HTSMA: a Hybrid Temporal-Spatial Multi-Channel Assignment Scheme in Heterogeneous Wireless Mesh Networks

Yan Jin[1], Ju-Yeon Jo[2], Mei Yang[1], Yoohwan Kim[3], Yingtao Jiang[1], John Gowens[4]
[1]Department of Electrical and Computer Engineering, [2]School of Informatics, [3]School of Computer Science
University of Nevada, Las Vegas, Las Vegas, NV 89154
[4]Army Research Laboratory, Adelphi, MD

***Abstract*—A number of multi-channel assignment schemes have recently been proposed to improve the throughput of IEEE 802.11-based multi-hop wireless mesh networks (WMNs). In these schemes, channel coordination is done either through time synchronization across all the hosts, or through the use of a dedicated channel for the transmission of necessary control messages. Either way, excessive system overhead and/or waste of bandwidth resource become unavoidable, undermining the overall network throughput. To maximize the network throughput, we propose a synchronization-free, hybrid temporal-spatial multi-channel assignment scheme in a random heterogeneous network requiring only a single radio interface per host. In this scheme, the gateway is allowed to use all the available channels sequentially in a round-robin fashion. This temporal channel assignment approach ensures that all the neighboring hosts that communicate with the gateway directly shall have a fair access to the gateway. The channel assignment for the remaining wireless hosts is based on the geographical location and channel availability (a spatial approach) to avoid the interference within the communication region of each sender host in its transmission time period. Compared with another multi-channel scheme MMAC, extensive simulation results demonstrate that our proposed scheme can improve the network throughput substantially with the acceptable collision ratio.**

I. INTRODUCTION

To meet the increasing demand for better and seamless network services, wireless mesh networks (WMNs) have emerged and attracted more attention [1]. Due to the limited radio transmission range, each mesh host has to utilize other mesh hosts to communicate with the gateway in distance through multi-hop communications. As a result, the network performance of multi-hop WMNs tends to degrade sharply when the number of communication hops traversed increases, mainly due to the increased channel contention/collision rate and transmission latency. As such, to preserve a reasonable network throughput, it is necessary to ensure the packets are routed through the routing paths that have no or less interference. In conventional IEEE 802.11 standard (single-channel), the bandwidth available for an end-to-end connection decreases by $\theta(B_w / \sqrt{n})$ bits/s, where B_w is the available bandwidth and n is the number of hosts [2]. Fortunately, the IEEE 802.11b/802.11g standards and IEEE 802.11a standard provide 3 and 12 orthogonal (non-overlapping) channels respectively, which could be used simultaneously within a neighborhood. Intuitively, one can see that the ability to intelligently utilize multiple channels within the region increases the effective bandwidth available to a WMN substantially, leading to great improvement of the network throughput.

Wu *et al.* proposed a dynamic channel assignment (DCA) scheme to maintain a dedicated channel for control messages and other channels for data packets [3]. Each host has two radio interfaces, so that it can listen on the control channel and the data channel simultaneously. However, this scheme wastes the valuable system resources, especially when the number of available channels is limited. On the other hand, if the number of channels is large, the control channel can become a bottleneck for performance and prevent data channels from being fully utilized. Raniwala *et al.* [4] and Avallone *et al.* [5] addressed the joint multi-channel assignment and routing problem, a known NP-complete problem reduced from the Multiple Subset Sum Problem [6], and proposed different centralized approximate solutions to solve it. A distributed heuristic scheme that adapts to the dynamic traffic loads was proposed in [7]. Actually, having multiple radio interfaces can be too expensive for small and low-cost devices. In addition, multiple interfaces can also cause unpredictable inter-interface interference. Motivated by this observation, So *et al.* proposed a multi-channel MAC (MMAC) protocol, which requires only one interface per host [8]. In this protocol, the beacon interval is divided into cycles composed of two phases: control phase and data exchange phase. At the beginning of each cycle, the control phase is initialized and all hosts listen to a predefined, common channel to negotiate a channel to be used during the data exchanged phase. When the control phase ends, the data exchange phase starts. The disadvantage of this protocol is that it needs global time synchronization so that all hosts begin their beacon interval at the same time. Consequently, it increases the system overhead and is infeasible to be applied to a WMN with a large number of hosts.

