# Enhancing the Support of Multi-Hopping in an IEEE 802.11 Ad-Hoc Network


Hanem A. Eladly
*SMU, Dept of EE, PO Box 750338, Dallas, TX, USA*
heladly@mail.smu.edu

Thomas M. Chen
*SMU, Dept of EE, PO Box 750338, Dallas, TX, USA*
tchen@mail.smu.edu



## Abstract

*Previous studies have proved that the performance of IEEE 802.11 protocol is poor when used as a platform to implement multi-hop ad-hoc networks. In this paper, we elaborate on the negative behavior caused by having multiple overlapping IBSSs (Independent Basic Service Sets) operating on the same frequency channel in an 802.11 ad-hoc network. Then we propose an overlay algorithm on top of 802.11 to help discipline the interaction of overlapping IBSSs with one or more stations in the overlap zone. Such algorithm enhances the support of multi-hopping in 802.11 ad-hoc mode. Performance analysis and Simulation results of this algorithm show noticeable improvement in overall system throughput as well as stability.*


## 1. Introduction

Rapid advancement in portable computing platforms and wireless communication technology has led to significant interest in the design and development of instantly deployable, wireless networks often referred to as MANETs 'Mobile Ad-hoc NETworks'. MANETs have no fixed routers; all nodes are capable of movement and can be connected dynamically in an arbitrary manner. They have several advantages such as: on demand setup, fault tolerance and unconstrained connectivity.

Currently, the widespread-use of IEEE 802.11 cards makes this technology the most interesting off-the-shelf enabler for ad-hoc networks. However, the IEEE 802.11 protocol was not designed to be used in multi-hop wireless links. It may work well in small enterprises or homes where a single hop network may exist but not in a large-scale network where multi-hopping is a necessity.

In this paper, we will elaborate on the negative behavior of 802.11 in multi-hop ad-hoc networks. Then, we will present an overlay algorithm on top of 802.11 to correct such negative behavior.

This paper is outlined as follows, Section 2 elaborates on the negative behavior encountered in IEEE 802.11-based multi-hop networks. In section 3 we propose VAPA 'virtual access point algorithm', our overlay algorithm that counter acts such behavior. Section 4 evaluates the performance of VAPA. Section 5 presents simulation results and finally section 6 concludes our work.

## 2. Performance of Overlapping IBSSs in 802.11 ad-hoc mode

It has been shown in [2][3][4][5][6][7] that the IEEE 802.11 MAC protocol has several problems when used in a multi-hop network. Although it can support some kind of ad-hoc network architecture, which only means operation without an access point in a single-hop network composed of one IBSS, it is not intended to support the wireless mobile ad-hoc network, in which multi-hop connectivity is one of its most prominent features. Through the following simple example, we are going to elaborate on the kind of problems that may be encountered.

Assume a simple topology as in Fig.1 in which we have IBSS1 that contains nodes A, B, C and D. Node X lies within the communication range with node A only. Suppose there is an ongoing communication between nodes B and C. Node X wants to communicate with node A. It senses the medium and can not detect the ongoing communication in IBSS1 so it sends an RTS message to node A. In this case we may have two scenarios. First, IBSS1 may be physically busy and therefore the RTS message will collide at node A. Second, IBSS1 is virtually busy but physically idle. In this case, node A will receive the RTS but will cease to

0-7803-8623-X/04/$20.00 ©2004 IEEE

respond with a CTS because IBSS1 is engaged in an ongoing communication. After a time out period, node X will detect the failure of its attempt and will backoff and retry again and the whole scenario may repeat. The point to notice here is that, since node X can not sense IBSS1 medium, it will have difficulty to pick the right time to communicate with node A. It may retry several times and each time increase its backoff time until it reaches a maximum value and the packet is dropped after a specified number of retries.

Consider a more advanced scenario as in Fig.2 in which we have two simple overlapping IBSSs, IBSS1 and IBSS2. IBSS1 contains nodes B, C and D and IBSS2 contains nodes X, Y and Z. Node A belongs to both IBSSs as it is located in the overlap zone. It can communicate with all nodes in both IBSSs. For the same reasons in the previous case, nodes X, Y and Z may have difficulty communicating with node A.

