A Systematic Study of IEEE 802.11 DCF Network Optimization From Theory to Testbed

SOHAIB MANZOOR[ID]**, YACHAO YIN, YAYU GAO**[ID]**, (Member, IEEE),**
XIAOJUN HEI[ID]**, (Member, IEEE), AND WENQING CHENG, (Member, IEEE)**
School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan 430074, China

Corresponding author: Xiaojun Hei (heixj@hust.edu.cn)

ABSTRACT In IEEE 802.11 wireless local area networks (WLANs), an important technique for medium access control (MAC) is the distributed coordination function (DCF). Two access mechanisms are provided by DCF, the default basic access mechanism and the optional request-to-send/clear-to-send (RTS/CTS) mechanism. The performance of IEEE 802.11 DCF networks has been predicted recently by NS-2 simulator based on a unified analytical model presenting the delay, throughput and stability. NS-3 and OMNeT++ provide an essential platform to model IEEE 802.11 physical (PHY) and MAC layers, nevertheless the accuracy of which is yet not investigated. In this article we present two studies, first is a comparative simulation study of the unified IEEE 802.11 DCF analytical model, by considering distinct network conditions, various topologies, different access modes and discrete system parameters in NS-3 and OMNeT++. A Linux based testbed is setup to validate the mathematical model and the simulation results. The second is the optimization study to adaptively tune the RTS threshold, so that the network operates in an access mode which steers to the maximum network throughput performance. An explicit expression of RTS threshold, verified by the simulations in NS-3 and OMNeT++, is obtained in contrast to previous studies based on channel estimation and numerical calculations. Performance evaluation is done by comparing the simulation, testbed and theoretical results. This study not only proves the credibility of the theoretical model of IEEE 802.11 DCF, but also assures that the results obtained from NS-3 and OMNeT++ are persuasive and provides a foundation for RTS threshold analysis in IEEE 802.11 WLANs for practical network design considerations.

INDEX TERMS IEEE 802.11, distributed coordination function (DCF), RTS threshold, NS-3, OMNeT++, testbed, optimization, performance evaluation.

I. INTRODUCTION

WLANs have become increasingly important in the past decade due to their facile connectivity, low cost network construction and simple technical implementation [2]. The most important and basic access mechanism in the IEEE 802.11 medium access control (MAC) layer is the distributed coordination function (DCF) protocol, which is equipped with two-way handshake, basic access mechanism and the four-way handshake request-to-send/clear-to-send (RTS/CTS) mechanism. Recently Dai has proposed a unified analytical framework for IEEE 802.11 DCF networks [1] which is different from the Bianchi's model [3] as the former employees a discrete-time Markov renewal process to study

the behavior of each Head-of-Line (HOL) packet. NS-2 [4] has been used to evaluate the analytical framework showing the simplicity and accuracy of the model of IEEE 802.11 DCF networks.

The RTS/CTS mechanism unlike the basic access mechanism in the DCF requires an exchange of RTS and CTS short frames in prior to reserve the channel which degrades the network performance for small data packets. Therefore IEEE 802.11 standards demand a feasible parameter, namely RTS threshold for nodes to activate RTS/CTS mechanism for data packet lengths that will not deteriorate the network performance in WLANs [5]. To switch between the appropriate access mode is crucial in terms of achieving high throughput in WLANs. Number of studies have focused on the optimal RTS threshold for IEEE 802.11 WLANs [6]–[10]. A set of complex non-linear equations based on the classic

Bianchi's model [3] were used to calculate the RTS threshold [7], [9]. The results show that the RTS threshold increase when the number of nodes decrease or the data rate increase [8], [9]. Various network assumptions and implicit nature of solution make the relationship of RTS to key system parameters uncertain, leading to tremendous difficulties in switching between basic and RTS/CTS mechanisms in practical WLANs.

