Molecular MAC for Multichannel Wireless Mesh Networks

Mohammad Nassiri*†, Fabrice Theoleyre*, Martin Heusse*‡, and Andrzej Duda*

* Grenoble Informatics Laboratory, Saint Martin d'Hères, France
Email: {firstname.lastname}@imag.fr
† Bu-Ali Sina University, Hamedan, Iran
‡ University of Colorado, Boulder, USA

***Abstract*—We propose a novel view on efficient packet forwarding in wireless mesh networks based on a *molecular* analogy in which mesh routers are either *nuclei* or *electrons* in an *atom* that corresponds to a 802.11 cell. In this view, a mesh network appears as a collection of spatially distributed 802.11 cells operating on different channels. We define *Molecular MAC* that uses dynamic channel switching at neighbor mesh routers to efficiently forward packets over multiple hops. To avoid deafness, nuclei *notify* electrons about pending packets before electrons *pull* them from nuclei for reception or further forwarding. We evaluate the proposed scheme through simulation and compare with other proposals. Our results show that Molecular MAC obtains much better performance in terms of throughput, packet delivery rate, end-to-end delay, and fairness.**

I. Introduction

We consider spontaneous *wireless mesh networks* that use IEEE 802.11 wireless LANs for interconnecting mesh routers. They form *multihop* wireless networks that convey user traffic on behalf of client nodes. This kind of network is particularly useful for deploying community networks and can extend wireless coverage of classical wireless networks while reducing deployment costs, because only a small subset of mesh routers need to connect to the Internet via wired or long-haul wireless connections.

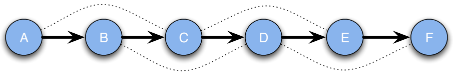


Fig. 1. Chain topology, single channel case.

The IEEE 802.11 technology has gained wide popularity when deployed in the infrastructure mode with an Access Point providing untethered connectivity to nomadic devices. However, there is a lot of evidence that the 802.11 MAC layer is not suitable for multihop wireless networks: when nodes use the same channel, the performance of packet forwarding over multiple hops quickly degrades with the number of hops due to channel contention and spatial problems such as hidden, exposed, masked, and blocked nodes [1], [2], [3]. Consider the chain topology in Figure 1: we assume that neighboring nodes are within their transmission range while 2-hop neighbors interfere (dashed lines link interfering nodes). In this topology, when node A transmits a frame to B, it blocks node C so it cannot transmit to D at the same time. The throughput of packet forwarding in this topology quickly declines in function of the number of nodes [1].

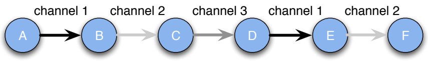


Fig. 2. Chain topology, multiple channel case.

If neighboring nodes use different channels, they can operate in parallel (cf. transmissions A-B and C-D in Figure 2) and obtain much better throughput. Using different channels can also improve throughput in other unfavorable configurations such as the topology of *3 parallel pairs* [1] illustrated in Figure 3. In this setup, parallel pairs are within interference range, so that if all nodes use the same channel, two external pairs can transmit in parallel. The pair in the middle will hardly ever transmit, because two other pairs monopolize the channel. When the middle pair uses a different channel, all nodes can transmit in parallel and obtain better throughput.

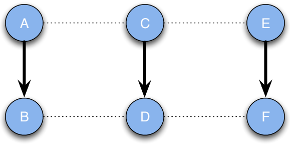


Fig. 3. 3 parallel pairs, single channel case.

Mesh routers can take advantage of parallel transmissions over neighbor links by using multiple channels at

This work was partially supported by the European Commission project WIP under contract 2740 and the French Ministry of Research project AIRNET under contract ANR-05-RNRT-012-01.

978-1-4244-5113-5/09/$25.00 ©2009 IEEE

various time scales. When nodes have multiple interfaces, they can statically allocate channels to achieve high spatial reuse and good performance [4], [5]. Nodes with single interfaces can switch channels on a per frame basis, however they face in this case the problem of *deafness* when a node tries to send a frame on one channel while the intended receiver is listening to or is sending on another one. In this paper, we consider the latter case—we focus on wireless mesh networks with nodes with single interfaces that switch channels for each frame transmission or reception. Although we have designed our scheme for single interface nodes, our approach generalizes to nodes with multiple interfaces.

Several authors considered mesh networks with nodes that switch between multiple channels. MMAC (Multi-Channel MAC) proposes to create periodical RendezVous Points on a common control channel during which each pair of nodes reserves the channel used for their next data transmissions [6]. MMAC improves performance in the chain topology, but sometimes nodes need to wait for the next rendezvous to reserve a channel. Similarly, in the topology of 3 pairs MMAC reduces the probability of the middle pair having the same channel as the other pairs, but does not completely solve the problem. SSCH (Slotted Seeded Channel Hopping) [7] divides the time into super-frames composed of slots. Each node chooses a pseudo-random hopping sequence published by broadcasting a seed. A node switches its channel during each time slot according to the pseudo-random sequence and can communicate with a neighbor if they use the same channel. In the chain topology, the schedules of neighboring node do not perfectly overlap so that some slots are wasted, which results in poor performance. In the topology of 3 pairs SSCH can obtain good performance only if both nodes in each pair use exactly the same schedule. MMAC and SSCH present however the advantage of being compatible with the 802.11 standard.

In this paper, we propose *Molecular MAC*, a modification of the standard IEEE 802.11 DCF access method that uses dynamic channel switching to efficiently forward packets over multiple hops. The idea is to partition a mesh network into spatially distinct *atoms* that are similar to traditional IEEE 802.11 infrastructure mode cells: their *nuclei* always use a chosen channel that is different from a neighbor nucleus. Nodes between two nuclei, which we call *electrons*, dynamically switch channels to communicate with neighbor nuclei.

Molecular MAC builds on good efficiency of IEEE 802.11 DCF in the infrastructure mode while extending it to operate in mesh networks following the dynamic multichannel approach. Molecular MAC achieves efficient multiplexing of transmissions across orthogonal channels while avoiding the classical deafness problem.