Moreover, when node X, for instance, sends an RTS to node A, all the other nodes in IBSS2 will cease to engage in communication for a period equal to the NAV value included in this RTS message. Despite the fact that node X has realized after a time out period that the communication attempt has failed, the rest of the nodes will not realize that until the NAV period is over. This is known as "False Blocking" [8].

In another scenario, suppose node A wants to start communication with one of the nodes in IBSS2. Node A is exposed to the ongoing communication in IBSS1. Node A will have to verify that both media are idle before it starts any transmission. Knowing that the rest of the nodes will only have to verify that one medium is idle, node A has fewer chances to initiate any communication than any of the other nodes.

Collectively, the above reasons will result in a situation such that when an IBSS grabs the medium, it will cause the performance of the other IBSS to degrade significantly. If we look at this problem from a multi-hop perspective, we will notice that the negative behavior explained above has its direct effect on multi-hop traffic that spans across IBSSs. The traffic that is directed to the overlap zone may have difficulty to engage in a successful communication. Also, the traffic that originates from the overlap zone has fewer chances to engage in a successful communication. As a consequence, a multi-hop ad-hoc network built on top of 802.11 will suffer from poor performance.

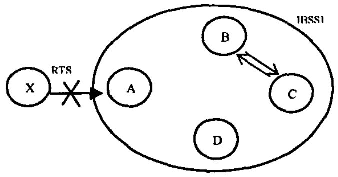


Fig. 1 Node X unable to sense IBSS1

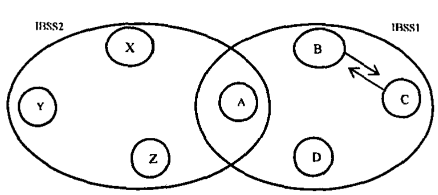


Fig. 2 Two overlapping IBSSs

## 3. VAPA: Virtual Access Point Algorithm

In this section we present an overlay algorithm deployed over the 802.11 platform. In the algorithm, nodes in overlap zones will have a critical role since they will be responsible for regulating the exchange of traffic among overlapping IBSSs operating on the same frequency channel. We call those nodes 'virtual access points' (VAPs) as they perform a similar distribution role as the access points in the IEEE 802.11 infrastructure mode.

Assume a network topology as in Fig. 3 where two IBSSs, IBSS1 and IBSS2, overlap each other with multiple VAPs. From the perspective of higher layer protocols, those VAPs provide the means for communications between different IBSSs in a multi-hop scenario. In our algorithm, we regulate the communication with VAPs. VAPs are only allowed to send or receive packets during specified periods. Normal nodes are only allowed to send packets to the VAPs during specified periods and otherwise should only send packets not directed to VAPs. This imposed regulation will eliminate false blocking problems described in the previous section. Moreover, nodes who want to communicate with VAPs will now know when it is allowed to do so. This will protect those nodes from the repeated failed attempts demonstrated as well in the previous section.

The algorithm is designed to allow one of the VAPs to seize the medium after IIT "IBSS Intercommunication Time" elapses to give a chance for intercommunciation between IBSSs.

1) During IIT period, communication between any station and any VAP is disallowed until the VAP procedure is ON.

2) VAPs contend for the medium with a high priority, every IIT period until one of them captures it, assume VAP1. The priority of VAPs is raised by setting its contention window limits to be less than the other contending nodes.

3) Once VAP1 captures the medium, i.e., the VAP procedure is ON, it will send accumulated packets until either it has no more packets to send or a maximum period of $T\_VAP_{\max}$ is reached.

4) VAP1 picks IBSSi (i = 1,2) to poll its stations for its traffic. Assume it picks IBSS2.

5) VAP1 broadcasts an RRV "Ready to ReceiVe" message for IBSS2 to indicate that it has seized the medium (VAPs alternate among IBSSs to ensure fairness). Nodes in IBSS1 should cease to transmit until the procedure is over.

6) Upon receipt of RRV, nodes in IBSS1 (including VAP2 and VAP3) that have packets for VAP1 will contend for the medium and whoever wins will respond with an RTS towards VAP1. VAP1 will respond with a CTS and normal handshake will continue.

7) VAP1 will repeat sending the RRV message until either stations in IBSS1 have no more VAP1 packets to send or a limit of *T_RRV* is reached.