Students, developers and researchers study the behavior of real time systems by performing simulations, in order to get flexibility in terms of repeatability, scalability and stability [11]–[14]. Fig. 1 shows the comparison between the performance evaluation tools. NS-2, NS-3 [15] and OMNeT++ [16] shows positive trends in reproducibility, low cost environment and least complexity. These simulators have a drawback of being least realistic in comparison to the testbed tool such as PlanetLab [17]. NS-3 and OMNeT++, being the emerging simulators, are commonly used these days. NS-3 helps in providing a real time simulation environment, with a python scripting interface and a modular type structure written in C++, which supports software integration. OMNeT++ is also a modular, extensible and component based C++ library with GUI support which allows the kernel to be embedded easily in user applications. Both the simulators have built in IEEE 802.11 modules following the MAC and DCF, rules and regulations. A very few studies have considered the validation of NS-3 and OMNeT++ PHY/MAC layer due to its complexity. The studies that validate the PHY and MAC layer in NS-3 and OMNeT++ are [18]–[26].
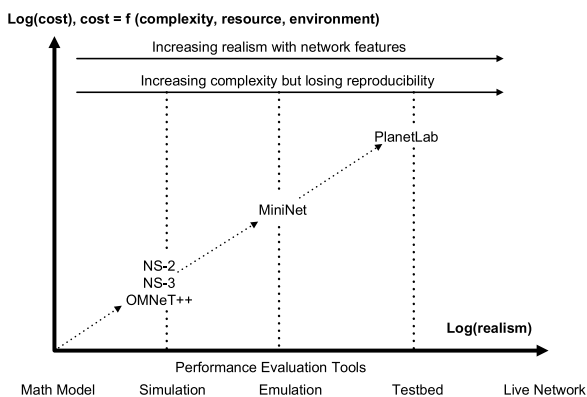

FIGURE 1. A comparison between performance evaluation tools.

In this article we present two studies, first is the comparative simulation study of the unified IEEE 802.11 DCF analytical model presented by Dai and Sun [1], in NS-3 and OMNeT++. The WiFi modules of both the simulators are compared in detail and a Linux based testbed is setup to validate the mathematical analysis and the results obtained from the simulations. Distinct network conditions, various topologies, discrete system parameters and two access mechanisms

are utilized to validate the MAC/PHY layers of NS-3 and OMNeT++ against Dai's model [1]. In the second study we address an open issue of how to attain an optimal throughput performance in a network by automatically and accurately switching between the basic access and RTS/CTS mechanism. An explicit expression for optimal RTS threshold in IEEE 802.11 WLANs is formulated. In continuation to Dai's model [1], a closed-form solution is used to relate RTS threshold to key network parameters: 1) the data rate R_D, 2) number of nodes n, 3) the initial backoff window size W and 4) the basic rate R_B. The results further show that the network throughput performance achieved from optimal RTS threshold is superior than the standard setting. This work to the best of our knowledge is first of its kind that can serve as the validation of NS-3 and OMNeT++ MAC model for IEEE 802.11 DCF networks and is extendable to practical implementation, as the APs can collect the optimal RTS threshold with the help of a closed-form expression and determine an appropriate access mode.

The core contributions of this study are summarized as follows:

- We validate the PHY and MAC layers of NS-3 and OMNeT++ by an unified analytical model [1], which incorporates the fundamental features of IEEE 802.11a DCF networks by considering discrete system parameters, different access modes, various topologies and distinct network conditions.
- We formulate an explicit expression for optimal RTS threshold and optimize it adaptively in NS-3 and OMNeT++, which guarantees the best network throughput performance and the ease to be adapted in the practical network design for IEEE 802.11 WLANs.
- We analyze and compare the main features, credibility and performance of WiFi modules in NS-3 and OMNeT++, while simulating the theoretical model of IEEE 802.11 DCF and the optimal RTS threshold expression obtained in this study.
- We design a Linux based testbed, which subsumes a key system parameter such as the initial backoff window size W for different number of wireless nodes n to validate the unified analytical model [1] and the simulation results.

The remainder of this article is organized as follows. Section II explores the related work. Problem formulation based on the previous research is dealt within Section III. Section IV presents the significant results of the unified analytical framework. Section V exposes the analysis and formulation of the optimal RTS threshold. Section VI recapitulates the WiFi modules in NS-3 and OMNeT++. Section VII discusses the experimental setup for simulation and testbed. Section VIII describes how the simulators are tuned to get the simulation results and the analogy between simulation results, testbed results and theoretical results is discussed. Finally the paper is concluded in section IX. Table 1 and 2 show the list of abbreviations and symbols, respectively, used in this study.