The contribution of this paper is twofold. First, we present a new efficient MAC mechanism for forwarding packets using dynamic channel switching in a wireless mesh network. We base our work on 802.11 and the DCF access method, although we can similarly extend any other random access method. Second, we evaluate the proposed mechanism through simulation and compare with other schemes. Molecular MAC outperforms them in terms of throughput, packet delivery rate, end-to-end delay, and fairness. We also provide some details about the construction of the molecular architecture and topology requirements.

II. Molecular approach

The approach of Molecular MAC to dynamic channel switching profoundly differs from MMAC and SSCH. Unlike these proposals, it aims at taking advantage of long-lasting spatial reuse of channels, so that the whole mesh network looks like a collection of spatially distributed 802.11 cells operating on different channels. Let us explain the approach below (we assume mesh routers with single network interfaces).

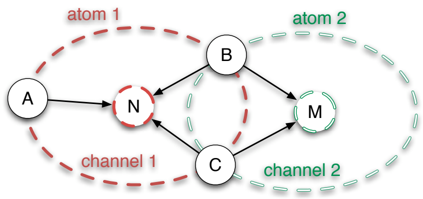


Fig. 4. Two atoms sharing two electrons.

We adopt a *molecular* analogy to organize a wireless mesh network (cf. Figure 4). An *atom* is a basic entity for constructing a mesh network. It is composed of a *nucleus*, a mesh router that uses a fixed channel for communicating with *electrons*, its immediate neighbors. Atoms bond together to form a *molecule*, which corresponds to a connected wireless mesh network. Electrons belong to adjacent atoms and communicate with their nuclei. Figure 4 illustrates this view. Mesh routers N and M are nuclei of two atoms bonded by two mesh routers corresponding to electrons B and C. There is no direct link between two electrons so to communicate, they use a nucleus like an access point in a 802.11 WLAN. Two nuclei do not directly communicate neither, because they use different channels. If they need to communicate, they can do it through neighboring electrons.

To operate in parallel without interference, two neighboring atoms should use orthogonal channels. In this way, parallel frame transmissions will not suffer from contention nor from any undesirable spatial effect such as hidden and exposed nodes. At the same time, we need a means for forwarding packets between atoms

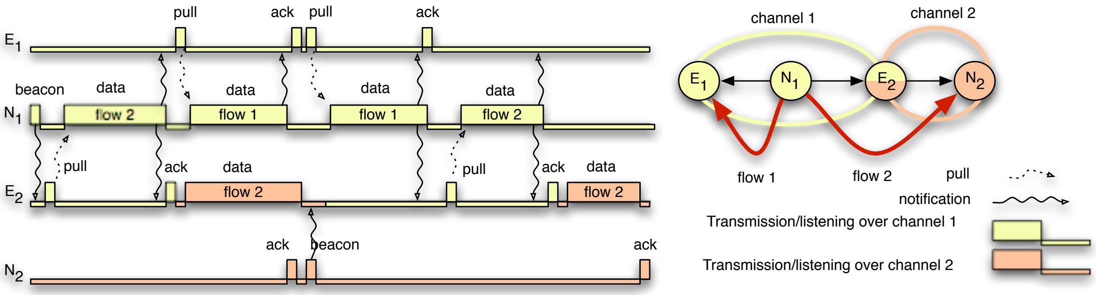


Fig. 5. Pull mechanism and notifications for packet forwarding between nodes E_1 and N_2.

that use different channels. Molecular MAC uses a nucleus as a kind of a *virtual access point*: it is either a receiver or a transmitter of any frame in an atom. Packet forwarding relies on electrons that switch between channels of all neighboring atoms. This scheme assigns one channel to each nucleus and all electrons use it for communication with the nucleus (either for reception or for transmission). Upon construction of a mesh, a nucleus dynamically chooses its channel depending on the channels used in its neighborhood by other atoms. By choosing different channels, two neighboring atoms that would have interfered, if they had used the same channel, can limit interference problems, which contributes to achieving high capacity.

Nodes that use the same channel in an atom (e.g. N, A, B, and C in Atom 1) contend for channel access according to the standard IEEE 802.11 DCF or any of its extensions. Forwarding packets on a path between a source and a destination involves a series of alternating electron and nucleus nodes. For instance, node A can send packets to node M over the path A, N, C, M spanning two atoms that use different channels (cf. Figure 4). The next section presents how we achieve efficient packet forwarding with Molecular MAC.

III. Molecular MAC

To interconnect atoms and achieve efficient packet forwarding electrons need to switch between channels used by neighboring nuclei so that packets going through adjacent atoms benefit from parallel communications. Current IEEE 802.11 802.11a/b/g wireless cards can switch channels in less than $30\mu s$ [8] so electrons can alternate transmissions on different channels on a per-packet basis. However, when neighboring nodes dynamically choose different channels for transmissions, they face the problem of *deafness* when a node tries to send a frame on one channel while the intended receiver is listening or sending on another one.

A. Deafness avoidance

Deafness is a challenging problem when interfaces frequently switch channels. There are two main approaches to solve it: either nodes reserve a common channel for signaling and use it to agree on another channel for a given transmission [9], [10], or use periodical rendezvous points to negotiate which channel to use for further transmissions [6], [7], [11]. Both solutions present drawbacks: a fixed signaling channel means less available resources for data transmission, while periodical rendezvous increases the overhead and requires some form of temporal synchronization between mesh routers.

Molecular MAC solves the deafness problem without these drawbacks: a nucleus chooses its channel, announces it to all electrons, and stays tuned to the channel. An electron belonging to several atoms knows the channels of its nuclei and can switch to a given channel when it wants to communicate. Packet forwarding in this case further requires two functionalities: a nucleus needs to *notify* an electron that there is a packet to forward and the electron needs to *pull* the frame containing the packet from the nucleus. We propose to implement frame notification through two mechanisms: piggybacking on data frames or including the information in periodically sent beacons. We redefine the semantics of the CTS control frame to use it when an electron needs to pull a frame from a nucleus. The role of a nucleus in forwarding reduces to buffering packets for electrons, notifying them that there is something to receive, and transmitting a packet on-demand when an electron asks for a frame.