To address the aforementioned challenges, we propose a Hybrid Temporal-Spatial Multi-channel Assignment (HTSMA) scheme in the heterogeneous WMNs where neither a dedicated channel nor time synchronization is needed. It uses multiple orthogonal channels among the hosts equipped with single radio interface to improve the network throughput. HTSMA has the following characteristics:

- Each sender host does not need to find a complete routing path toward the gateway before the transmission. Comparably, it only needs to find the next hop host for its transmission, which decreases the algorithm complexity;
- The gateway utilizes available channels fairly by switching channels sequentially in a round-robin fashion to collect the packets from its neighbors, which is considered the *temporal* property of HTSMA. Other hosts utilize different orthogonal channels simultaneously within their

neighborhoods, which is considered the *spatial* property of HTSMA.

II. PROBLEM FORMULATION

A. Notations, assumptions and network topology

Table I lists the notations used in this paper.

TABLE I. NOTATIONS.

Symbol	Description
S	Set of all hosts in the network except the gateway
C_N	Number of channels
B_w	Bandwidth of each channel (bps, bits per second)
r_u	Radio range of host u
B	Buffer capacity of each host in S (pks, number of data packets)
$B_a(u)$	Available buffer capacity of host u ($B_a(u) \leq B$, $u \in S$)
$H(u)$	Hop count of host u
$S_N(u)$	Neighbor set of host u in a undirected bilateral graph

Assumptions:

(1) C_N orthogonal channels (denoted as channel 0, 1,…, C_N-1) are available for use, and all channels have the same bandwidth B_w;

(2) Each host is equipped with a single tunable half-duplex transceiver. That is, a host can listen to or transmit on only one channel at a time, but it can switch to different channels over time. The channel switching occurs instantaneously;

(3) The distribution of the hosts is random and each host in S has the limited buffer capacity B. Each host may function as a source host which generates data packets;

(4) All hosts maintain their own clocks independently.

As most of the traffic in a WMN is directed to the wired network, each host needs to discover a path to one or multiple wired gateway hosts. In this paper, we focus on the case where all hosts over the network are actively associated with only one gateway.

Definition 1. A network can be modeled as an undirected, bilateral communication graph $G = (S \cup \{GW\}, E)$. Here GW denotes the gateway. Given a host u and a host v, there is a edge (u, v) in E if and only if $r_u \geq dis(u, v)$ and $r_v \geq dis(u, v)$ where $dis(u, v)$ is the Euclidean distance between host u and host v. In this case, $v \in S_N(u)$ if and only if $u \in S_N(v)$.

Definition 2. The hop count of host u is h if the least number of hops traversed by a packet between u and gateway in an undirected, bilateral communication graph G is h ($h \geq 1$). The gateway is called the 0-hop host.

Such an undirected, bilateral communication graph G is formed using the distributed algorithm in Fig. 1. A back-off system timer is used in Phase II so that each host can receive more hello messages to determine its hop count. After Phase II, each host obtains its hop count and maintains a neighbor set.

B. Problem description

Given: C_N orthogonal channels and an undirected, bilateral communication graph G that consists of a gateway and a number of hosts.

Objective: Assign C_N channels to a number of sender hosts (including the source hosts and the forwarding hosts on the routing paths from the source hosts to the gateway) to maximize the network throughput.

Constraints: The proposed scheme should not require global time synchronization and there is less interference within the communication region of each sender host in its transmission time period.

This problem is inherently NP-complete. In next section, we will propose a distributed heuristic, the HTSMA scheme, to solve this problem.

```
Communication Graph Formation Algorithm
// Phase I: Initialization
Switch to a predefined, common channel for each u ∈ S ∪ {GW};
Set the hop count of gateway H(GW)←0;
Set H(u)←∞ for each u ∈ S;
Set S_N(u)←Φ and bInitialized←0 for each u ∈ S ∪ {GW};
Gateway broadcasts a hello message with its radio range r_gw and H(GW);
// Phase II: Main_Processing (u: host id)    //  u ∈ S ∪ {GW}
While (bInitialized is 1 and current clock time<system timer) or S_N(u) is Φ
{   If  (host u receives a hello message from host v)
        Estimate dis(u, v) based on the receiving signal intensity;
        If  r_u ≥ dis(u, v)
            Set S_N(u) ← S_N(u) ∪ {v};
            If  |S_N(u)| = = 1
                Set bInitialized ← 1;
                Initialize a back-off system timer;
        Else  ignore this hello message;                }
H(u) ← min { min { H(w)| ∀ w ∈ S_N(u) } + 1, H(u)};
Broadcast a hello message, piggybacking its radio range r_u and H(u);
```

Figure 1. Pseudo-code of network topology formation algorithm.