TABLE 1. List of abbreviations.

Abbreviation	Description
ACK	Acknowledgment
AP	Access point
API	Application programming interface
CDF	Cumulative distributive frequency
cfg80211	Configuration of APIs for 802.11 devices in Linux
CMU	Cafe emulation
CTS	Clear to send
CW	Contention window
DCF	Distributed coordination function
DIFS	DCF interframe space
DoS	Denial of service
EDCA	Enhanced distributed channel access
HCF	Hybrid coordination function
HOL	Head of line
hostapd	Host access point daemon
IPV4	Internet protocol version 4
IPV6	Internet protocol version 6
iw	Configuration tool for wireless devices
LDP	Label distribution protocol
MAC	Medium access control
MH	MAC header
MPLS	Multi protocol label switching
NIC	Network interface card
nl80211	Netlink interface public header
PH	PHY header
PHY	Physical layer
PPP	Point to point protocol
RSPV-TE	Resource reservation protocol-traffic engineering
RSSI	Received signal strength indicator
RTS	Request to send
RT*	RTS threshold
SCTP	Stream control transmission protocol
SIFS	Short interframe space
SSH	Secure shell
SSID	Service set identifier
Sta	Station
TCP/IP	Transmission control protocol/Internet protocol
Txop	Transmission opportunity
UDP	User datagram protocol
VoIP	Voice over internet protocol
WLAN	Wireless local area network
WiFi	Wireless fidelity

TABLE 2. List of symbols.

Symbols	Description
σ	Length of time slots
$\hat{D}$	Throughput (network sum rate)
F_i	State of collision
K	Cutoff phase
λ	Traffic arrival rate
$\hat{\lambda}_{out}$	Normalized throughput
n	Number of nodes
P	Idle channel
p_A	Non-zero root of fixed point equation
PL	Packet payload length
R_B	Basic rate
R_D	Transmission rate
R_i	State of waiting for a request
T	State of successful transmission
τ_F	Holding times for collision states
τ_T	Holding times for successful transmission
τ_F^{ba}	Holding times for collision states in basic access
τ_T^{ba}	Holding times for successful transmission in basic access
τ_F^{rts}	Holding times for collision states in RTS/CTS
τ_T^{rts}	Holding times for successful transmission in RTS/CTS
W	Window size

II. RELATED WORK

A. VALIDATION OF PHY AND MAC LAYERS IN NS-3 AND OMNeT++

Modeling and simulation of IEEE 802.11 "g" standard is performed in OMNeT++ simulator [27]. The performance evaluation is performed using round trip times between two ping messages neglecting major performance metrics such as throughput versus the number of nodes, packet payloads, cutoff phase and initial backoff window sizes. The IEEE 802.11 "b" standard is validated in NS-3 against the results obtained from a CMU wireless network emulator [28]. The authors focus on the PHY layer while ignoring the aspects of the MAC layer. The validation of IEEE 802.11 MAC model is performed in NS-3 using a testbed [29]. The authors consider different scenarios for the validation approach, such as communication with single pair of nodes, communication in the presence of hidden nodes and communication using either saturated traffic or VoIP flows, neglecting various network topologies and different access modes. The reliability of OMNeT++ in WLANs is validated through a testbed [30], where the throughput, end-to-end delay and the packet loss ratio are taken as performance metrics. DoS attacks are the main area of concern for the authors neglecting the tuning of system parameters and different network conditions such as saturated and unsaturated networks. The validation of MAC layer in NS-3 against a bi-dimensional Markov chain model for saturated throughput under ideal conditions is reported in [31]. The throughput is only measured against the number of nodes neglecting the factors such as packet payloads, initial backoff window sizes and cutoff phase. A software defined testbed is designed to see the performance of WiFi DCF networks [32], [33]. The authors consider measuring the throughput against the number of nodes for different contention window sizes. The study however does not include tuning other network parameters such as traffic arrival rate,

cutoff phase and transmission rates to evaluate the throughput performance. IEEE 802.11p and short range cellular-vehicle-to-anything (C-V2X) technologies are used study and support the idea of short-range wireless communications [34], [35]. The performance and the contributions of the MAC and PHY layers are isolated and discussed. The study provides limited information in terms of simulator WiFi modules, which are in charge of carrying PHY and MAC operations. The key system parameters that effect the network throughput performance and different network conditions such as saturated and unsaturated networks are ignored.