8) VAP1 sends a clear message for all stations in both IBSSs to indicate that the VAP procedure is OFF and stations can resume normal communication.

9) After another IIT elapses VAPs should repeat the same procedure again. It is to be noted that, the VAP who wins the contention should alternate to a different IBSS to perform the VAP procedure. For instance if the winner VAP is VAP1 again, then the VAP procedure should be performed on IBSS1 to ensure fairness among IBSSs.

## 4. Performance analysis of VAPA

There have been various attempts to model and analyze the saturation throughput of the IEEE 802.11 DCF protocol. A model for analyzing the binary exponential backoff mechanism of 802.11 DCF was introduced in [9]. Other models have used simplified

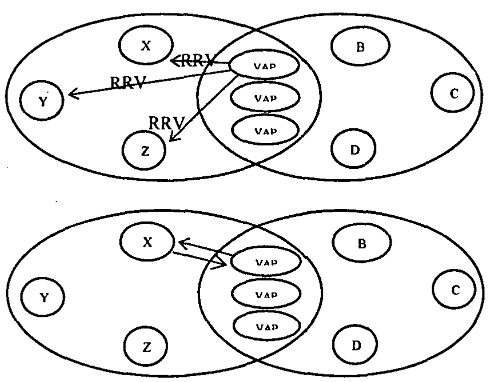


Fig. 3 VAPA in case of multiple VAPs

backoff rule assumptions such as in [10]. Another model for predicting the saturation throughput of 802.11 DCF in broadcast networks is presented in [11]. The model presented in [9] represents a model with minimal simplifying assumptions and it captures most of the details of the 802.11 DCF mode. Therefore, in this paper we choose to utilize the model in [9] to evaluate the performance of the VAPA algorithm.

Assume the topology shown in Fig 3 in which each IBSS has $(n + n_{vap})$ stations. $n_{vap}$ is the number of virtual access points in overlap zone and $n$ is the number of non VAP stations per IBSS. In this network, we may have two types of traffic. The first type of traffic is the VAP traffic which is the traffic that involves any VAP as a source or destination. The second type is the Non-VAP traffic which does not involve a VAP as the source or destination. Accordingly, we may have three cases, Non-VAP traffic, VAP traffic with a VAP destination and finally, VAP traffic with a VAP source. We can apply superposition on the three cases to study the behavior of our system.

### 4.1. Non-VAP traffic

Non VAP traffic is only exchanged during the IIT period. In this case, VAPs do not contend for the medium and the number of contending stations will be *n* stations. Therefore, we can apply the model of a single IBSS with *n* stations on this case [9]. The end result for $\tau$, the probability of transmitting a packet in a random slot time can be used:

$$\tau = \frac{2(1-2P)}{(1-2P)(W+1)+(PW)(1-(2P)^m)} \qquad (1)$$

where W is the minimum contention window size, m is the backoff stage and P is the collision probability. Also, P can be expressed as follows:

$$P = 1-(1-\tau)^{n-1} \qquad (2)$$

Solving (1) and (2) together will yield a unique solution for $\tau$ and P that can be used to calculate the IBSS throughput during the IIT period. Let $P_{tr}$ be the probability that there is at least one transmission in the considered slot time and $P_s$ be the probability that a transmission occurring on the channel is successful. Being *E[P]* the average packet payload-size, the average amount of payload-information successfully transmitted in a slot time is $P_{tr} P_s E[P]$, since a successful transmission occurs in a slot time with probability $P_{tr} P_s$. The average length of a slot time is obtained considering that, with probability $1 - P_{tr}$, the slot time ($\sigma$) is empty; with probability $P_{tr} P_s$ it contains a successful transmission and with probability $P_{tr}(1 - P_s)$ it contains a collision. Therefore, $S_1$, the saturation throughput per IBSS can be expressed as:

$$S_1 = \frac{P_s P_{tr} E[P]}{(1 - P_{tr})\sigma + P_{tr} P_s T_s + P_{tr}(1 - P_s) T_c} \quad (3)$$

Where,

$$P_{tr} = 1 - (1 - \tau)^n \quad (4)$$

$$P_s = \frac{n\tau(1 - \tau)^{n-1}}{P_{tr}} \quad (5)$$

$$T_s = RTS + SIFS + \delta + CTS + SIFS + \delta + H + E[P] + SIFS + \delta + ACK + DIFS + \delta$$

$$T_c = RTS + DIFS + \delta$$

$\delta$ accounts for the propagation delay, RTS for the length of the RTS packet, CTS for the CTS packet, H for the header length... etc.