Figure 5 presents an example of packet forwarding along two flows: Flow 1 between nodes N_1 and E_1, and Flow 2 between nodes N_1 and N_2. To simplify, the figure neglects all backoffs generated by the underlying channel access method. The example starts when node N_1 has a packet to send to electron E_2. It notifies the electron through a beacon on channel 1 that there is a pending frame to pull. Electron E_2 sends the CTS control frame to request the data frame. N_1 sends it and includes piggybacked notification about a pending packet of Flow 1 ready for electron E_1. When N_1 sends this frame containing the packet of Flow 1 on channel 1, electron E_2 can simultaneously forward the packet of Flow 2 to N_2 on channel 2. Note that transmissions do not suffer from deafness, because nuclei always use their channels and electrons pull data frames on a given channel before receiving them on the same one. Moreover, if an electron misses a beacon or a notification from a nucleus, because it is communicating with another nucleus, it can further receive next beacons or notifications when it switches back to another channel.

B. Frame notification

An electron must learn about data frames buffered at a nucleus. As several electrons are potential destinations of the frames, a nucleus needs to maintain a list of frame destinations and advertise it to all electrons. One way of notifying is to piggyback the list of pending destinations onto data frames. We can easily implement piggybacking in 802.11, because the maximal frame size is much larger than the common limitation of 1500 B due to Ethernet compatibility.

When an atom is idle, i.e. there is no traffic to forward, there is also no opportunity to piggyback notifications onto data frames. We thus propose to enhance 802.11 `beacon` frames to include the list of pending destinations in periodic broadcasts. A nucleus must send either a beacon or a notification piggybacked onto a data frame at most every T_b interval. However, to speed up notification, the nucleus can send beacons with a higher frequency (e.g. an interval of several DIFS in the case of DCF), if it is idle, because anyway there is no other traffic to forward (we have used $T_b = 5ms$ in our simulations).

Since an electron may belong to several atoms, it needs to periodically switch its channel to listen to beacons from all neighboring nucleus. Thus, an electron must listen during at most T_N interval to each of its neighboring nuclei. As soon as the electron receives either a `beacon` or a piggybacked notification, the electron can send a `pull` control frame (CTS) to the nucleus to receive the data frame. On the contrary, if the electron does not need to receive a data frame, it can switch to another atom.

When an electron knows that at least one nucleus has a data frame to transmit and at the same time it also has a data frame to send, it randomly chooses between sending or pulling the frame. Such a strategy maximizes fairness and leads to shorter forwarding delays since it does not privilege neither reception nor emission.

When an electron knows that one of its atoms is busy, it can perform other tasks, i.e. listen to other nuclei for notifications or perform neighbor discovery to establish links with nodes entering the mesh (cf. Section IV-B).

C. Optimization: fast reply

To improve throughput and lower the overhead due to piggybacking and pulling, we propose a fast reply mechanism: upon receiving a data frame from an electron, a nucleus can reply with another data frame, if present in its buffer, piggybacking the required acknowledgement. Figure 6 illustrates this sequence of three frames `data/data/ack`, two data frames being transmitted in opposite directions. The scheme reduces the overhead introduced by piggybacking and pulling if flows are bidirectional.

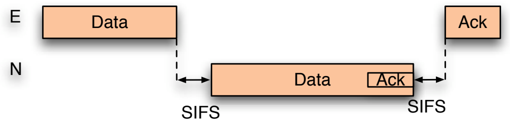


Fig. 6. Fast reply optimization for frame transmission Electron→Nucleus.

D. Multi-interface mesh networks

Although we have originally designed Molecular MAC for mesh routers with single network interfaces, it can also operate without any modification when nodes have multiple interfaces. A node has just to assign the role of an electron or a nucleus to each of its interfaces. Moreover, if we want to achieve the best network throughput, we need to carefully assign roles, for instance, two neighbors that are electrons for their first interface should maintain a radio link with each other for their second interface: one of them should become a nucleus. In this way, we can reduce the route stretch factor: two neighbors will probably act as a nucleus for one of their interfaces and all the radio links can forward traffic.

IV. Molecular topology

In this section, we describe the mesh topology required by Molecular MAC to operate and present the mechanisms for distributed multichannel neighborhood discovery.

A. Molecule construction

Molecular MAC enables efficient packet forwarding over a path of alternating electron and nucleus nodes. In terms of graphs, this means that a molecule mesh network needs to form a bipartite graph. In addition to that, we can add more constraints that aim at guaranteeing connectivity and allocating channels in a way that limits interference.

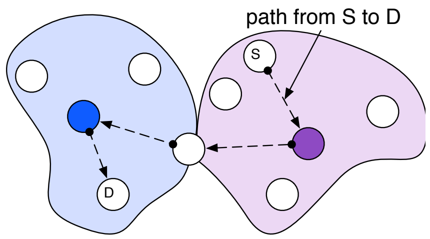


Fig. 7. Molecular architecture.

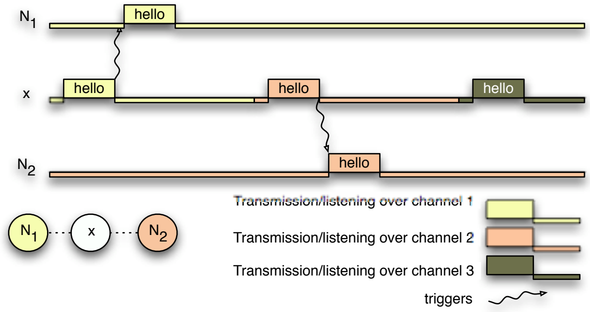
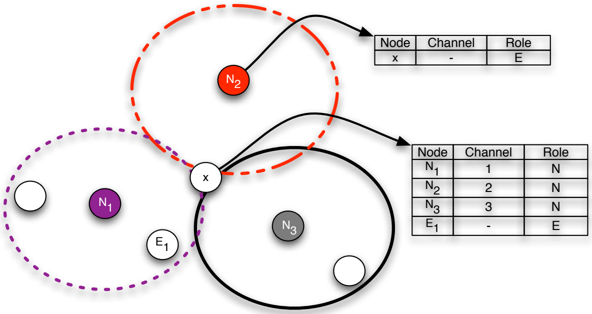

N1
hello
x
hello
hello
hello
N2
hello
N1
x
N2
Transmission/listening over channel 1
Transmission/listening over channel 2
Transmission/listening over channel 3
triggers

Fig. 8. An example of new node *x* joining a molecular mesh network.



