common neighboring electron when the network is sufficiently dense. This restriction does not influence the network structure, because usually nuclei are far from each other, at least by two hops, to minimize interference.

Figure 3 illustrates neighbor discovery in this case. Electrons scan the channels of their atoms, while their nuclei remain on fixed channels. The main role of electrons is to forward packets, but periodically, they enter discovery phase to cope with potential topology changes. To introduce some randomness in the process and avoid possible synchronizations of discovery times for two neighbors, the period between two discovery phases is a sum of a random jitter and T_{cycle}, where the jitter is chosen uniformly between $[-\frac{T_{cycle}}{10}, \frac{T_{cycle}}{10}]$.

When an electron is idle, i.e., it has neither a `data` frame to send nor `data` to receive (it did not receive any notification of pending data frames from its nuclei), it initiates a discovery phase to accelerate neighbor discovery while limiting the impact on data traffic.

B. Multichannel Activity Measurement

As a node has to scan each channel to discover neighbors, it can also measure its activity. A node can gather the following statistics:

1) number of received/transmitted data frames ($\mathcal{D}ata(atom)$): an electron can estimate the amount of forwarded traffic. We can in this way obtain a measure of the activity per channel per atom;
2) collision rate ($\mathcal{C}oll(k)$): a node that retransmits a frame on channel k because of a timeout may consider that its packet suffered from a collision;
3) interference ($\mathcal{I}(k)$): a node considers as interference a signal on channel k above the carrier sense threshold but not decodable.

We can distinguish between channels used for data forwarding (we call them *active channels*) and the ones explored during neighbor discovery that we call *passive channels*.

Obviously, we need to associate weight with these statistics, because otherwise the activity would be over-estimated for active channels: a node listens most of the time to these channels so it will capture more activity. Thus, each node computes the activity every T_{cycle} and measures the duration T_i for which it was listening to channel i. The node will weight each activity estimation with this duration. Let $\mathcal{C}$ be the set of channels, the activity on channel k will be the following:

$$\mathcal{A}ct(k) = \frac{T_k}{\sum_{i \in \mathcal{C}} T_i} \left(\sum_{atom\ uses\ channel\ k} \mathcal{D}ata(atom) + \mathcal{C}oll(k) + \mathcal{I}(k) \right) \quad (1)$$

In Molecular MAC, each nucleus chooses its fixed channel in function of the traffic of other atoms. Thus, a nucleus can extract a global measure of activity for each channel in the same way. However, it discards the traffic from its own atom (i.e., only external interference is taken into account).

We can observe that such an activity measure can be easily extended to the multiradio multichannel case. Any node will initiate neighbor discovery and measure the activity in the same manner. The only difference consists in classifying the data traffic per neighbor and not per atom since this concept only exists in Molecular MAC.

IV. Analysis of the Discovery Time

We will now analyze the average discovery time of neighbors in multichannel mesh networks. We will focus here on Molecular MAC [7] to provide a comprehensive description. However, this study can be easily extended to more general cases as we will highlight it further.

We assume that the traffic forwarded by an atom is uniformly distributed over all available channels and nodes operate asynchronously. Besides, we assume that a `hello` packet is always correctly received if the receiver and the transmitter use the same channel, i.e., we neglect collisions and time used by `data` packets.

A. Preliminaries

When an electron is not in discovery mode, it listens to one of its active channels. We neglect collisions and time used by `data` packets. Let $\gamma = \frac{T_{disc}}{T_{cycle}}$ be the ratio of the discovery time to the cycle duration. Probability $P_{listen}(e,k)$ that an electron e listens to a given channel k is:

$$P_{listen}(e,k) = \frac{1}{|\mathcal{C}|} \cdot \frac{T_{cycle} - T_{disc}}{T_{cycle}} = \frac{1}{|\mathcal{C}|}(1-\gamma). \quad (2)$$

Probability $P_{disc}(e,k)$ that an electron is in discovery mode on a given channel k is:

$$P_{disc}(e,k) = \frac{T_{disc}/T_{cycle}}{|\mathcal{C}|} = \frac{\gamma}{|\mathcal{C}|}. \quad (3)$$

We can highlight three different cases for the discovery between two neighbor nodes:

1) two nuclei: since a nucleus has a static channel to avoid deafness, two nuclei will never discover each other. If so, such a case would be rare since the nuclei are usually far away to reduce interference;
2) nucleus–electron: since an electron has to scan all the channels, it can discover a nucleus that never switches its channel. The average discovery time is consequently $\frac{T_{cycle}}{2}$;
3) two electrons: one electron must discover the other one when it is either in listen or in discovery mode.