III. HTSMA: A HYBRID TEMPORAL-SPATIAL MULTI-CHANNEL ASSIGNMENT SCHEME

The HTSMA scheme consists of two parts, the channel assignment scheme for a h-hop host (h>1), and the channel assignment scheme for the gateway and 1-hop host, which is described in Section III.A and Section III.B, respectively.

A. h-hop host (h>1): a spatial channel assignment scheme

Each host in S maintains a channel utilization table (CUT). Each entry in this table includes three fields: the numbered channel occupied (NCO) by a neighbor, the communication beginning time (CBT) and the announced duration period (ADP) on this channel. The ith row in a CUT records the utilization information of channel i. Initially, there is no packet traffic in the network, thus the CUT of each host is empty. Before we present the detailed algorithm for any h-hop host u, the control message types and the timer types used in this algorithm are listed in Table II. Without loss of generality, at some moment t, the algorithm performed at any h-hop host u ($h \geq 1$) is presented in Fig. 2.

TABLE II. MESSAGE TYPE AND TIMER TYPE.

Name	Description
RTT	Request-To-Transmit message transmitted by a sender host
RTR	Request-To-Reply message transmitted by a receiver host
CRE	Channel-Reserve message transmitted by a sender host
T_{neg}	System timer triggered by a sender host
$T(u)$	Timer triggered by a potential receiver host u

As shown in lines 2-4, any 1-hop sender host merely needs to stay on its current channel, waiting for the opportunity to communicate with the gateway directly. The detailed algorithm of 1-hop hosts is described in subsection III.B. For a h-hop sender host (h>1), if all channels are occupied by its neighbors, then it will have to defer its transmission until one of its neighbors releases the channel (as shown in lines 5-8).

Otherwise (lines 9-12), it will broadcast a RTT message on the lowest-numbered free channel and then initialize a system timer T_{neg}. The value of T_{neg} can be calculated by adding the maximal back-off time period to the time required to transmit control messages and the time needed for interframe spacings. On the other hand, as shown in lines 20-27, if a h-hop host u receives a RTT message from one of neighbors, say host m, host u will defer broadcasting its RTR message after a back-off time $T(u)$ expires ($0 \leq T(u) \leq T_{neg}$). The value of $T(u)$ is calculated by:

$$T(u)=T_{neg}[\alpha(1-B_a(u)/B)+\beta(1-N_{ac}(u)/C_N)+(1-\alpha-\beta)H(u)/H_{max}] \quad (1)$$

where α and β are the adjustable parameters ($\alpha \geq 0$, $\beta \geq 0$, $0 \leq \alpha+\beta \leq 1$), $N_{ac}(u)$ is the number of current available channels for host u, and H_{max} is the maximal hop count of all host over the network. Based on Eq. (1), the host with the larger available buffer space, the larger number of available channels and a smaller hop count has a higher probability to be a receiver.

```
HTSMA (u: host id)        // u ∈ S
1.  Set N_ac(u) ← the number of current available channels of host u;
2.  If B_a(u)<B           // host u has packets to transmit
3.     If H(u) = = 1   // host u is a 1-hop host
4.            Stay on its current channel;
5.     ElseIf N_ac(u) = = 0    // host u is a h-hop host (h>1)
6.            Set  i ← arg min {CUT[x].CBT+CUT[x].ADP};
7.            Delay (CUT[i].CBT+CUT[i].ADP-current clock time);
8.            HTSMA (u);
9.     Else   Set  j ← the lowest-numbered free channel;
10.           Switch to channel j;
11.           Broadcast a RTT message on channel j, piggybacking
              the value B_a(u);
12.           Initialize  a timer T_neg;
13.           While (current clock time t < T_neg)
14.           {  If  (host u receives a RTR message from a host v)
15.                 Cancel the timer T_neg;
16.                 Broadcast a CRE message on all channels,
                    piggybacking its current channel number and the
                    announced duration period min{B-B_a(u), B_a(v)}/B_w;
17.                 Begin data transmission;          }
18.           Mark its current channel as unavailable;
19.           HTSMA (u);
20. If  (a idle host u receives a RTT message from a host m and m ∈ S_N(u))
21.    Set T(u)←T_neg[α(1-B_a(u)/B)+ β(1-N_ac(u)/C_N) + (1-α-β)H(u)/H_max];
22.    Initialize a timer T(u);
23.    While (current clock timer t < T(u))
24.    {  If  (host u receives a RTR message or a CRE message)
25.          Cancel the timer T(u);
26.          Insert the channel occupation information into its CUT;   }
27.    Broadcast a RTR message on all channels, piggybacking its ID,
          the occupied channel number and the announced duration
          period min{B-B_a(m), B_a(u)}/B_w;
28. If  (a idle host u receives a RTR message or a CRE message from a
       host p and p ∈ S_N(u))
29.    Insert the channel occupation information into its CUT;
```