B. RTS THRESHOLD IN IEEE 802.11 WLANs
The impact of RTS threshold on throughput is studied in multi-rate networks [36]. The simulations are performed for various rates in IEEE 802.11 ''b'' standard. The main findings of the paper are the analysis of when to increase or decrease the RTS threshold for a specific data rate or number of nodes. No optimal tuning of RTS threshold was performed. Tunning of RTS threshold is performed based on packet distribution [37]. Transmission range and the interference ranges are assumed equal to expose the hidden node problem. Cumulative distributive function (CDF) is used to predict the value of RTS threshold. Simulations are performed with uniform packet generator with stationary mobile nodes. The scheme only relies on the network size considering the number of nodes while ignoring other network parameters such as data rates, basic rates, packet payloads and initial backoff window sizes. Performance of DCF MAC is presented by varying RTS threshold values [38]. QoS parameters such as end-to-end delay, media access delay, retransmission attempts and network load are evaluated at fixed RTS threshold values, such as 128, 256, 512 and 1024 bytes. Automatic tunning of RTS threshold is not performed. Optimal throughput performance is analyzed for multi-hop networks using RTS/CTS [39]. One-way flow in string multi-hop networks is considered for the performance evaluation. The analysis focuses on the transmission failure probability considering the network allocation vector (NAV). The study neglects the adaptive network switching between the basic access and the RTS/CTS modes.

An algorithm is designed to handle the RTS threshold based on packet delivery ratio [40]. The packet delivery ratio is considered as the threshold to decide which access mode needs to be chosen by the network. The algorithm design, to calculate the RTS threshold values, is based on the classic Bianchi's model [3]. The traffic is kept as static and performance evaluation is done against only the number of nodes which do not prove the effectiveness of the scheme, furthermore complex non-linear analysis is made to determine the RTS threshold values. The study regarding the importance of RTS threshold and the guideline to adjust the RTS threshold parameter automatically, is made in [41]. The correlation of the number of nodes to RTS threshold is only considered while the other factors such as packet arrival rate and packet lengths are ignored. RTS/CTS mechanism is considered superior to basic access in terms of network

throughput and authors suggested to use the RTS/CTS mechanism rather than the basic access for all packet sizes. Ignoring the basic access is not an appropriate approach and against the IEEE 802.11 standard [5]. Our proposed approach tunes the RTS threshold automatically and efficiently to switch between either RTS/CTS mechanism or basic access in order to get the maximum network throughput performance. The IEEE 802.11 binary exponential backoff (BEB) algorithm is modified by a history-based adaptive backoff (HBAB) to enhance the QoS performance [42]. HBAB relies on the adaptive delta modulation scheme used in communication. The performance metrics considered are the average packet delay and packet delivery fraction. The only system parameter considered is the contention window. The results taken are in the standard setting ignoring the optimal tuning of the RTS threshold parameter. The research gaps related to RTS threshold in previous studies and the proposed solution are summarized in Table 3.

III. PROBLEM STATEMENT
In this section, the research gaps related to DCF simulation and RTS threshold analysis, that arose in previous research studies as discussed in section I and II are summarized. The hierarchy of the problems addressed in this study can be visualized in Fig. 2.

A. SIMULATORS PHY/MAC LAYER VALIDATION
The studies focusing on NS-3 and OMNeT++ PHY/MAC layer validation are few due to the complexity involved in tuning the system parameters for PHY and MAC layers. A detailed literature survey has revealed that for the PHY/MAC layer validation, the previous research only considered a single simulator or a testbed and did not include all the key system parameters, different network conditions, distinct network topologies and various access modes. However by looking at Fig. 2a, this study, for validating the PHY and MAC layers in NS-3 and OMNeT++, takes into account a mathematical model, two simulators and a testbed including all key system parameters, different network conditions, topologies and access modes.

