Two Phase Dynamic Channel Assignment Protocol on Multi-Channel Multi-Interface Wireless Ad Hoc Networks

Abdul Goni, Goutam Gorai & Santanu Rakshit

National Institute of Technology, Durgapur
E-mail : abdulgoni2009@hotmail.com, goutam.gorai@gmail.com, santanurksht@gmail.com

***Abstract* – Using multiple channel over single channel the performance can be significantly increased since the use of multiple channels can reduce interference influence. Channel assignment is one of the most basic and major concern in such networks. By effectively utilizing multiple non overlapping channels and multiple interfaces, collision and co-channel interference can be reduced. This allows more concurrent transmissions and thus enhances the network capacity. In this technique an interface is allowed to dynamically change its working mode between send and receive on a call-by-call basis, which enhances the utilization of both interface and channel. Unlike existing techniques , in this paper control channel has been taken into account only for transmitting the control information not for transmitting data in any situation .Two channels are grouped into a single channel and assign both the channels together at a time in a particular link for both sending and receiving purpose. The two phases of channel assignment are – (i) Pseudo channel assignment and (iii) Actual channel assignment. A heuristic approach will be used to choose the best path. The best path will be decided using the following parameter – (a) Cost of the path [energy saving] and (b) Interference range [must be less than the threshold value]. After selection of the best path among all the possible paths, the actual channel assignment will be done. With a comparable complexity as the existing schemes, this technique provides much higher system good puts and shorter end-to-end packet delays.**

Keywords – BESTPATH, Control Channel, Control Interface, Data channel, Data interface, PATHFIND.

I. INTRODUCTION

In this paper the multiple numbers of channels and multiple numbers of interfaces are used to improve the performance of the wireless mesh network. A wireless mesh network [3] is a multi-hop wireless network consisting of a large number of wireless nodes, some of which are called gateway nodes and connected with a network. In case of traditional *single channel single interface* mobile ad hoc networks (MANETs), network capacity decreases with the increasing number of nodes, due to collision and interference in the single shared medium. By effectively utilizing multiple non-overlapping channels (12 in IEEE 802.11a), the network capacity can be significantly enhanced by allowing more concurrent transmissions. In a single channel wireless network, two transmissions in a neighborhood are not allowed to happen at the same time because of contention for the shared wireless channel. However, in a multi-channel network [6], no collision will be caused by such simultaneous transmissions as long as they work on different channels. In this paper the channel assignment has done in two phases. In first phase assign pseudo channel in the all probable link. After assigning channels to all links (i.e. from sender to receiver) the routing has done to choose the best path. Among the n number of paths from sender to receive the algorithm first choose the m number of lowest cost paths before the timer gets expired. After that the algorithm choose that path which has lowest secondary interference and path cost accumulatively. After getting the final path the second phase i.e the actual channel assignment has done. In a dynamic mobile ad hoc network (MANET) [5], the network topology is usually not known in advance (due to nodal mobility) and the traffic demands are even harder to predict. Centralized algorithms for joint channel assignment and routing cannot be applied. Many distributed algorithms [4] are proposed to optimize the network performance in real-time and on a call-by-call basis. Unlike [4] in this project the control channel is used only for transfer control message. Here all links are assigned with pair of channel, that look like a single channel, one of which is for sending data and another is for receiving data. We show that with a comparable complexity as the existing schemes, this technique provides much higher system outputs and shorter end-to-end packet delays.

The remainder of the paper is organized as follows. In section II the objective, in Section III Related works, in section IV Interface Assignment, in section V Channel Interference Index, in section VI Bi-Directional Path Setup, in section VII Algorithms, in section VIII handling channel conflicts and in section IX Conclusion are discussed.

II. OBJECTIVE

In this paper the problem of receiving channel assignment of J-CAR [4] is resolved by the assigning of both the sending and receiving channels at the same time. So the data transfer using the control channel is fully restricted in this technique. Here control channel is responsible only for the travelling of control information. To overcome the problem of receiving channel, in this paper the pair of channel is used to assign channel for sending and receiving. Pair of channel is treated as one single channel (create a group) and both channels are assigned together all the time. In this paper the time for constructing the path is also minimized.