Node	Channel	Role
x	-	E

Node	Channel	Role
N_1	1	N
N_2	2	N
N_3	3	N
E_1	-	E

Fig. 9. Content of the neighborhood table.

TABLE I
SIMULATION PARAMETERS

Simulation duration	240s
Bit rate Packet reception threshold Transmit power	54Mb/s -86dBm 5mW
RTS/CTS Buffer size Packet size	inactive 256 kbits 1,500 bytes
T_b: Maximum time between two notifications T_N: Maximum listen time for notification Hello interval (Molecular)	5ms 10ms 1s
Beacon Interval (MMAC) ATIM Window (MMAC)	25ms 5ms

The first problem is role assignment: a node need to decide to operate as a nucleus or an electron so that the resulting molecule has the following properties:

1) a node can communicate with all other nodes via multi-hop forwarding;
2) only nuclei and electrons can directly communicate with each other, i.e. we exclude links between two electrons or two nuclei;
3) the capacity of the network should be maximal. In particular, two neighboring atoms, which can interfere, need to use different channels.

The construction of the mesh molecule is closely related to a well-known graph structure problem—a Weakly Connected Dominating Set (WCDS) [12]. The set of nuclei form a restricted WCDS, i.e. a WCDS in which the graph weakly induced by the edges (nucleus, electron) forms a connected set (cf. Figure 7).

In another paper, we have formulated the problem of electing nuclei and assigning channels as a generic MILP (Mixed Integer Linear Programming) problem [13]. Its solution leads to the optimal assignment of roles and channels in a molecular mesh. We have proposed a centralized protocol that provides an upper bound for constructing the molecular structure and two distributed self-stabilizing heuristic protocols derived from the protocols for constructing respectively a Maximum Independent Set and a Spanning Tree.

Once a node has decided on its role, a nucleus has to choose a channel used for all transmissions in its atom. We adopt a distributed channel assignment that aims at minimizing interference. An electron monitors the activity on each channel and reports it to all neighboring nuclei. Thus, a nucleus can just choose the channel that minimizes the maximum activity among all its electrons.

B. Multichannel neighborhood discovery

Molecule construction and deciding on the roles of nodes requires neighbor discovery to learn which nodes are around, what are their roles (nucleus or electron), and which channels the neighbor nuclei use. Our neighbor discovery scheme builds upon periodic broadcasting of `hello` packets: nuclei send them on their fixed channels and electrons on all channels they use. When a node wants to join a mesh molecule, it waits for a `hello` packet on a random channel. If it does not receive any `hello` packet during a certain time, it will scan all the channels by sending a `hello` packet. If a neighbor receives a `hello` packet, it replies with the required information (its channel, role, and the identity of its neighbors). Thus, a node can achieve neighbor discovery in a single scan. Figure 8 shows how new node x joins the mesh. It starts scanning the channels and sends the first `hello` on channel 1. Nucleus N_1 that listens to this channel immediately replies. Then, node x iteratively scans other channels. Figure 9 gives an overview of the content of different neighborhood tables: a node must store the address, role (either nucleus or electron), and the channel used by its neighboring nuclei if it is an electron.

A node considers a link with a given neighbor broken, if it does not receive neither data traffic nor `hello` packets.

V. PERFORMANCE EVALUATION

We have simulated Molecular MAC in OPNET with the parameters presented in Table I. We have compared the performance of our proposal with the standard IEEE 802.11 and MMAC (Multi-Channel MAC) [6]. We assume that for the standard IEEE 802.11 all nodes use one channel in the whole mesh. We have chosen MMAC as

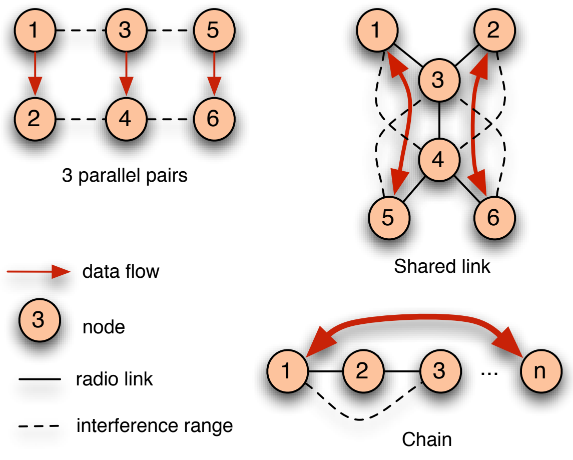


Fig. 10. 3 pairs, shared link, and chain configurations.

one of the representative mechanisms that use dynamic channel switching: it adopts IEEE 802.11 by multiplexing simultaneous transmissions over different channels with one single network interface. To achieve this objective, a node under MMAC makes channel reservations during rendezvous points on a control channel. It publishes reservations through the ATIM window (Announcement Traffic Indication Message) of IEEE 802.11. After a rendezvous, a node switches to the chosen channel and exchanges packets with a given destination.

We have evaluated all three MAC protocols in various configurations: a chain, a shared link, 3 parallel pairs (cf. Figure 10), and a random topology of nodes placed in a circular simulation area. The three basic topologies are well known challenging cases that we consider to gain insight into the behavior of the considered protocols. Data traffic consists of several constant-bitrate (CBR) flows, their rate being represented in the figures below as the offered load in Mb/s. For the basic configurations, Figure 10 presents the source and the destination of each flow. Each simulation runs for a period of 240 seconds. We have averaged the results presented below over 5 different simulation runs and the 95% confidence intervals are within 1% of a given value so we have chosen not to represent them in the figures for better readability. 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.

We have evaluated the performance of three MAC layers according to three following metrics:

1) *Packet delivery ratio*: the ratio of packets received by the destination and the total number of generated packets.
2) *End-to-end delay*: the delay between packet generation and its reception by the final destination.
3) *Aggregated throughput*: the volume of all received data in the network per unit time in Mb/s.
4) *Jain index* defined as:

$$F_J = \frac{(\sum_{i=1}^{n} X_i)^2}{n\sum_{i=1}^{n} X_i^2}, \qquad (1)$$

where X_i is the throughput obtained by flow i, measures throughput fairness of different flows in the network. A low Jain index means poor fairness.