### 4.2. VAP traffic with a VAP source

A VAP source is not allowed to send any traffic during the IIT period. It accumulates packets until the IIT counter expires, and then it contends for the medium with a high priority. Once a VAP captures the medium, it sends its traffic without any contention from the rest of the nodes. The VAP is allowed to hog the medium for a maximum duration of $T\_VAP_{\max}$. Since we assume that we operate under saturation conditions ($T\_VAP_{\max}$ limit is always reached), the saturation throughput for the VAP station can be calculated as follows:

$$S_2 = \frac{E[P]}{T_s} \quad (6)$$

The value of $S_2$ is sustained only during the $T\_VAP_{\max}$ period. Otherwise the VAP is not allowed to send any traffic.

### 4.3. VAP traffic with a VAP destination

After the expiry of the IIT counter, the VAP sends all the accumulated packets with no contention from other stations. Under saturation conditions, each station always has packets available for transmission. Therefore, the limit on the number of packets sent by the VAP is always reached. When this limit is reached the VAP ceases to transmit and it polls all the other stations in a specific IBSS for their VAP traffic. In this case, the VAP does not contend for the medium. This case resembles the first case except that the number of contending stations will be ($n + n_{vap} - 1$). Note that, the other VAPs behave like normal nodes in this case. This means that they contend with the same contention window minimum and maximum limits and should wait to be polled by the winner VAP before they send their VAP traffic.

$$\tau_{vap} = \frac{2(1-2P_{vap})}{(1-2P_{vap})(W+1)+(P_{vap}W)(1-(2P_{vap})^m)} \quad (7)$$

$$P_{vap} = 1 - (1 - \tau_{vap})^{n+n_{vap}-1} \quad (8)$$

$$S_3 = \frac{P_{s_{vap}} P_{tr_{vap}} E[P]}{(1 - P_{tr_{vap}})\sigma + P_{tr_{vap}} P_{s_{vap}} T_s + P_{tr_{vap}}(1 - P_{s_{vap}}) T_c} \quad (9)$$

Where,

$$P_{tr_{vap}} = 1 - (1 - \tau)^{(n+n_{vap}-1)} \quad (10)$$

$$P_{s_{vap}} = \frac{(n + n_{vap} - 1)\tau(1 - \tau)^{(n+n_{vap}-2)}}{P_{tr_{vap}}} \quad (11)$$

### 4.4. Overall Throughput

Applying superposition on the above three cases, the overall system throughput per IBSS can be expressed as follows:

$$S = \frac{nS_1 IIT + (1/2) S_2 T\_VAP_{\max} + (n+n_{vap}-1) S_3 T\_RRV}{(n+n_{vap})(IIT + T\_VAP_{\max} + T\_RRV)} \quad (12)$$

where $S_2$ is multiplied by (1/2) to account for a VAP source sending traffic to a single IBSS.

## 5. Simulation Results and Model Validation

We have developed a java-based simulator to simulate the basic features of the current IEEE 802.11 protocol as well as VAPA algorithm. Traffic sources are CBR (constant bit rate) and destinations are chosen based on a uniform distribution. Table I summarizes the system parameters used in our simulations. The behavior of the current 802.11 protocol is demonstrated in the first two experiments. In the first experiment we had one IBSS consisting of twenty stations. In the second experiment we had two overlapping IBSSs. Each IBSS had twenty stations. Three nodes belong to the overlap zone. The behavior of the VAPA algorithm is demonstrated in the third

TABLE I. ANALYTICAL AND SIMULATION PARAMETERS

| | |
|---|---|
| Packet payload | 2400 bits |
| ACK | 112 bits |
| RTS | 160 bits |
| CTS | 112 bits |
| Channel bit rate | 1 Mbps |
| SIFS | 28 micro seconds |
| DIFS | 128 micro seconds |
| ACK-Timeout | 300 micro seconds |
| CTS-Timeout | 300 micro seconds |

experiment. The IIT period was set to a value of 50*NAV.

