On the Construction of a Collision-Free Schedule of Beacons in a Wireless Multi-Hop Network

Name Surname, Name Surname, Name Surname, *Member, IEEE,* Name Surname, and Name Surname, *Senior Member, IEEE*

***Abstract*—Organizing multi-hop wireless networks medium access control is a challenge due to the presence of the hidden node problem. It requires reliable coordination among neighbours and second-hop neighbours. In this paper we propose the construction of a collision-free beacon schedule that can be a first step towards the orchestration of effective medium access. The construction process involves iterative trial and error until a collision-free schedule is found. The participating nodes will keep changing the schedule until all the beacons are successfully received. At this point, the nodes simply keep repeating the successful schedule.**

***Index Terms*—Medium Access Control, decentralized constraint satisfaction solver, learning MAC protocol**

I. Introduction

MEDIUM access in multi-hop wireless mesh networks is complicated by the fact that not all the stations are in each other's transmissions range. Therefore is no longer possible to rely on Carrier Sense Multiple Access to coordinate the access to the medium. It is likely that transmissions overlap and interfere with each other. A completely random MAC protocol will not offer satisfactory performance. Some kind of coordination among neighboring nodes is needed.

In this paper, we propose the initial construction of a collision-free schedule of beacons. This process has to be decentralized as it takes place before the network is operative, which means that the participating stations can exchange data only with their neighbors. The construction of the schedule is an iterative trial-and-error process that continues until a collision-free schedule of beacons is built. At that point, it is possible for the nodes to reliable exchange information and transmit data with success guarantees.

II. The Construction of the Collision-Free Schedule

A. Assumptions

We model the network topology as a graph. In the topology graph, the nodes represent wireless stations and the arcs between two nodes mean that those nodes can communicate with each other. For illustration purposes, we will use the ring topology in Fig. 1.

If a node has two neighbours transmitting simultaneously, it is assumed that that particular node cannot decode either of the transmissions. For example, assume that at a given point of time nodes F and B in Fig. 1 are active. Node A will be unable to recover the transmissions.

Note that collisions occur at reception. The fact that a collision occurs at A when F and B simultaneously transmit does not mean that E or C cannot recover those messages, even though they overlap in time. A transmission may collide in one receiver and be successful in another one.

A node can only detect and receive the transmissions of its immediate neighbours, and only if there is a single neighbour is active for the whole duration of the transmission.

These graph model represents strong assumptions because, in reality, wireless propagation is much more complicated. The carrier sense range exceeds the transmission range and there is capture effect. There are a large number of factors that determine whether a transmission is successful or not. Nevertheless, for tractability, we use the above described graph model.

We also consider synchronization out of the scope of the present paper. Even though beacons can be used to synchronize the network, we will not cover the initial synchronization of clocks and posterior clock drifts.

B. A Beacon Schedule

A recurrent solution in wireless networks is the use of broadcast beacons for synchronization and signaling. Beacon messages are very short, which means that the likelihood of collisions is lower. This beacons can be very useful for network coordination and management.

A schedule of beacons defines special times in which beacons can be transmitted. Let us assume that beacons should be transmitted every T_b seconds. We will name each of the time instants at which a beacon can be transmitted a *spot*. Each spot can be either empty, successful or collision. Note that, again, the perception of the spot depends on the position of the station in the topology. For a given slot, it may happen that one station detects it as empty, another station as successful and another station as collision.

Then, it is also necessary to define a schedule length S which is the number of spots. Each station transmits a beacon once in every schedule in one of the S allowed spots. When the schedule is over, it is repeated again. The length of the schedule in time units is simply $T_s = S * T_b$.

An example of a beacon schedule is represented in Fig. 3. The axis represents time and each of the shaded blocks represents an spot in which beacon transmissions occur. The dotted box is a spot in which no transmission occur.

The authors are with Universitat Pompeu Fabra. Roc Boronat 138, 08018 Barcelona, Catalunya, Spain. E-mail: jaume.barcelo@upf.edu This paper has been submitted to an IEEE journal.