A. Shared link topology

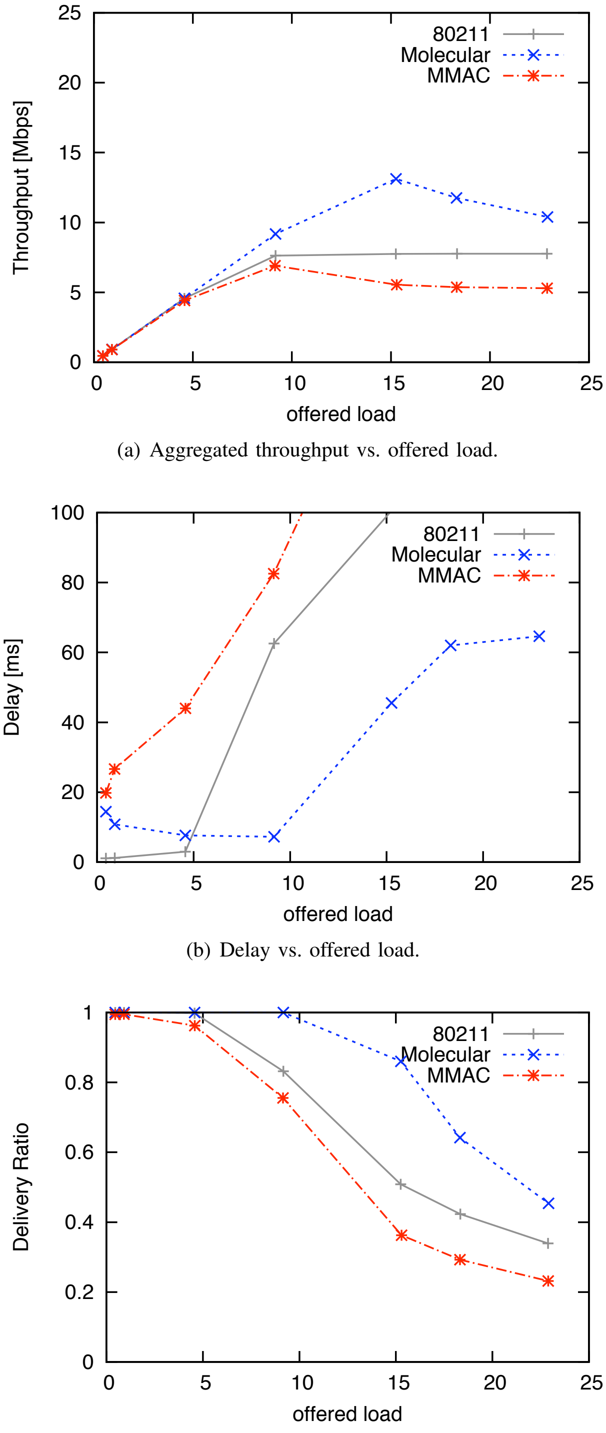
We first consider the shared link topology (cf. Figure 10). Under Molecular MAC, nodes 1, 2, and 4 become nuclei and choose different orthogonal channels while the other nodes become electrons. Figure 11 shows the simulations results in this topology. Since all traffic passes through nodes 3 and 4, the link between them becomes a bottleneck.

Figure 11(a) shows that even in saturation, MMAC obtains less throughput than 802.11 DCF. Indeed, MMAC assumes that nodes have to forward the same amount of traffic during one beacon interval although it is not the case in multihop networks (e.g. link (4,3) must forward more traffic than link (5,6)). Currently, MMAC does not consider the load on each radio link for channel reservation. Moreover, channel reservation of MMAC can become inefficient in some cases due to cascading effects: the choice of a channel implies the choices of other channels. For instance, if nodes 3 and 4 reserve a channel, all other nodes have to choose the same channel, because they have to transmit frames either to node 3 or to node 4. During such a beacon interval, MMAC would perform exactly like a single-channel IEEE 802.11. Conversely, MMAC creates islands isolated from each other during the beacon interval. For instance, it creates two different groups if radio links (3,1) and (4,5) reserve a different channel. Since the number of packets to buffer during a beacon interval can be large, buffer overflows are frequent, which explains the lower throughput compared to IEEE 802.11 in this topology. Accordingly, the delay with MMAC increases sharply with the load (cf. Figure 11(b)).

We can see from Figure 11 that in the shared link topology Molecular MAC outperforms both IEEE 802.11 and MMAC in terms of all indices: throughput, delay, and packet delivery ratio. This last measure decreases fairly fast under IEEE 802.11 and MMAC for increasing load while Molecular MAC still obtains good results for the load of 15 Mb/s. Note that the average end-to-end delay remains acceptable for Molecular MAC even in saturation while for two other access methods this metric increases sharply with the load (cf. Figure 11(b)).

B. Chain

The chain topology often appears in multihop networks when nodes forward a packet along a route [1].



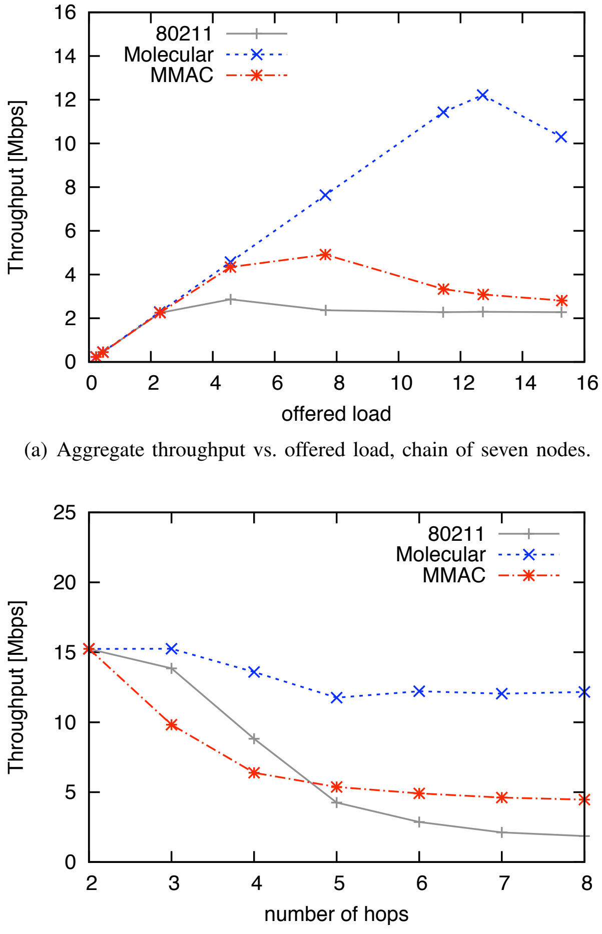
(a) Aggregated throughput vs. offered load.