B. Discovery of Two Electrons

Let us now examine the probability that two electrons discover each other, i.e., $P_{disc_2elec}(e_1, e_2)$. If e_1 activates neighbor discovery on channel c, it will succeed to discover e_2 only in the following cases:

- e_2 is in discovery mode on the same channel,
- e_2 is in listening mode on the same channel.

Therefore, probability p that e_1 and e_2 discover each other is:

$$\begin{aligned} p &= P_{disc}(e_1, k) \cdot P_{listen}(e_2, k) \\ &+ P_{disc}(e_1, k) \cdot P_{disc}(e_2, k) \\ &+ P_{disc}(e_2, k) \cdot P_{listen}(e_1, k). \end{aligned} \quad (4)$$

By combining Eqs. 2, 3, and 4 we get:

$$p = \frac{2\gamma - \gamma^2}{|\mathcal{C}|^2}. \quad (5)$$

Probability $\bar{p}$ that they do not discover each other is:

$$\bar{p} = 1 - p. \quad (6)$$

Therefore, probability $\bar{P}$ that there will be no discovery on $|\mathcal{C}|$ channels is:

$$\bar{P} = \bar{p}^{|\mathcal{C}|} = (1-p)^{|\mathcal{C}|}. \quad (7)$$

Finally, probability P that e_1 and e_2 discover each other on all channels is:

$$P = (1 - \bar{P}) = 1 - (1-p)^{|\mathcal{C}|}. \quad (8)$$

Figure 4 presents the impact of the number of channels and the ratio dedicated for neighbor discovery on the average discovery time. If $\gamma = 0$, the discovery time becomes infinite since the electrons never send `hello` packets. For large values of γ, two neighbors will discover each other, but with an increasing overhead, i.e., the nodes spend more time to send `hellos`. We clearly have a trade-off between the overhead of `hellos` and the discovery delay. We can note that the number of channels only slightly impacts the discovery time.

Through this analysis, we are able to compute the overhead associated with a given discovery time. We are consequently able to tune the parameter values of the neighbor discovery protocol.

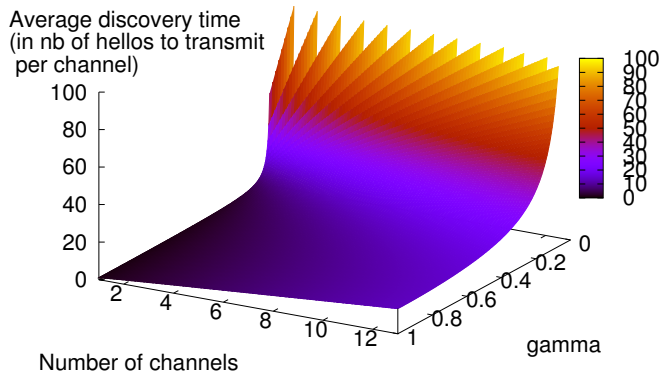

Fig. 4. Average duration to discover a neighbor for $T_{cycle} = 1s$.

C. Generalization for the Multichannel Multiradio Case

We can easily extend this study to the multichannel multiradio case: the set of active channels becomes the set of static channels used by all radio interfaces, other channels being considered as inactive. Besides, a node can aggregate the neighbor discovery information obtained on all interfaces: if it scans one channel using one interface, there is no need to rescan it on another interface. Consequently, we just obtain different numerical values after deriving the same equations. Because of the lack of space, we will not derive them here.

V. Channel Assignment

To take advantage of our activity measurement scheme, we propose to integrate in Molecular MAC a simple channel assignment protocol inspired by a previous work [8]. Its authors proposed to assign channels in a distributed way to different interfering WLAN cells: each Access Point monitors the activity in its vicinity and accordingly chooses its channel in a probabilistic manner. If the channel results in bad performance, the AP will surely change its channel later and the global scheme will converge to an interference-free allocation. Thus, our protocol will act in the same way: each atom will monitor the channel activity and will assign the best channel on the basis of the measure.