B. RTS THRESHOLD
The RTS threshold allows the network to achieve optimal performance by switching to an appropriate access mode. It can be seen from Fig. 2b that even for small packet lengths, RTS/CTS can replace basic access when collisions are higher. This is due to the less collision duration in RTS/CTS mechanism. The detailed comparison between basic access and RTS/CTS mechanism is presented in section V-A. Research involving RTS threshold, however presented complex methods to tune RTS threshold, which were not practically feasible. Previous studies ignored the adaptive tuning of RTS threshold in emerging simulators like NS-3 and OMNeT++ and related RTS threshold to limited network parameters. This made it difficult to analyze, optimize, understand and design a practicable RTS threshold. This study provides a

TABLE 3. RTS threshold study comparison table.

S.No	Authors	Main findings (research gaps)	Proposed solutions
1	Wang et al. [36]	Simulations are performed for multiple transmission rates. RTS threshold adjusted manually for different transmission rates.	Adaptive tunning of RTS threshold performed.
2	Ahsan et al. [37]	RTS threshold tuned in accordance to packet distribution. Stationary wireless nodes are deployed. RTS threshold predicted using cumulative frequency distribution. Performance evaluation considered only the number of nodes.	RTS threshold is calculated through two simulators and a mathematical model. Wireless nodes are mobile. The RTS threshold is not related only to number of nodes but transmission rates and window size.
3	Singh et al. [38]	Fixed RTS rates are used such as 128, 256, 512 and 1024 bytes.	Adaptive tunning of RTS threshold performed according to network conditions.
4	Sugimoto et al. [39]	Throughput performance analyzed through transmission failure probability. Network mode switching neglected.	The network mode selection is tunned adaptively.
5	Mjidi et al. [40]	RTS calculated based on packet delivery ratio. Bianchi's model [3] is used. RTS threshold adjusted manually.	RTS threshold calculated based on network conditions. Dai's model [1] is used. RTS adjusted automatically.
6	Sheu et al. [41]	RTS threshold relation to number of nodes (n) is considered. Basic access mode is ignored.	RTS threshold is related to all the key system parameters (n, W, R_D and R_D). Both basic access and RTS/CTS modes are considered.
7	Nasir et al. [42]	QoS performance analyzed by changing binary backoff algorithm. Only contention window is considered as a system parameter. IEEE 802.11 standard settings are used.	Network performance is analyzed by considering key system parameters (n, W, R_D and R_D). IEEE 802.11 standard and proposed optimal settings are used.

vivid expression for RTS threshold and relates it to the key network parameters such number of nodes n, initial back-off window size W, data rate R_D and basic rate R_B. The RTS threshold is then optimized adaptively in NS-3 and OMNeT++ and is practically feasible to be implemented in IEEE 802.11 WLANs.

C. SIMULATORS WiFi MODULES

The major contribution in the validation of the PHY and MAC layers of any simulator is of the credibility of its WiFi module. Unfortunately the detailed analysis and performance comparison of WiFi modules in NS-3 and OMNeT++ has been ignored in the previous studies. This study provides a detailed analysis and comparison of the WiFi modules in both the simulators. The basic submodules constituting the WiFi modules in NS-3 and OMNeT++ are shown in Fig. 2c.

D. TESTBED DESIGN

Past research has shown that an immense complexity is involved in designing a testbed for DCF validation. The

complexity incurred extra cost while increasing the number of wireless nodes and researchers found it hard to configure and tune the DCF parameters each time, to achieve different performance attributes. The testbed designed in this study is software defined which makes the configuration relatively easy. Portable and low cost, USB WiFi cards serve as wireless nodes as seen in Fig.2d, which can be easily increased in number without incurring extra costs and complexity to the testbed design.