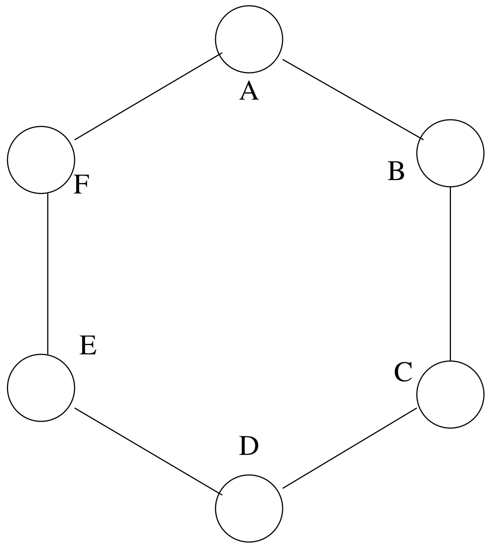

Fig. 1. Topology

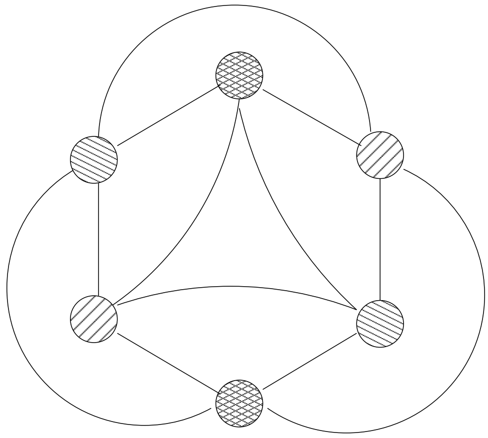

Fig. 2. Interference

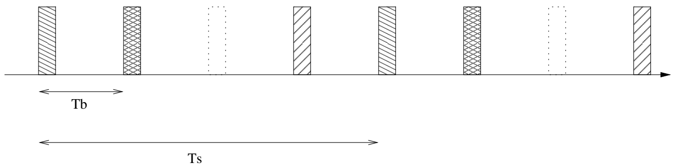

Fig. 3. Timeline

Note that the starting and ending position of the schedule is only of local significance. We take the convention that each node considers the initial position of the schedule the one in which it transmits, and is numbered as zero.

A beacon includes acknowledgement information for all the beacons received by the station in the last schedule. When a station A receives beacons from all its neighbors confirming that they have received the beacon of station A, this station can assume that its beacon has not suffered any collision.

The challenge is to construct a schedule in such a way that all the neighbors of a station can listen to its beacon. When such schedule is successfully constructed, all the nodes will receive confirmation of successful reception of its beacon from all neighboring stations.

C. The Interference Graph

From the topology graph we derive the interference graph. In the interference graph, there is an arc between two nodes the simultaneous transmission of those two nodes results in interference. Given the interference graph, the problem finding the collision-free schedule of beacons reduces to a map coloring problem.

We are interested in solving this problem in a decentralized way, as there is no central entity that has the global vision of the network and can magically configure all the nodes. Only *after* the problem has been solved, the network will work and it would be possible for a central point to gather all the information and remotely configure the nodes.

In order to solve this problem in a decentralized way, we need that the nodes keep trying different spots (or colours) until a solution is found. Otherwise it will change the color.

D. The construction of a beacon schedule

Each station picks a random spot in the schedule. If it receives confirmation from all of its neighbors, it assumes that no collisions have occurred and keeps transmitting at that position. This means that no other neighbor in the two-hop neighborhood has picked the same color. In this case, the station will defer the transmission of the next beacon a deterministic number of spots equal to the schedule length S.

If a node in the two-node neighborhood pick the same color, the node that is a common neighbor of both will not receive the beacon and therefore will not confirm the correct reception. In this case, the station will defer the transmission of the next beacon a random number of spots. A sensible choice is to use an uniform variable between 1 and $2S$ so that the expectation of the random variable roughly coincides with the deterministic backoff used by successful stations.

E. Distributed Adjustment of the Schedule Length

The beacon interval T_b is the same for all the participating stations in the network and is set by the standard or network administrator. In static networks, it is also possible for the administrator to set the schedule length S. However, in more dynamic environments, the administrator might only be able to set an initial value that needs to be continually adjusted as the topology changes.

Note that there is no such thing as an optimal schedule length. Each choice of a schedule length represents a point in the trade-off between convergence speed and performance in the stationary state. Shorter schedules result in longer convergence time and higher efficiency when convergence is reached. The actual performance will depend node density, dynamics of the network and the frequency of channel errors.

Here we present an heuristic oriented to provide reasonable performance by preventing two extreme cases. One of them occurs when in a particular neighbourhood of the network there are more stations than spots in the schedule and a collision-free schedule cannot be reached. The other extreme case occurs when the spots that actually contain a beacon are a small minority, and therefore the gains of the proposed approach are also minimal.

Each node collects information of its perception of the schedule occupation O, which is the ratio between occupied positions and total positions. A position with collision is assumed to be occupied. To this extend it is necessary that the nodes can use energy detection to determine that a position is occupied.

The nodes share this information with their neighbours and then adjusts its own schedule according to the following rules.

- The schedule is doubled if any of the neighbours report occupation equal to 100%.
- The schedule is doubled if all the neighbors report occupation higher than 75%.
- The schedule is halved if all the neighbors report occupation lower than 25%.

F. Data Included in the Beacons

In order to construct the collision-free schedule of beacons and facilitate subsequent data transmission, it is necessary to include some signaling information. For each spot of the schedule, the stations reports two different values: the beacon that has been received in that spot (if any) and the number of packets ready to send to that particular destination.

For example purposes we imagine that A in Fig. 1 in its second transmission in Fig. 3 has correctly received the beacons from its neighbours B and F, and has two outstanding packets for B in its queues. Then, the data information in the beacon includes:

- Position 1: empty beacon
- Position 2: acknowledgement for beacon B, 2 outstanding packets for B
- Position 3: acknowledgement for beacon F, 0 outstanding packets for F

Additionally, each station also includes beacons for all the received data packets since the last beacon by that station. This is a difference with respect to typical IEEE 802.11 implementation in which data acknowledgement is sent right after the correct reception. As the acknowledgement is sent by the receiver of the data, it interferes with different transmissions than the original data. These short acknowledgement packets can have devastating effects if they destroy a long data packets. Furthermore, the acknowledgement requires a short inter-frame interval to account for the round-trip time and the time required for the hardware to switch from reception to transmission. Channel time fragmentation is explained in [1] For these reasons, it is better to delay the transmission of the acks and send them all together in the beacon.