Obviously, we have to adapt the original protocol to take into account the specificity of Molecular MAC. First, a nucleus in Molecular MAC cannot directly measure the activity on all the channels since it would become deaf to data transmissions. Only electrons can dynamically switch their channels. Thus, a nucleus first collects the information from all its electrons and assumes that the aggregated measures reflect well the activity of all nodes interfering with its atom.

The exchange of information follows a classical scheme: a nucleus broadcasts a `channel_activity_request` and each electron replies with a unicast `channel_activity_reply` transmitted after a random interval to avoid simultaneous transmissions. An electron must reply with an activity measure reflecting the ability to use the corresponding channel for all the transmissions in the atom. Thus, the `channel_activity_reply`

TABLE I
SIMULATION PARAMETERS

Bit rate	54Mb/s
Packet reception threshold	-86dBm
Transmission power	5mW
RTS/CTS	inactive
Buffer size	256 kbits
Packet size	1,500 bytes
T_{cycle} (discovery cycle)	1s
γ (ratio of discovery time to the cycle duration)	from 0.5% to 3.5% (depends on the number of channels)

includes a metric that aggregates the number of erroneous frames and the number of `data` frames received/sent for each channel. As explained previously, an electron excludes the traffic generated by the nucleus that has sent `channel_activity_request`. Indeed, the nucleus needs to take into account only the traffic of other atoms for its channel allocation decision.

Finally, the nucleus can decide on the best channel for its atom: to maximize the fairness, it chooses the channel that is the least used among its electrons. It thus chooses the channel that minimizes the maximum load monitored by its electrons.

A. Performance Evaluation

We have simulated the proposed neighbor discovery scheme and Molecular MAC in OPNET [12] with the parameters presented in Table I. We have evaluated the performance of our channel assignment protocol (called here *localized*). We have compared it with a random strategy in which each node randomly chooses one channel independently of the load (called *random*). Data traffic consists of several constant-bitrate (CBR) flows. We have averaged the results presented below and plotted 95% confidence intervals. We have run simulations with and without the RTS/CTS option and obtained results that are not significantly different, so we have decided not to represent them in the figures. A synchronized protocol with a common control channel would allow two nodes to discover each other in one `hello` period. However, such a scheme does not work for an unsynchronized node that joins the network, which is the case in spontaneous wireless mesh networks. Besides, to the best of our knowledge, no protocol achieves such a feature. Thus, we cannot compare our results with another existing scheme. We use the algorithm to assign the roles to the nodes developed earlier [11].

First, we have simulated a simple network of six nodes in line and measured the associated delay (cf. Figure 5). The delay reflects well the impact of the traffic on packet forwarding: when the network operates near its capacity, the delay increases. In particular, the traffic is too high to be forwarded using a single channel. This explains the large delay for small number of channels. When nodes have a small number of channels, they have a large probability to collide with each other when channels are allocated randomly. This explains why the delay is larger than with our channel assignment method that locally estimates the load on each channel. When we have a large number of channels, the impact is much less important, which explains a lower delay. We can