IV. PRELIMINARY ANALYSIS AND MODELING

The unified analytical framework for IEEE 802.11 DCF networks [1] is presented in this section along with the summary of the interpretation of throughput $\hat{D}$ for both saturated and unsaturated networks. The objective is to compare the accuracy of NS-3 and OMNeT++ MAC layers to the analytical results obtained by IEEE 802.11 DCF theoretical model, which has been validated by a well known simulator, NS-2 in [1].
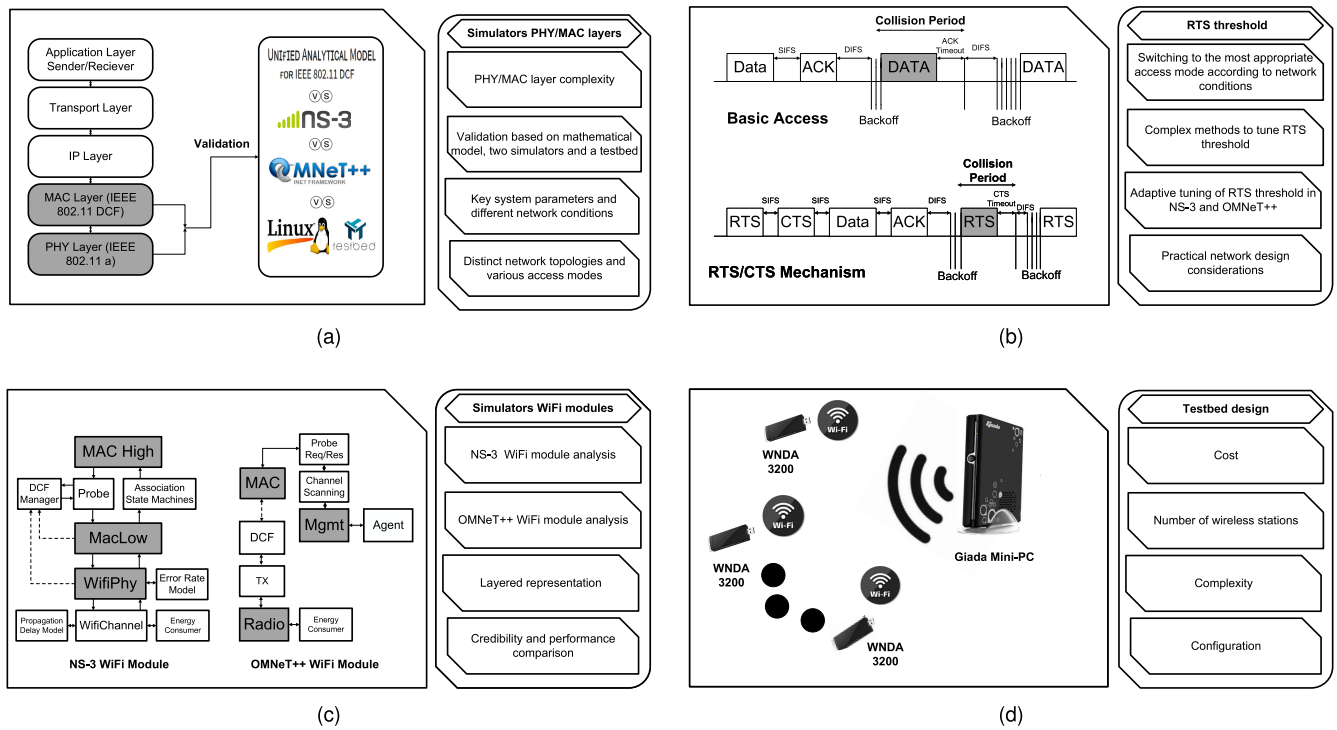
FIGURE 2. Hierarchy of the problems in DCF simulation and RTS threshold optimization addressed in this study. (a) PHY/MAC layer validation. (b) RTS threshold analysis. (c) Simulators WiFi modules. (d) Testbed design.
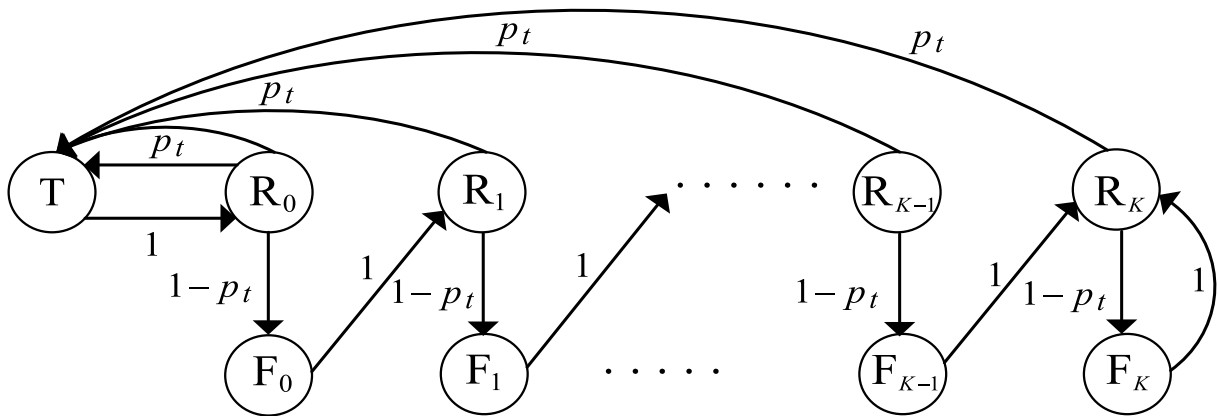