From Fig.4 we can see that the first experiment achieves a saturation throughput of 60% at a traffic rate of 0.7 Mbps. The second experiment shows that IBSS1 and IBSS2 saturate at a value of approximately 40% at 0.45 Mbps (Fig.5). Also, it shows the instability resulting at higher traffic rates. The third experiment shows the effect of our proposed algorithm where we can see that IBSS1 and IBSS2 saturate at a value of approximately 60 % at 0.7 Mbps (Fig.6). Moreover the system can operate safely on this maximum saturation throughput value with out experiencing the unstable behavior encountered in experiment 2.

Figure 7 shows the effect of varying the IIT on the average normalized throughput per station. As the value of IIT increases, the VAP throughput decreases. This is because the VAP captures the medium less frequently. On the other hand, as IIT increases the throughput of each of the other stations increases. This is because the VAP interrupts the medium less frequently and those stations are allowed to receive more packets. The optimum value of IIT to operate at would be the value that enables each and every station to have a fair share of the medium i.e. the point of intersection of both curves. With the parameters used in this specific experiment, the optimum value to operate at would be approximately 25 ms.

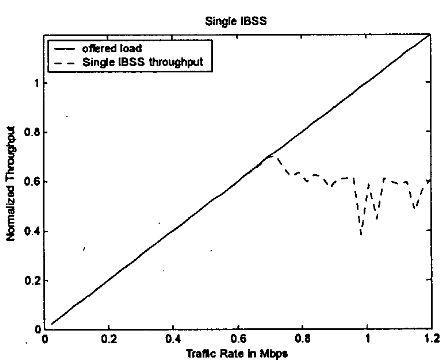


Fig. 4 Single IBSS throughput

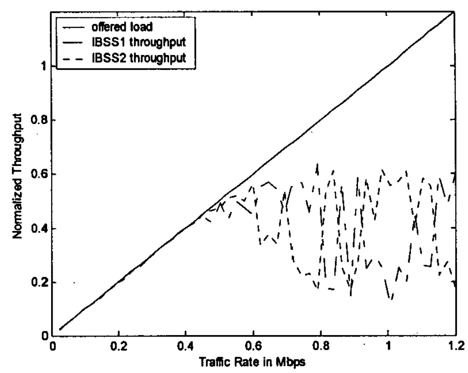


Fig. 5 Overlapping IBSSs throughput

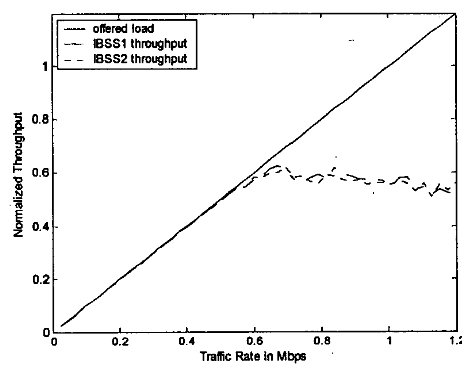


Fig. 6 Algorithm throughput

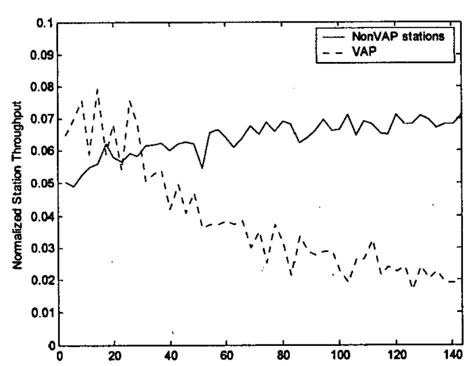


Fig. 7 Optimum IIT value

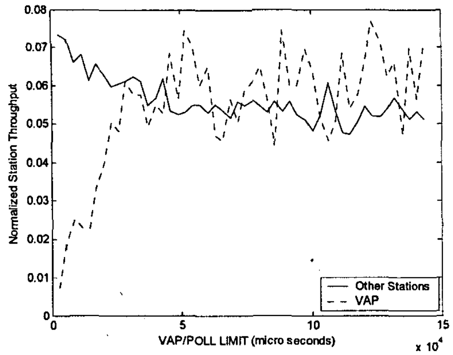


Fig. 8 Optimum VAP/POLL limit value

Figure 8 shows the effect of varying $T\_VAP_{max}/T\_RRV$ on the average normalized throughput per station. As we can see from figure, very low values of $T\_VAP_{max}/T\_RRV$ causes the VAP to have a low value for throughput and other stations to have a high value. This is because at small values of $T\_VAP_{max}/T\_RRV$, the VAP doesn't have enough chance to send the packets that it has accumulated. On the other hand, other stations are not interrupted for long periods of time and therefore have better chances to send a larger portion of packets. As the value of $T\_VAP_{max}/T\_RRV$ increases the VAP throughput improves and catches up with the rest of the stations and the other stations throughput decreases until it stabilizes at a value close to the VAP throughput. It is to be noted that as $T\_VAP_{max}/T\_RRV$ increases, its effect on the system throughput decreases as $T\_VAP_{max}/T\_RRV$ may be larger than what the station needs to send its accumulated packets for the VAP. In this case it ends the VAP poll period and normal operation is resumed. Figure 9 plots the variation of normalized throughput with respect to the number of stations. Analytical results are highly accurate and match with the simulation results.