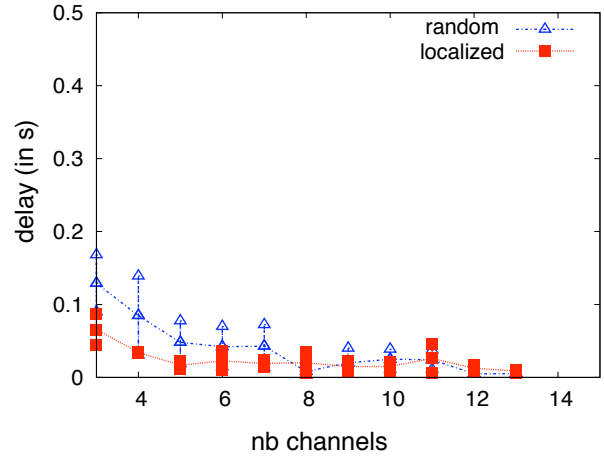

Fig. 5. Delay in a network with six nodes in line, inter-packet time of 10ms

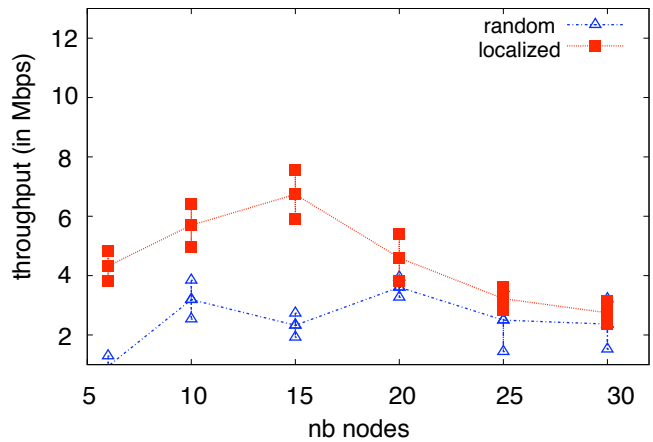

Fig. 6. Throughput in a random mesh network with 5 channels, an inter-packet time of 10ms

also verify that the delay is much smaller when a node takes into account the sampled measures of activity on its channels.

Then, we have simulated a random mesh topology (nodes are placed randomly) and varied the number of nodes while maintaining a constant average node degree of 8 (each node has eight neighbors on the average) and with nodes having 5 channels (cf. Figure 6). We have measured the throughput, i.e., the rate of data packets that arrive to destinations. We can note that the *localized* protocol optimizes the throughput: the random scheme cannot avoid interference, which has a severe impact on performance. Besides, we can observe a local maximum for 15 nodes: more nodes require different nodes to re-use the same channels with more collisions. Moreover, routes are longer and require more capacity.

VI. RELATED WORK

Neighbor discovery is a classical problem in wireless multihop networks and is widely considered as a basic building block of any protocol. Solutions are often based on transmission and reception of `hello` packets [9]. Some solutions extends this scheme to cope with specific situations. For instance,

Vasudevan et al. deal with directional antennas: since neighbor discovery depends on the direction of the antenna, they introduced a probabilistic approach [13]. Some solutions do not require `hello` packets. In particular, Luo et al. proposed a low-level solution in which all nodes simultaneously send a unique sequence of on-off micro-slots that correspond to their unique signatures (equivalent to a sequence of bits) [14]. Thus, neighbor discovery becomes a group testing problem. To the best of our knowledge, all the schemes were not extended to the multichannel case.

Multichannel single-interface MAC protocols have recently drown considerable attention. MMAC [4] proposed to introduce a reservation period using the IEEE 802.11 ATIM window during which all the nodes reserve the channels to use for data transmissions. Classical neighbor discovery may happen during the ATIM window, but would reduce the time let for data transmissions and reservations. This kind of protocols can implement a classical neighbor discovery protocol since they periodically use a common control channel. However, the control channel may quickly become a bottleneck. Moreover, such protocols require a global synchronization scheme.

SSCH [6] proposed a pseudo-random hopping sequence: a node switches its channel following a pseudo-random sequence. By publishing its seed, all its neighbors can compute the same channel sequence and know when their own schedule will coincide. However, the authors do not discuss neighbor discovery. Moreover, the scheme requires fine grained synchronization so that all nodes perform frequency hopping synchronously.

Finally, Molecular MAC [7] also proposed a dynamic multichannel MAC by creating a molecule structure as described previously. However, the authors do not discuss the problem of multichannel neighbor discovery.

Several authors proposed protocols and algorithms for multichannel multiradio mesh networks by assigning channels to limit interference [15], [16], [17]. However, the problem corresponds to a static graph coloring problem and they do not discuss topology changes. Such solutions assume that each node a priori knows each of its neighbors or that nodes perform an initial neighbor discovery on a common control channel. Thus, such solutions do not take into account topology changes.