FIGURE 3. State transition of each HOL packet in an IEEE 802.11 DCF network by an embedded Markov Chain $\{X_j\}$ [1].

A. UNIFIED ANALYTICAL FRAMEWORK FOR IEEE 802.11 DCF NETWORKS

To model the behavior of each HOL packet as discrete-time Markov renewal process, a unified analytical framework for IEEE 802.11 DCF networks is established. An embedded Markov chain X_j is shown in Fig. 3, where the jth transition state, state of successful transmission T, state of collision F_i, and state of waiting for request R_i of a HOL packet are presented.

In the IEEE 802.11 DCF networks we consider n number of nodes with an infinite buffer space transmitting in a noise less channel and each head-of-line (HOL) packet can be retransmitted infinite number of times. The backoff parameters such

as cutoff phase K and initial backoff windows size W are considered identical for each node. Each node is assumed to have a traffic arrival rate of λ. The percentage of time when successful transmissions are made in an unsaturated network in [1], is given by the normalized throughput $\hat{\lambda}_{out}$ as

$$\hat{\lambda}_{out} = n\lambda. \tag{1}$$

When a network gets saturated, every node still has the capacity to transmit a packet so $\hat{\lambda}_{out}$ can be derived as

$$\hat{\lambda}_{out} = \frac{-\tau_T p_A \ln p_A}{1 + \tau_F - \tau_F p_A - (\tau_T - \tau_F) p_A \ln p_A}. \tag{2}$$

Holding times representing the successful transmission and collision states (measured in units of time slots) of HOL

packets are denoted by τ_T and τ_F respectively, where p_A is the non-zero root of the fixed-point equation presenting the steady-state probability that the HOL packets are transmitted successfully in an idle channel p:

$$p = \exp\left\{-\frac{2n}{W \cdot \left(\frac{p}{2p-1} + \left(1 - \frac{p}{2p-1}\right)(2(1-p))^K\right)}\right\}, \quad (3)$$

where W is the initial backoff window size, K is the cutoff phase ($K = \log_2(\frac{CW_{\max}}{CW_{\min}})$), and n is the number of nodes.

$\hat{\lambda}_{out}$ does not reflect any information regarding the data transmitted in terms of bits per seconds, rather it helps to evaluate how efficient the time is consumed for successful transmissions. Therefore, the paper only focuses on the throughput which is defined as the actual data that is transmitted successfully per second. The throughput $\hat{D}$ in an unsaturated network can be written as

$$\hat{D} = \hat{\lambda}_{out} \cdot \frac{\frac{8PL}{R_D\sigma}}{\tau_T} \cdot R_D = \frac{8PL \cdot n\lambda}{\sigma \tau_T}, \quad (4)$$

from Eq. 1, where PL is the packet payload length measured in the units of bytes.