Figure 2. Pseudo-code of HTSMA scheme for a h-hop host u ($h \geq 1$).

Note that when the sender host has found the receiver host on a certain channel j, it should ensure that all neighbors of this sender-receiver host pair become aware of transmission (as shown in lines 28-29), thus avoiding the multi-channel hidden terminal problem as well as the deafness problem [9]. For this purpose, the sender host and the receiver host needs to perform “active scanning” to broadcast the control message CRE and RTR on all channels respectively, which is shown in line 16 and 27. Finally, if a sender host has not received RTR message after T_{neg} expires, it derives that none of its neighbors is idle or stays on the channel that it has switched to. As a result, the sender host marks its current channel as unavailable and executes this algorithm iteratively, which is shown in lines 18-19.

B. Gateway: a temporal channel assignment scheme

In the multi-channel scenario, to utilize each channel and receive the packets from each 1-hop host, the gateway should switch the working channels following the order of channel 0, channel 1…. For fairness purpose, the original working interval of each channel is set to be equal, denoted as T_c. Without loss of generality, at a certain moment t, the detailed algorithm performed at the gateway (or a 1-hop host) is presented in Fig. 3, which utilizes the control messages and timers described in Table III.

TABLE III. MESSAGE TYPE AND TIMER TYPE.

Name	Description
RTRV	Request-To-Receive message transmitted by the gateway
RESV	Reserve message transmitted by the gateway
RTSW	Request-To-Switch message transmitted by a 1-hop host
STOP	Stop message transmitted by the gateway
T_c	Expected working interval of each channel
T_{neg}	System timer triggered by the gateway
$T(u)$	Timer triggered by a 1-hop host u

```
HTSMA (u: host id, j: current channel number)
If  H(u) = = 0        // host u is a gateway
   Broadcast a RTRV message on channel j;
   Initialize a timer T_neg;
   While  (current clock time t < T_neg)
   {  If   (host u receives a reply message)
         Cancel the timer T_neg;
         Broadcast a Reserve (RESV) message;
         Initialize a timer T_c;
         While  (current clock time t < T_c)
         {  If   (host u receives a RTSW from host v)
               Cancel the timer T_c;
               Set k←(j+1)mod C_N;
               Switch to channel k reactively;
               HTSMA (u, k);                          }
         Broadcast a Stop (STOP) message;
         Set k←(j+1)mod C_N;
         Switch to channel k proactively;
         HTSMA (u, k);         }
   Set k←(j+1)mod C_N;
   Switch to channel k proactively;
   HTSMA (u, k);
If  H(u) = = 1        // host u is a 1-hop host
   If  (host u receives a RTRV message)
      If  B_a(u)<B       // host u has data packets to transmit
         Set   N_t(u)←the number of transmissions between host u and
               gateway before current moment;
         Set   T(u)←T_neg [ δ B_a(u)/B) + (1-δ) C_N N_t(u)/N_SC ];
         Initialize the timer T(u);
         While  (current time t < T(u))
         {  If   (host u receives a RESV message)
               Cancel the timer T(u);    Exit;    }
         Mark itself as a sender host;
         Broadcast a reply message;
   If  (host u receives a RESV message and host u is a sender host)
      Transmit its data packets to the gateway;
      Broadcast a RTSW message;
   If  (host u receives a STOP message and host u is a sender host)
      Stop data packets transmission;
```

Figure 3. Pseudo-code of HTSMA scheme for the gateway and 1-hop host.