III. RELATED WORKS

The single interface and fixed channel assignment protocol is very simple to implement but the capacity of the network is limited. In case of single interface the throughput of nodes are not good because the node can only either send or receive data from other node. In case of shortest path based topology [1] the secondary interference can harm the data transfer. Fixed channel assignment [8] is easier than the dynamic channel assignment but in case of dynamic channel assignment the throughput is better than the static channel assignment.

In [4] they have used the joint channel assignment and routing protocol. In this technique bidirectional path set up has to be done. The propose and approve based path setup is done in this technique. As in [2] a receive mode interface is not switchable and if receiving node cannot support the proposed channel, then it falls back to the control channel. The control channel can be used as the data channel to transfer data. This may degrade the performance significantly if the size of the network is large. If control channel is busy then if the data channels are available though the nodes cannot make new connection.

IV. INTERFACE ASSIGNMENT

In this paper we consider multiple numbers of interfaces, one of which is only for control information, called control channel. It is used only for carrying control information (broadcast packets). The remaining interfaces are *data interfaces*, for carrying data packets on different data channels. Since RTS-CTS handshaking in the control channel [11] has been shown [3] to be inefficient, J-CAR performs the four-way 802.11 handshaking of RTS-CTS-DATA-ACK in the selected data channels. When a call arrives and a route is to be set up, a data interface becomes active (where a data channel is assigned to it) and switches to either *send mode* or *receive mode*, depending on the direction of the traffic flow. To save energy, an inactive data interface can be turned off, or put in *sleep mode*. Send mode and receive mode differ as follows. In send mode, an interface can switch to different data channels to send data packets to different neighbors (so effectively it works on multiple channels), but it is prohibited from receiving data packets. In receive mode, an interface can only work on a *single* data channel, mainly for data receiving.

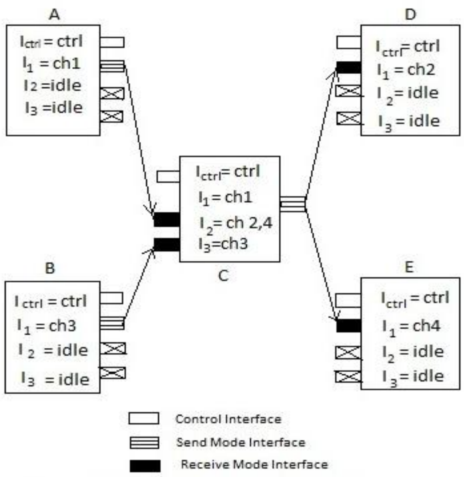

Fig. 1 : Use of Multiple interfaces for Multiple Channel Assignments

An Example

The potential gain of using switchable working modes can be illustrated by Fig. 1. The network consists of five nodes, each has four interfaces labeled as I_{ctrl}, I_1, I_2, and I_3, where I_{ctrl} is the control interface listening on the control channel, I_1, I_2, and I_3 are data interfaces. There are two paths, A-C-D and B-C-E, intersect at node C. Consider the path AC- D. With J-CAR, data channel 1 (ch1) is selected by link A-C. The two nodes communicate by having I1 at node A in send mode and I1 at node C in receive mode. Similarly, ch2 is selected by link C-D, occupying I_2 at C (send mode) and I_1 at D (receive mode). We can see that the two links use different channels. This eliminates mutual interference. Consider the second path B-C-E. With J-CAR, ch3 is selected by link B-C, occupying I_1 at B (send mode) and

I_3 at C (receive mode). In order not to interfere with the data receiving at I_1 and I_3, node C selects I_2 and ch4 for link C-E. Since I_2 at node C is in send mode, it can switch between ch2 and ch4 for data sending. Finally, node E switches its I_1 to ch4 (receive mode). It should be noted that node C receives data using two interfaces, the throughput performance is better than the receiving channel pre-assignment schemes [12], [13], due to the reduced packet collision probability.