(b) Delay vs. offered load.

(c) Delivery ratio vs. offered load.

Fig. 11. Simulation results for the shared link topology with two active CBR bidirectional flows between nodes 1-5 and 2-6.

We have simulated a chain with bidirectional traffic (two CBR flows in opposite directions). Under Molecular MAC, the chain becomes a series of alternating electron and nucleus nodes.



(a) Aggregate throughput vs. offered load, chain of seven nodes.

(b) Aggregate throughput vs. length of the chain.

Fig. 12. Simulation results for the chain topology with one active CBR bidirectional flow between nodes 1-7.

Figure 12 shows the aggregate throughput with respect to the offered load in a chain of seven nodes. We can observe that the Molecular MAC achieves much higher throughput than two other methods especially under high load.

We have also studied the impact of the route length on the performance. Figure 12(b) presents the maximal throughput obtained by a given method for a given length of the chain. We can observe that Molecular MAC is almost insensitive to the route length. Conversely, IEEE 802.11 suffers from poor throughput that decreases with the number of hops. MMAC is more scalable than IEEE 802.11, but also obtains lower throughput, because it benefits less from channel diversity than Molecular MAC: channel reservations are dynamical and can create

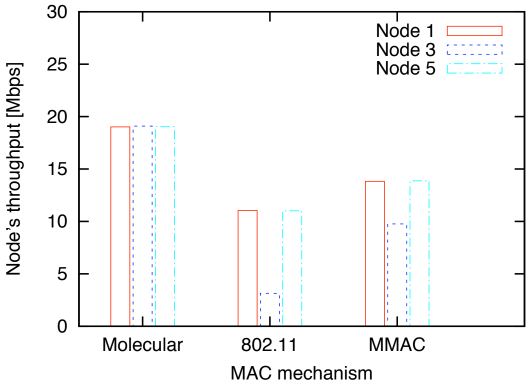


Fig. 13. Throughput in the topology of 3 parallel pairs.

cascading effects as explained in detail above.

C. 3 parallel pairs

In this scenario, three pairs of stations communicate in parallel: 1 → 2, 3 → 4, 5 → 6 (cf. Figure 10). The middle pair senses the carrier of two other pairs, but it is too far to decode their frames. IEEE 802.11 performs very poorly in these conditions: the middle pair is starved while other pairs operate at the maximal throughput [1].

Molecular MAC solves the problem of the unfairness in the 3 pairs topology and decreases the collision probability to 0. Indeed, the three pairs naturally create three different and non interfering atoms. In particular, the atom in the middle chooses a different channel than the other pairs and the extreme pairs uses a different channel than the pair in the middle. With any channel usage measurement mechanism, channel allocation will converge to form such non interfering atoms. We used in this topology two orthogonal channels for Molecular MAC and MMAC.

In this configuration, Molecular MAC provides totally fair allocation to three pairs (cf. Figure 13) while the middle pair suffers from starvation under 802.11 and even under MMAC. The problem of MMAC for this topology is that during channel negotiation, interfering pairs cannot decode ATIM packets. In other words, interfering pairs may end up choosing a common channel, which lowers throughput.

D. Random topology

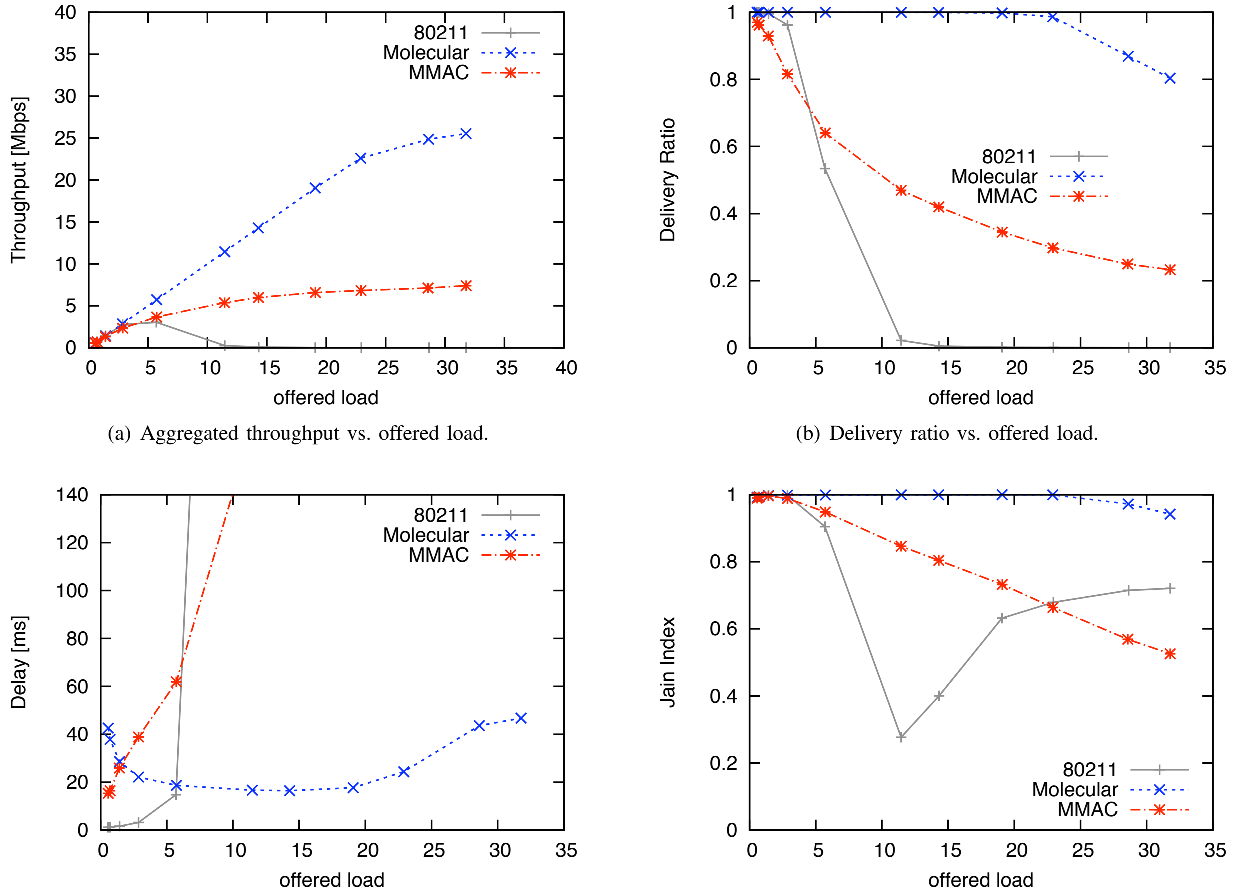
We have also evaluated the performance of the three access methods in more complex topologies corresponding to realistic wireless mesh networks. We have simulated a dense mesh network of 50 nodes randomly deployed in a circular simulation area of radius 270m. Traffic consists of 25 bidirectionnal CBR flows between randomly chosen pairs of source and destination.