As shown in lines 1-21, if the gateway has not received a reply message to its RTRV message after the timer T_{neg} expires, it indicates: 1) none of 1-hop hosts stays on channel j, or 2) the 1-hop host that stays on channel j has no packet to transmit. In this case, the gateway will switch to channel (j+1) mod C_N proactively to reduce the idle cycle on this channel, which increases the bandwidth utilization and thus improves the network throughput. Lines 22-32 show that for a 1-hop host, say host u, once receiving a RTRV message (from the gateway), it will defer broadcasting its reply message after a back-off time $T(u)$ ($0 \le T(u) \le T_{\text{neg}}$) where T_{neg} is defined as Eq. (1). The value of $T(u)$ is calculated by:

$$T(u)=T_{\text{neg}}[\delta B_a(u)/B)+(1-\delta)C_N N_t(u)/N_{SC}] \quad (2)$$

where δ is an adjustable parameter ($0 \le \delta \le 1$), N_{SC} is the total number that the gateway has switched and $N_t(u)$ is the number of transmissions as a sender host by the current moment. Based on Eq. (2), the 1-hop host with a less available buffer size or a smaller number of transmissions as a sender has the higher chance to transmit its packets to the gateway. Here the advantage of introducing back-off time $T(u)$ is to prevent multiple 1-hop hosts that work on the same channel from transmitting their packets to the gateway simultaneously, which leads to the receiving interference at the gateway. Note that to minimize the overhead, the value of T_c is set large compared to that of T_{neg}. In the simulations (section IV), we have $T_c=4T_{\text{neg}}$.

Fig. 4 illustrates an example. In Fig. 4(a), hosts M, N, P, Q and W are five 1-hop hosts. Each 1-hop host, say host M, is labeled with its ID and the ($B_a(M)$, $N_t(M)$) pair. Given C_N=4, host M stays on channel 0, host N and host Q stay on channel 1, host W stays on channel 2, and host P stays on channel 3 respectively. The gateway can receive the data packets from 1-hop hosts by switching to channel 0, channel 1, channel 2, and channel 3…, periodically.

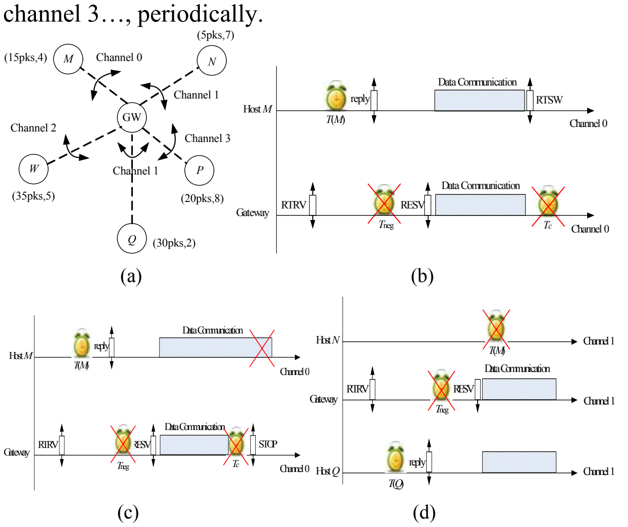


Figure 4. An illustration example with a gateway and some 1-hop hosts.

After the gateway switches to channel 0, it first broadcasts a RTRV message on this channel. Meanwhile, the timer T_{neg} is also initialized (as shown in lines 1-3 of Fig. 3). Once a 1-hop host (host M in this case) receives the RTRV message, it will defer broadcasting a reply message to compete for the access medium after a back-off time $T(M)$ (as shown in lines 22-32 of Fig. 3). When the gateway receives a reply message (from host M) before T_{neg} expires, it will broadcast a RESV message to prohibit the transmission from other 1-hop hosts which stay on channel 0 except for host M, as well as initializing a timer T_c to limit the excessive utilization duration of channel 0, which is shown in lines 4-8 of Fig. 3. When the 1-hop sender host M receives the RESV message, it begins transmitting its packets to the gateway on channel 0 (as shown in lines 33-34 of Fig. 3). If host M finishes the data transmission before the timer T_c expires, it will disconnect the data communication proactively by broadcasting a RTSW message (in line 35 of Fig. 3). The complete time lines for the gateway and host M are showed in Fig. 4(b). On receiving this message, the gateway will switch to channel 1 immediately (as shown in lines 9-14 of Fig. 3). On the other hand, if host M does not finish the data transmission after the timer T_c expires (i.e., the gateway has not received the RTSW message after T_c expires), the gateway will disconnect the data communication proactively by broadcasting a STOP message and switch to channel 1 immediately (as shown in lines 15-18 of Fig. 3). In this case, the complete time lines for the gateway and host M are showed in Fig. 4(c). On receiving this message, the sender host M will stop transmitting, which is shown in lines 36-37 of Fig. 3. After the gateway switches to channel 1, either host N or host Q has the chance to access the medium, which depends on the value of Eq. (2). Given B=50pks and N_{SC}=50, when δ=0.3 we have $T(N)=0.43T_{\text{neg}}$ and $T(Q)=0.34T_{\text{neg}}$, which means that host Q can be a sender host on channel 1. The time lines for the gateway, host N and host Q are showed in Fig. 4 (d); when δ=0.8, we have $T(N)=0.21T_{\text{neg}}$ and $T(Q)=0.63T_{\text{neg}}$ which means that host N can be a sender host on channel 1. Similarly, host W and host P also can obtain the chance to communicate with the gateway on channel 2, channel 3 respectively.