V. CHANNEL INETRFERENCE INDEX

At each hop along a candidate path for a new call, this technique selects the best channel to use based on the *channel interference index*, which is a heuristic measure of the "goodness" of a channel as perceived by the node under consideration. We define the interference index of channel *i* as perceived by the node under consideration as:

$$\text{Index}_i = \sum_{j=1}^{k} \frac{\text{usage}(i,j)}{j^{y}}$$

where *k* is the interference range, y is the path loss exponent , and *usage*(*i, j*) is the *channel usage table*. *usage*(*i, j*) is maintained at each node for keeping track the amount of data sent by nodes using channel *i* at the *j*-th hop, where *i€* [0, *num_channel*-1] and *j€* [0, *k*+1]. When *j*=0, *usage*(*i,* 0) is the channel usage of the node under consideration. Each node estimates the amount of data it sent (send load) using (1), and exchange its send load with neighbors (within *k*+1 hops) via the periodic broadcast HELLO packets. Note that due to mobility and possible packet collision, HELLO packets may be lost or carry obsolete information. But this would have little impact to the overall system performance because the channel selection is based on *Index*, where the contribution by lost/obsolete HELLO packets will be minor.

VI. BI-DIRECTIONAL PATH SETUP

In this paper the channels are assigned in a group of two channels for sending and receiving purpose. So the channel assignment is easier in this technique because both the sending and receiving channels are assigned in forward path. Both the sending and receiving channels will be used as sub channel.

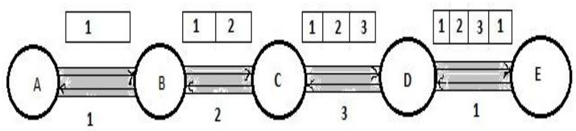

Fig. 2 : A propose and approve process for bi-directional setup (k=3)

In RREQ message there will be a list that holds the proposed channel number. The receiver will grant (pseudo channel assignment) it only if the proposed channel is free and non interfering with other channel. The RREQ message is broadcasted by every receiver. After reaching the message to the final node through several paths, m (m<n) number of best path will be chosen according to the path cost. Then the interference of m paths will be checked one by one. The lowest cost path will be checked first and so on. If the interference is below the threshold level and summation of path cost and interference is minimum then that path will be chosen finally. The route reply will confirm the channel assignment (actual channel assignment) of whole path.

VII. ALGORITHMS

Algorithm for Finding Paths:

We can represent any wireless mesh network as a graph. And we can represent a graph in computer using the matrix representation.

PATHFIND {path[ ], init, sender, receiver, destination, PC_{total}, ch }

PC_{total} =total path cost .

Initialize: path[ ]= null , sender=initiator, PC_{total}=0, ch=available channel.

Step1: Broadcast the RREQ packet P from initiator node to the adjacent .

Step2: After receiving the packet , the receiver will do the following two job:

Step2.1: The receiver will check the proposed channel whether it is available for that link or not from the channel interference index. If the proposed channel is acceptable for the receiver then the receiver will accept the packet and do the next step, but if the channel is not acceptable then the receiver will send a request to the sender with a proposed channel. This process will continue until the channel is granted by both the node.

Step2.2: The receiver will update the packet content as follows:

path[ ] =path[ ] + link{sender > receiver}

$PC_{total} = PC_{total}$ + cost (link{sender > recVr})

sender = receiver

Step3: Now all the receiver will send the updated packet to the adjacent node except higher level node as the new sender node.

Step4: Repeat Step2 and Step3 until receiver = destination

Step5: Destination node will compare PC_{total} factor for all packet. Using any good sorting algorithm this node will sort all the packets (i.e. all the paths) in ascending order according to the path cost. An array SP[ ] store sorted paths.

Step6: Return SP[ ] to the Algorithm BESTPATH{} for choosing best path among all the paths.

Step7: End.