To organize the random topology in the molecule mesh network, we have used a geographic-oriented approach to construct a WCDS: we place nodes on a grid composed of sufficiently small squares and we elect one node as a nucleus in each square similarly to the GAF (Geographical Adaptive Fidelity) algorithm [14]. Other nodes automatically become electrons. Since our random network is sufficiently dense, there is at least one electron in each square to interconnect nuclei. Moreover, the small size of a square guarantees that an electron correctly receives packets from neighboring nuclei so that the resulting graph is connected. The molecule construction algorithm elects 12 nuclei on the average in our random topologies.

Figure 14 presents the performance indices in the randomly generated topology. We can observe that when load increases, IEEE 802.11 DCF performs poorly because of spatial problems (hidden and exposed nodes) and increased contention. The aggregated throughput becomes almost null when offered load is larger than 12 Mb/s. On the contrary, MMAC and Molecular MAC are much more scalable, but Molecular MAC outperforms MMAC: while MMAC begins to saturate when the load reaches 20 Mb/s with the throughput of 7 Mb/s, Molecular MAC attains the level of 26 Mb/s in saturation. The throughput of Molecular MAC is roughly 4 times larger than under MMAC. The measures of the packet delivery ratio further show the performance advantage of Molecular MAC over MMAC (cf. Figure 14(b)).

Figure 14(c) presents the end-to-end delay that under Molecular MAC is slightly longer than under IEEE 802.11 or MMAC for low load. This is expected because of the constant overhead experienced by electrons when they pull data frames from nuclei. However when load increases, nuclei can piggyback notifications in data frames so that the delay decreases. For increasing load, the delay under IEEE 802.11 or MMAC quickly becomes fairly long while under Molecular MAC it is short and much more scalable with only a slight increase in function of load—it achieves packet delivery in less than 50ms independently of the load. Even in saturation, Molecular MAC offers an acceptable delay.

Finally, we have evaluated fairness measured by the Jain index (cf. Figure 14(d)). IEEE 802.11 presents the worst fairness for intermediate levels of load: nodes suffer from the inherent unfairness of the exponential backoff further amplified by spatial problems. The Jain index of IEEE 802.11 does increase in saturation, but it is an artifact, because only a small fraction of packets reaches the destination (cf. Figure 14(b)). The fairness of MMAC decreases linearly with increasing load, because MMAC does not fairly share the bandwidth among different flows when a bottleneck appears. Thus, higher



(a) Aggregated throughput vs. offered load.

(b) Delivery ratio vs. offered load.

(c) Delay vs. offered load.

(d) Jain index vs. offered load.

Fig. 14. Random mesh with 50 nodes uniformly distributed within a circular area with the radius of 270 m.

load amplifies the unfairness. Finally, we can observe that molecular MAC achieves almost a perfect fair allocation of capacity among different flows even in saturation. This nice property comes from the fact that electrons equally share their activity among neighboring nuclei.

VI. Related Work

As explained previously, mesh networks based on the IEEE 802.11 suffer from severe performance problems for some common topologies (chain, parallel pairs) [1]. To improve performance several authors have proposed to modify the principles on which the standard IEEE 802.11 protocol builds. Talucci *et* al. [15] have defined the MACA-BI (MACA By Invitation) access method that modifies the MACA method [16] to make it receiver oriented by keeping only the CTS part of the RTS-CTS handshake. However, the destination must be aware of current active flows. RIMA proposes a similar variant of this approach in which a polling node can also send a request to an inactive polled node [17].

Other authors have proposed to simultaneously use several channels to improve network capacity. Proposed solutions mainly rely on a common control channel or frequency hopping to alleviate the deafness problem—we have already introduced MMAC [6] and SSCH [7] in Section I as the most representative protocols. Wu *et al.* [9] were the first to adapt IEEE 802.11 to mesh networks by using a RTS/CTS exchange on a control channel to reserve another one for data transmission. Chen *et* al. [11] further extended this method to allow for broadcast and multicast transmissions and presented a channel allocation algorithm to maximize channel utilization.

Tang *et* al. [18] have proposed a frequency hopping approach in which nodes are organized in groups with a common frequency hopping sequence. A pair of nodes reserves the channel with a RTS/CTS exchange and stay on the same channel for data transmission while other nodes keep on changing their channels. Such an approach requires synchronized nodes and does not completely solve the deafness problem.