IV. Simulations and Results

A. Simulation environment

To evaluate the network performance of HTSMA, extensive simulations have been conducted. The network simulator ns-2 with CMU wireless extensions [10] is used in our simulations. Totally 100 hosts are randomly distributed in a rectangular region with the gateway located in the center of the region. Two network scenarios (with different host density) are simulated. The first (second) scenario is created by randomly placing 100 hosts in a $500m \times 500m$ ($1000m \times 1000m$) area. The basic simulation parameters are shown in Table IV. Here 50 constant-bit rate (CBR) traffic flows are generated from 50 randomly picked hosts. The data packet generation rate for each flow is varied to vary the load in the network and simulations are performed for different number of channels.

For comparison purpose, besides HTSMA, we have simulated the two-interface DCA protocol [3], and single-interface MMAC [8]. For DCA protocol, if there are C_N channels available, then one channel is designated as the control channel and the rest C_N-1 channels are used as data channels. For MMAC protocol, it is assumed that the time slot (equals to the beacon period) of 100ms is sub-divided into ad hoc traffic indication message (ATIM) window of 20ms and data window of 80ms. To compare HTSMA and MMAC fairly, the value of T_c in HTSMA is set to be the duration of data window in MMAC.

TABLE IV. SIMULATION PARAMETERS.

Parameter	**Value**
Network region size	500*m*×500*m*, 1000*m*×1000*m*
Number of hosts	100
Number of channels	4, 9
Channel bandwidth (B_w)	1Mbps
Radio propagation model	Two-ray ground model
Host radio transmission range	[100*m*, 150*m*]
Number of CBR traffic flows	50
Control message size	10bytes
Data packet size	1000bytes
Total Buffer capacity of each *h*-hop host (*h*>0)	150packets
Timer T_{neg}	20*ms*
Timer T_c	80*ms*
Adjustable parameters α, β and δ	0.3, 0.2, 0.6

The following performance metrics are collected:

(1) *Aggregate network throughput*. As a measure of total network capacity, it is defined as the received bits by all hosts and the gateway per second;

(2) *Saturation network throughput*. It is the maximal throughput that a network can accommodate, i.e., the aggregate network throughput will not be larger than the saturation network throughput even if the hosts generate more data packets. The scheme with higher saturation network throughput indicates its better scalability to accommodate heavy traffic load;

(3) *Collision ratio*. It is defined as the ratio of the number of data packets that are lost during the communication to the number of data packets that are generated by the source hosts. The scheme with lower collision ratio indicates less overhead in the network.

B. Simulation results

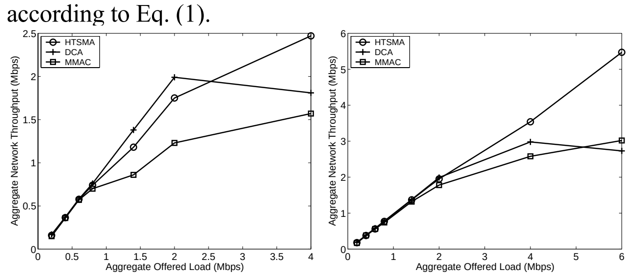
In our simulations, 10 different random network topologies are generated. For each topology, 50 sets of randomly selected source hosts are selected. Each scheme is evaluated on each set of source hosts. The simulation results are plotted using the average values derived from 500 experiments, with a 95% confidence interval. A more effective scheme is signified by the higher values in the aggregate network throughput and saturation network throughput, as well as by the lower values in the collision ratio.