Algorithm for Best Path Finding And Actual Channel Assignment:

BESTPATH { SP[ ] }

Step 1: While ($Interference_i$ > Threshold)

[Interference of the i th path will be the sum of all the channel's interference assigned to that path .The interference index of all the channels will be available in the Interference Index Table .]

$Interference_i = \sum$ All the channel's interference index

Step 2: FinalPath = $path_i$

When the Interference of any path is less than the threshold value, that means the interference is tolerable then no more checking is required. This path is the least cost path with minimum interference.

Step 3: Send a reply message from the destination node to the initiator node. This message will confirm the channel assignment. This is actual channel assignment.

Step 4: End.

VIII. HANDLING CHANNEL CONFLICT

During the propose and approve channel setup process the receiver may reject the channel proposed by sender. Then the receiver will chose more suitable channel and request to the sender. If it is available to the sender then accept the channel and send confirmation otherwise it propose new channel again and continue the process until the channel is approved by the both nodes.

The channel conflict scenario may occur due to the lack of available interface. In that case the request should wait until the interface became free or the interface can switch from one channel to other in round robin fashion.

IX. CONCLUSION

In this paper we have used IEEE 802.11 a multi-channel and multi-interface to gear up the system performance. Unlike the J-CAR [4] in this paper we used the control channel just only for control information. The control channel will never used for sending or receiving data. This will enhance the performance when the network is more crowded. In case of J-CAR [4] if the control channel is used by one path and at same time if another node requires the control channel to send the control information, then that node has to be wait until the control channel is released. This may reduce the performance of the system. Here the control channel is always ready only for travelling of control information. As we cannot use the control channel to carry data then the problem of receiving channel assignment [4] is removed by assigning both the sending and receiving channel at the same time. This can be achieved by grouping the two channels in a single channel. So the assigning of channels is easier here. Finding of all the paths is not necessary for this technique. A timer T is used in RREQ packet. When the timer is expired then the packet will be dropped by that node. It avoids the infinite waiting for path generating. Before the timer expiration which packet will reach to the destination, those packets will be considered for paths only. As all paths are sorted according to the path cost, the algorithm needs to check only some number of paths from the SP[ ] array until it get the interference lower than the threshold value. It reduces the required time to set up the path for transferring data and make the system faster.

X. REFERENCES

[1] Yao Shen, yunze Cai, Xiaoming Xu. A Shortest-Path-Based Topology Control Algorithm in Wireless Multihop Networks, in: ACM SIGCOMM Computer Communication Review, October 2007.

[2] Lichun Bao, J.J Garcia-Luna-Aceves. Stable Energy-aware Topology Management in Ad hoc Networks, in: ElseverB.V, September 2009.

[3] Li Li, Bin Qin, Chunyuan Zhang. Interface-aware Broadcast Routing and Channel Assignment in Multi-Radio Wireless Mesh Network, in: IEEE , 2009.

[4] Hon Sun Chiu, Kwan L. Yeung and King-Shan Lui. J-CAR: An Efficient Joint Channel Assignment and Routing Protocol for IEEE 802.11-Based Multi-Channel Multi-Interface Mobile Ad Hoc Networks, in: IEEE Xplore, December 16, 2009.

[5] Rouzi Sun, Jian Yuan, Ilsun You, Xiuming Shan, Yong Ren. Energy-aware weighted graph based dynamic topology control algorithm, in: Elsever B.V , September, 2010.

[6] Weiyi Zhang, Farah Kandah, Jian Tang and Kendall Nygard. Interface-Aware Robust Topology Design in Multi-Channel Wireless Mesh Nwtworks, in: IEEE , 2010.

[7] Wen-Hwa Liao, Sital Prasad Kedia, Avinash Kumar Dubey. Scheduling and Channel Assignment Algorithm for IEEE 802.16 Mesh Networks Using Clique Partitioning Technique, in:Elsever, May,2012.

[8] Arindam K. Das, Hamed M. K. Alazemi, Rajiv Vijayakumar, Sumit Roy**.** Optimization Models for Fixed Channel Assignment in Wireless Mesh Networks with Multiple Radios.

◈◈◈