VII. Conclusion & Future work

We have presented neighbor discovery for multichannel wireless mesh networks. We have proposed to couple neighbor discovery and activity measurement: each node monitors channel activity by sampling the traffic on all channels. By adopting a systematic channel switching strategy to discover neighbors, we were able to provide a probabilistic analysis of the average discovery delay. We have verified with a simple distributed channel assignment algorithm that the measurement strategy correctly measures channel activity and helps to decide which channel to use. We plan to extend this work to deal with cognitive networks [18] and study how to adapt the discovery time as well as investigate the best strategies to efficiently discover new neighbors.

References

[1] C. Chaudet, D. Dhoutaut, and I. Guerin Lassous, "Performance Issues with IEEE 802.11 in Ad Hoc Networking," *IEEE Communications Magazine*, vol. 43, no. 7, 2005.

[2] H. L. Nguyen and U. T. Nguyen, "Channel Assignment for Multicast in Multi-channel Multi-radio Wireless Mesh Networks," *Wireless Communications and Mobile Computing*, 2008.

[3] S. Avallone and I. Akyildiz, "A Channel Assignment Algorithm for Multi-Radio Wireless Mesh Networks," *Computer Communications*, vol. 31, no. 7, 2008.

[4] J. So and N. H. Vaidya, "Multi-Channel MAC for Ad Hoc Networks: Handling Multi-Channel Hidden Terminals Using a Single Transceiver." in *Proc. MOBIHOC*. Tokyo, Japan: ACM, May 2004.

[5] W.-T. Chen, T.-K. Huang, Y.-C. Chang, and J.-C. Liu, "An Adaptive Multi-Channel MAC Protocol for Wireless Ad Hoc Networks," in *Proc. ICC*. Istanbul, Turkey: IEEE, 2006.

[6] V. Bahl, R. Chandra, and J. Dunagan, "SSCH: Slotted Seeded Channel Hopping For Capacity Improvement in IEEE 802.11 Ad Hoc Wireless Networks," in *Proc. MOBICOM*. ACM, 2004.

[7] M. Nassiri, F. Theoleyre, M. Heusse, and A. Duda, "Molecular Architecture for Autonomic Wireless Mesh Networks," in *Proc. MASS*. Macau: IEEE, October 2009.

[8] D. Leith and P. Clifford, "Channel Dependent Interference and Decentralized Colouring," in *Proc. NETCOOP*, ser. LNCS, 2007.

[9] T. Clausen, C. Dearlove, and J. Dean, "MANET Neighborhood Discovery Protocol (NHDP)," IETF, draft version 10, July 2009.

[10] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris, "A High-Throughput Path Metric for Multi-Hop Wireless Routing," in *Proc. MOBICOM*. ACM, September 2003.

[11] B. D. Fabrice Theoleyre and A. Duda, "Assignment of Roles and Channels for a Multichannel MAC in Wireless Mesh Networks," in *Proc. ICCCN*. San Francisco, USA: IEEE, August 2009.

[12] OPNET version 12.0.A, http://www.opnet.com.

[13] S. Vasudevan, J. Kurose, and D. Towsley, "On Neighbor Discovery in Wireless Networks with Directional Antennas," in *Proc. INFOCOM*. Miami, USA: IEEE, March 2005.

[14] J. Luo and D. Guo, "Neighbor Discovery in Wireless Ad Hoc Networks Based on Group Testing," in *Proc. Annual Allerton Conference*, September 2008.

[15] A. Raniwala and T. Chiueh, "Architecture and Algorithms for an IEEE 802.11-based Multi-Channel Wireless Mesh Network," in *Proc. INFOCOM*. IEEE, March 2005.

[16] P. Kyasanur and N. H. Vaidya, "Routing and Interface Assignment in Multi-Channel Multi-Interface Wireless Networks," in *Proc. WCNC*. IEEE, 2005.

[17] M. Alicherry, R. Bhatia, and L. E. Li, "Joint Channel Assignment and Routing for Throughput Optimization in Multi-Radio Wireless Mesh Networks," in *Proc. MOBICOM*. ACM, 2005.

[18] N. Mittal, S. Krishnamurthy, R. Chandrasekaran, S. Venkatesan, and Y. Zeng, "On Neighbor Discovery in Cognitive Radio Networks," *Journal of Parallel and Distributed Computing*, vol. 69, no. 7, July 2009.