Aggregate Network Throughput

Fig. 5 and Fig. 6 plot the simulation results of aggregate network throughput vs. aggregate offered load under the two network scenarios, respectively. The results of 4 channels and 9 channels for each of the two scenarios are presented in these two figures.

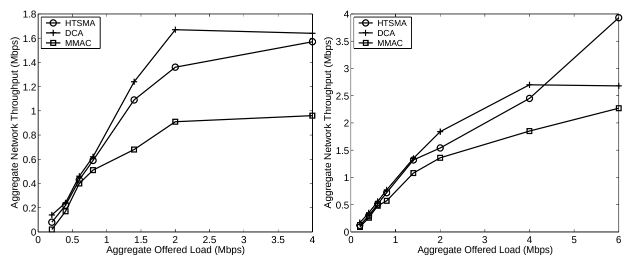
As expected, the two-interface protocol DCA generally performs better than the single-interface protocols (MMAC and HTSMA), except at high loads. The reason is that DCA uses an extra resource–the control interface (channel). The bandwidth for the control channel is traffic dependent: wide control channel may result in wastage of precious bandwidth, while narrow control channel may become a bottleneck, resulting in wastage of data channel bandwidth.

Comparing HTSMA and MMAC, HTSMA achieves better network throughput than MMAC in all cases. The reason is that in HTSMA, global time synchronization is not needed and the data communication duration can be adjusted based on the current available buffer size of the communicating host. As a result, HTSMA can decrease the idle time and thus increase the aggregate network throughput. However in MMAC, the duration of ATIM window and the data window are fixed, which are determined in advance. Besides, MMAC requires each pair of communication hosts keeps the same channel for every data window no matter if there are enough data packets to be sent during the current data window. As they cannot change the current channel until the end of data window, this results in wastage of bandwidth. In comparison, each host in HTSMA can utilize the available channel adaptively. Another reason is that in HTSMA the hop count factor is considered when we select a feasible next hop host for the sender host according to Eq. (1).



(a) 4 channels (b) 9 channels

Figure 5. Aggregate network throughput in a 500*m*×500*m* area.



(a) 4 channels (b) 9 channels

Figure 6. Aggregate network throughput in a 1000*m*×1000*m* area.

Comparing Fig. 5 and Fig. 6, it is obvious that aggregate network throughput increases with increase in the number of channels. This is due to the fact that the sender hosts can find more available channels to communicate while do not need to wait. We also note that aggregate network throughput of all three schemes decreases with increase in the deployment area. This is due to the fact that larger deployment area indicates the data packet has to traverse more hops to reach the gateway.

Saturation Network Throughput

One goal of our work is to demonstrate the performance benefit of using multiple channels in wireless networks. To show this aspect, we plot the average saturation network throughput of MMAC and HTSMA with different number of channels in Fig. 7. In this figure, single-channel IEEE 802.11 is also plotted for baseline comparison.

Fig. 7 shows that the saturation throughput of HTSMA and MMAC increases linearly with the increase of C_N from 4 to 9. And HTSMA achieves significant improvement (slightly less than C_N times for all cases) on the saturation throughput compared with the conventional IEEE 802.11 single-channel protocol and HTSMA has higher efficiency than MMAC. This is because HTSMA does not face control channel bottleneck issue as in DCA, nor does it face any control period inefficiency as in MMAC. Noticeably, for C_N=13, the saturation throughput of MMAC and HTSMA does not increase proportionally. For MMAC, this is due to the loss of bandwidth during the ATIM (control) window, where only one (common) channel is used. For HTSMA, the reason is that the hosts in the network have to switch to all channels to let their neighbors update their CUTs, which increases the control message overhead and thus affects the aggregate network throughput adversely.

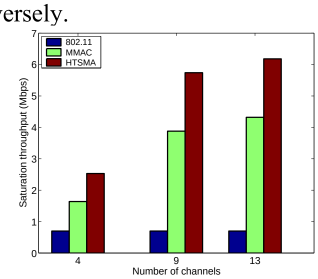
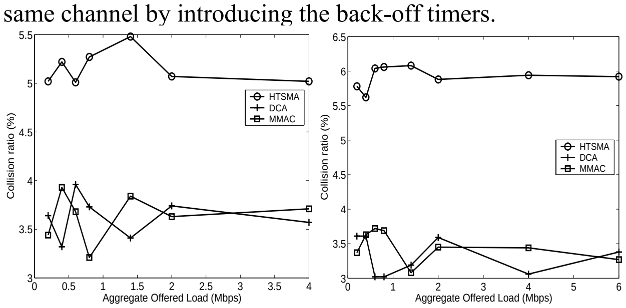


Figure 7. Saturation network throughput comparison in a 500*m*×500*m* area.