G. The Medium Access Control Protocol

Each station owns the time that follows its beacon until the next beacon spot. It can use this time to transmit packets to its neighbours and to request packets from its neighbours using *poll* packets. When the station is finished with transmitting and receiving packets, it sends a short *done* packet that allows its neighbours to contend for the medium using CSMA/ECA.

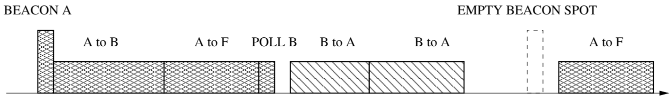

Fig. 4. Medium Access

The medium is also open for contention after *free* beacon spots. A station that detects that one of the beacon spots is not use for its neighbours can use CSMA/ECA to send packets to its neighbours.

H. An Example

An example of medium access is presented in Fig. 4. It shows the access to the medium from the point of view of station A. After transmitting its beacon, station A is free to transmit and request packets from and to its neighbours. In this particular example A transmits a packet to A and then a packet to B. Then, it uses its remaining time to request and receive two packets from A. The two packets are requested in the poll message.

The next beacon spot is free from A's point of view. Therefore A can contend for the medium using CSMA/CA and in this transmission it is exposed to the hidden terminal interference.

Even though it is not shown in the figure, at a later time B transmits its beacon. Then A has to remain silent unless B explicitly requests a packet from A using a poll.

III. Related Work

A first attempt of a collision-free MAC protocol for multi-hop wireless networks appeared in [1]. That solution does not use beacons, and therefore the convergence is slower and the efficiency is lower.

An initial step in the direction we are exploring in this later was taken in [2]. In that paper, the construction of a collision-free schedule is studied for a sensor network. That paper focused on a simple scenario in which all nodes could hear each other transmissions. Another difference between that paper and the present letter is that here we not aim to compute a schedule length which is similar or equal to the participating nodes. In the present letter the schedule is independently computed by each node and therefore it is possible that different nodes use different schedule lengths.

The distributed coloring of graphs has been studied in [3] which also considered the possibility of sensing restrictions. In the present work we do not consider such sensing restrictions. Another difference is that in the present work the beginning and ending of the schedule is different for different nodes, and even the schedule length can be different for different nodes.

A. IEEE 802.11s Mesh Beacon Collision Avoidance

A Mesh Beacon Collision Avoidance (MBCA) is described in [4]. The solution consists of sometimes delaying the beacon transmission to check for possible overlapping transmissions among the one-hop neighbours. It also involves including information about the timing of beacons in beacons.

This approach has some shortcomings. It does not prevent that two-hop neighbors choose overlapping beacon transmitting instants. The timing information in beacons is available only in case of successful transmission and reception. In the critical case in which collisions occurs and timing information is needed, it is not available. Delaying the beacon sometimes seems also problematic. It is not impossible that the two colliding beacons choose to delay simultaneously and therefore the collision is not detected. If the beacons are not delayed very frequently, it takes a long time to detect the collision. If the beacons are delayed frequently, additional randomness is introduced that can easily result in more collisions.

IV. Discussion

The proposed protocol has some shortcoming such as requiring periodical beacons that fragment and consume channel time, the possibility of unfairness if different nodes choose different schedule length, and the requirement for the stations to hold the packets longer in their queues due to the delayed acknowledgements. Nevertheless, the protocol also offers interesting aspects. In particular, each station owns a fraction of the channel time in which it can transmit without contention and without hidden problems. The reserved channel time that is not consumed is freed for so that other stations can use it.

Acknowledgment

The authors would like to thank ...

References

[1] J. Barcelo, B. Bellalta, C. Cano, A. Faridi, and M. Oliver, “On the distributed construction of a collision-free schedule in multi-hop packet radio networks,” *Telecommunication Systems*, pp. 1–14, 2013. [Online]. Available: http://dx.doi.org/10.1007/s11235-013-9836-5

[2] C. Cano, D. Malone, B. Bellalta, and J. Barcelo, “On the Improvement of Receiver-Initiated MAC Protocols for WSNs by Applying Scheduling.”

[3] A. Checco and D. Leith, “Learning-Based Constraint Satisfaction With Sensing Restrictions,” *Selected Topics in Signal Processing, IEEE Journal of*, vol. 7, no. 5, pp. 811–820, 2013.

[4] G. R. Hiertz, S. Max, R. Zhao, D. Denteneer, and L. Berlemann, “Principles of ieee 802.11 s,” in *Computer Communications and Networks, 2007. ICCCN 2007. Proceedings of 16th International Conference on*. IEEE, 2007, pp. 1002–1007.