Vedantham *et al.* [19] have proposed to assign one channel per flow, two intersecting flows having the same channel. This solution works in single-interface mesh networks, but interfering nodes can use the same channel. In particular, interference between packets of the same flow can reduce throughput as discussed previously.

Molecular MAC shares several similar concepts with Bluetooth. First of all, Bluetooth uses the same kind of the WCDS structure for scatternet construction [20]. The difference lies in the fact that in Molecular MAC a node cannot be both nucleus and electron. Moreover, Molecular MAC and Bluetooth both address the problem of multichannel MAC layers. However, they follow different approaches: while Bluetooth adopts a TDMA-like scheme, Molecular MAC adapts the IEEE 802.11 DCF. We consider that random access methods are more suitable for spontaneous mesh networks carrying bursty traffic. Achieving distributed TDMA scheduling for inter-piconets communications may become difficult in large wireless networks even knowing the capacity of each link and with a stable topology [21]. Moreover, it requires temporal synchronization so that slots allocated by different masters do not overlap. On the contrary, Molecular MAC does not require any synchronization, is robust to traffic variations, does not the number of active nodes in an atom, and is entirely distributed.

VII. Conclusion

In this paper, we have presented a novel view on efficient packet forwarding in wireless mesh networks. By adopting a molecular analogy, we can assign roles to mesh routers so that the resulting mesh is composed of independent atoms using different channels to limit interference. We define an efficient frame forwarding MAC mechanism between adjacent atoms by extending the operation of the standard 802.11 DCF. The modifications to the standard IEEE 802.11 frame format and operation required by our mechanisms are minor: we reuse the CTS control frame for pulling frames as well as beacon frames and piggybacking onto data frames for notifying pending frames.

Our performance comparison with IEEE 802.11 and MMAC shows that Molecular MAC obtains much better throughput and packet delivery ratio, offers short end-to-end delays even in saturation, and exhibits very good fairness. Scalability is also fairly good: the overall performance remains at a high level even for an increasing number of hops or for higher loads.

We continue to work on the design of efficient distributed protocols for molecule construction (role and channel assignment) and its integration with neighbor discovery.

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, pp. 110–116, July 2005.

[2] S. Ray, J. B. Carruthers, and D. Starobinski, "Evaluation of the Masked Node Problem in Ad Hoc Wireless LANs," *IEEE Transactions on Mobile Computing*, vol. 4, no. 5, pp. 430–442, 2005.

[3] A. Iyer and C. Rosenberg, "Understanding the Key Performance Issues with MAC Protocols for Multi-hop Wireless Networks," *Wireless Communications and Mobile Computing*, vol. 6, no. 6, Sep. 2006.

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

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

[6] J. So and N. H. Vaidya, "Multi-channel mac for ad hoc networks: handling multi-channel hidden terminals using a single transceiver." in *MOBIHOC*. ACM, May 2004.

[7] V. Bahl, R. Chandra, and J. Dunagan, "SSCH: Slotted seeded channel hopping for capacity improvement in ieee 802.11 ad hoc wireless networks," in *MOBICOM*. ACM, October 2004.

[8] M. Zannoth, T. Ruhlicke, and B.-U. Klepser, "A Highly Integrated Dual-Band Multimode Wireless LAN Transceiver," *IEEE Journal of Solid-State Circuits*, vol. 39, no. 7, pp. 1191–1195, Jul. 2004.

[9] S.-L. Wu, C.-Y. Lin, Y.-C. Tseng, and J.-P. Sheu, "A New Multi-Channel MAC Protocol with On-Demand Channel Assignment for Multi-Hop Mobile Ad Hoc Networks," in *ISPAN*. IEEE, December 2000.

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

[11] J. Chen and S.-T. Sheu, "Distributed Multichannel MAC Protocol for IEEE 802.11 Ad Hoc Wireless LANs," *Computer Communications*, vol. 28, no. 9, pp. 1000–1013, June 2005.

[12] J. E. Dunbar, J. W. Grossman, J. H. Hattingh, S. T. Hedetniemi, and A. A. McRae, "On weakly connected domination in graphs," *Discrete Mathematics*, vol. 167-168, pp. 261–269, April 1997.

[13] F. Theoleyre, B. Darties, and A. Duda, "Assignment of roles and channels for a multichannel mac in wireless mesh networks," in *ICCCN*. IEEE, August 2009.

[14] Y. Xu, J. Heidemann, and D. Estrin, "Geography-Informed Energy Conservation for Ad Hoc Routing," in *MOBICOM*. ACM, July 2001.

[15] F. Talucci and M. Gerla, "MACA-BI (MACA by Invitation). A Wireless MAC Protocol for High Speed Ad Hoc Networking," in *Proc. of the IEEE ICUPC*. San Diego, USA: IEEE, oct. 1997.

[16] P. Karn, "MACA: a New Channel Access Method for Packet Radio," in *Proc. of the Amateur Radio 9th Computer Networking Conference*. ARRL/CRRL, 1990, pp. 134–140.

[17] J. J. Garcia-Luna-Aceves and A. Tzamaloukas, "Reversing the Collision-Avoidance Handshake in Wireless Networks," in *Proc. of the ACM MOBICOM*, New York, USA, 1999, pp. 120–131.

[18] Z. Tang and J. J. Garcia-Luna-Aceves, "Hop-Reservation Multiple Access (HRMA) for Ad-Hoc Networks," in *INFOCOM*. IEEE, March 1999, pp. 194–201.

[19] R. Vedantham, S. Kakumanu, S. Lakshmanan, and R. Sivakumar, "Component Based Channel Assignment in Single Radio, Multi-channel Ad Hoc Networks," in *MOBICOM*. ACM, sept. 2006.

[20] S. Basagni, R. Bruno, G. Mambrini, and C. Petrioli, "Comparative Performance Evaluation of Scatternet Formation Protocols for Networks of Bluetooth Devices," *Wireless Networks*, vol. 10, no. 2, pp. 197–213, March 2004.

[21] W. Wang, Y. Wang, X.-Y. Li, W.-Z. Song, and O. Frieder, "Efficient Interference-Aware TDMA Link Scheduling for Static Wireless Networks," in *MOBICOM*. ACM, September 2006.