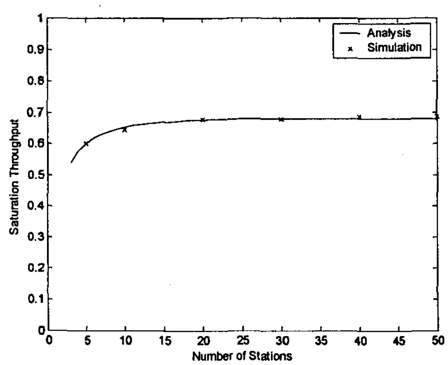


Fig. 9 Analysis versus Simulation

## 6. Conclusions

The current IEEE 802.11 standard does not handle the problems arising in the case where multiple IBSSs overlap each other. If a MANET is to be built on top of an 802.11 platform, those problems will need to be handled to yield an acceptable performance. We propose VAPA, an overlay algorithm that regulates the behavior of overlapping IBSSs operating at the same frequency channel. An analytical model for VAPA was presented and validated through simulations. VAPA yields a noticeable improvement in overall system throughput as well as system stability.

## References

[1] IEEE Std. 802.11,"Wireless LAN Media Access Control (MAC) and Physical Layer (PHY) Specifications," 1999.

[2] H. Eladly and T. Chen, "Performance Modeling of Overlapping IBSSs in the IEEE 802.11 DCF ad-hoc mode," IEEE MWSCAS 2003, Cairo, Egypt, Dec. 2003.

[3] S. Xu and T. Saadawi, "Does the IEEE 802.11 MAC Protocol Work Well In Multihop Wireless Ad-hoc Networks?," *IEEE Communications Magazine*, June 2001.

[4] M. Gerla, K. Tang, and R. Bagrodia, "TCP Performance In Wireless Multihop Networks," *IEEE WMCSA '99*, New Orleans, LA, Feb. 1999.

[5] K. Tang and M. Gerla, "Fair Sharing Of MAC Under TCP In Wireless Ad-hoc Networks," *Proc. IEEE MMT '99*, Venice, Italy, Oct. 1999.

[6] B. Bensaou, Y. Wang, and C. C. Ko, "Fair Media Access In 802.11 Based Wireless Ad-hoc Networks," *Proc. Mobihoc 2000*, Boston, MA, Aug. 2000.

[7] N. H. Vaidya, P. Bahl, and S. Gupta, "Distributed Fair Scheduling In A Wireless LAN," *Proc. Mobcom 2000*, Boston, MA, Aug. 2000.

[8] [11] S. Ray, J.B. Carruthers, and D. Starobinski, *"RTS/CTS-Induced Congestion in Ad-Hoc Wireless LANs,"* IEEE WCNC 2003

[9] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function", IEEE JSAC, Vol.18, No.3,Mar 2000, pp 535-47.

[10] H. S. Chhaya and S. Gupta, "Performance modeling of asynchronous data transfer methods of IEEE 802.11 MAC protocol," *Wireless Net-works*, vol. 3, pp. 217–234, 1997.

[11] F. Cali, M. Conti, and E. Gregori, "Dynamic tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit", IEEE/ACM Trans on Networking, Vol.8, No.6, Dec 2000, pp 785-799.