Collision Ratio

Any dynamic multi-channel protocol must ensure that the transmitter and receiver are on the same channel before communicating. To achieve this, it either ensures the communication hosts switch to a pre-determined channel at a pre-determined time (e.g., MMAC), or uses a separate control channel and interface to perform a channel negotiation (e.g., DCA). This either requires time synchronization or an additional packet interface and channel. For HTSMA, the sender host and the receiver host are guaranteed to be on the same channel by introducing the back-off timers.



(a) 4 channels (b) 9 channels

Figure 8. Collision ratio in a 500*m*×500*m* area.

Fig. 8 shows the collision ratio of the three schemes under the first network scenario. Compared with DCA and MMAC, HTSMA has a little higher collision ratio because the duration period for the communication between a pair of communication hosts is based on the estimated approach (as shown in Fig. 2 and Fig. 3), instead of the exact approach. By this way, HTSMA may adversely increase the possibility of communication interference, but it still can enhance the channel bandwidth utilization (e.g., improve the network throughput) since all hosts do not need to exchange the control messages on a common channel. As a matter of fact, in the worst case, the collision ratio of HTSMA is below 6.1%, which is still an acceptable value.

Compared with the result of HTSMA in Fig. 8(a), HTSMA in Fig. 8(b) has a slightly higher collision ratio since the communication host (including the sender host and the receiver host) has to switch to more available channels to broadcast their control messages, which introduces more collisions during the channel switching. This slight increase of collision, however, is well offset by the much improved network throughput, thus becoming invisible by the end users.

In summary, the comprehensive performance study in Section IV.B demonstrates that the proposed HTSMA scheme achieves higher aggregate and saturation network throughput and moderate collision ratio. It is expected that HTSMA is more competitive for dense networks.

V. Conclusions

In this paper, a hybrid temporal-spatial multi-channel assignment scheme (HTSMA) is proposed to solve the joint multi-channel assignment and routing problem in WMNs. Compared with other channel assignment schemes, HTSMA only requires a single radio interface for each host and does not require time synchronization for coordinating communications or a dedicated channel for exchanging the control messages. Simulation results confirm that HTSMA improves both the aggregate and saturation network throughput substantially with acceptable interference ratio. With its simplicity and flexibility, HTSMA is very suitable to be applied for large-scale WMNs.

References

[1] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: a survey," *Computer Networks*, vol. 47, no. 4, pp. 445-487, 2005.

[2] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Trans. Information Theory*, vol. 46, no. 2, pp. 388-404, 2000.

[3] 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," *Proc. Int'l Symp. Parallel Architectures, Algorithms and Networks (I-SPAN)*, pp. 232-237, 2000.

[4] A. Raniwala, K. Gopalan, and T. Chiueh, "Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks," *ACM Mobile Computing and Communications Review*, vol. 8, no. 2, pp. 50–65, 2004.

[5] S. Avallone and I. F. Akyildiz, "A channel assignment algorithm for multi-radio wireless mesh networks," *Computer Communications*, vol. 31, no. 7, pp. 1343-1353, 2008.

[6] A. Caprara, H. Kellerer, and U. Pferschy, "The multiple subset sum problem," *SIAM Journal on Optimization*, vol. 11, no. 2, pp. 308-319, 2000.

[7] A. Raniwala and T. Chiueh, "Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network," *Proc. of IEEE Annl. Conf. Computer Communications (INFOCOM)*, 2005, pp. 2223-2234.

[8] J. So and N. H. Vaidya, "Multi-channel MAC for ad hoc networks: handling multi-channel hidden terminals using a single transceiver," *Proc. of 5th ACM Int'l Sym. Mobile ad hoc networking and Computing (MobiHoc)*, 2004, pp. 222-233.

[9] J. Crichigno, M. Y. Wu, and W. Shu, "Protocols and architectures for channel assignments in wireless mesh networks," *Ad Hoc Networks*, vol. 6, no. 7, pp. 1051-1077, 2008.

[10] S. McCanne and S. Floyd. VINT Project, The ucb/lbnl/vint network simulator-ns, available at: http://www.isi.edu/nsnam/ns.