In a saturated network, the throughput $\hat{D}$ is determined from the time consumed by packet payload transmission for each successful transmission, the normalized throughput $\hat{\lambda}_{out}$ and the rate at which the transmission is made R_D. Using Eq. 2 $\hat{D}$ is given as

$$\begin{aligned}\hat{D} &= \hat{\lambda}_{out} \cdot \frac{\frac{8PL}{R_D\sigma}}{\tau_T} \cdot R_D \\ &= \frac{-8PL \cdot p_A \ln p_A}{\sigma\left(1 + \tau_F - \tau_F p_A - (\tau_T - \tau_F)p_A \ln p_A\right)}, \quad (5)\end{aligned}$$

where σ represents the length for time slots and is measured in units of μs.

B. BIANCHI's MODEL VERSUS DAI's MODEL
Bianchi proposed the most widely used model for IEEE 802.11 DCF networks [3]. A two dimensional Markov chain was proposed to support the backoff process of each saturated node. The main differences between classic Bianchi's model and unified Dai's model are:

1) Bianchi's model presents results for a saturated network as opposed to Dai's model where both the saturated and unsaturated networks are taken into account.
2) Bianchi's model only focuses on throughput whereas Dai's model presents a study for stability, throughput and delay performance. In a saturated throughput scenario the results of both the models are rational.
3) Dai's model is not only limited to homogeneous IEEE 802.11 DCF networks but it is extended to various heterogeneous IEEE 802.11 DCF networks [43]–[46].
4) The studies influenced by Bianchi's model present complex numerical calculations which are not easy to optimize. Dai's model being explicit in nature on the other hand presents explicit expressions for network

throughput and optimal backoff parameters, which are used in numerous follow up studies for both homogeneous and heterogeneous IEEE 802.11 DCF networks [1], [43], [45]–[47].

As the Bianchi's model is limited to the evaluation of throughput in a saturated network hence we use Dai's model to validate the WiFi modules of NS-3 and OMNeT++. In the past the accuracy of Dai's model has been verified by the NS-2 simulator, so we further enhance its reliability by performing a comparative simulation study using NS-3 and OMNeT++ WiFi modules.

V. OPTIMAL RTS THRESHOLD ANALYSIS
In this section, the optimal tuning of RTS threshold in IEEE 802.11 DCF networks is presented by applying the results from section IV. A brief review of the DCF protocol, considering the basic access and RTS/CTS mechanism is provided, which helped in characterizing the RTS threshold, and the inference on practical network design for IEEE 802.11 WLANs.

A. BASIC ACCESS VS RTS/CTS MECHANISM
To access the channel, the DCF protocol works in two access modes one is the default basic access and other is the optional RTS/CTS mechanism. IEEE 802.11 standard [5] provide the detailed specifications for DCF protocol. We present the basic working principle for the DCF.

Basic access is shown in Fig. 4, where, if the channel remains idle for a duration of Distributed Interframe Space (DIFS), the node will transmit a packet and wait for the acknowledgment (ACK) frame after waiting again for an idle time of Short Interframe Space (SIFS). If the ACK frame in not received in the ACK time-out period, the node will again retransmit the packet after enabling the backoff mechanism. It is important to find the holding times for the collision states and successful transmission for basic access and RTS/CTS mechanism, because in the parameter settings of NS-3 and OMNeT++, the node after encountering a collision needs to wait ACK time-out and CTS time-out period. Therefore in the basic access, the holding times for successful transmission and collision states can be written as

$$\tau_T^{ba} = \frac{\left(\frac{8PL}{R_D} + \frac{8MH}{R_D} + 2PH + \frac{8ACK}{R_B} + SIFS + DIFS\right)}{\sigma} \quad (6)$$

and

$$\tau_F^{ba} = \frac{\frac{8PL}{R_D} + \frac{8MH}{R_D} + PH + ACKTimeout + DIFS}{\sigma} \quad (7)$$

respectively. SIFS, DIFS and PHY header (PH) are in the units of μs, R_D and R_B are the data rate and basic rate respectively and measured in units of Mbps, MAC header (MH) and ACK frames are in the units of bytes.

The RTS/CTS mechanism is shown in the Fig. 4b, where the node sends the RTS frame after waiting for a duration of DIFS and receives a CTS frame after waiting a duration of SIFS. Packet transmission is initiated by the node after
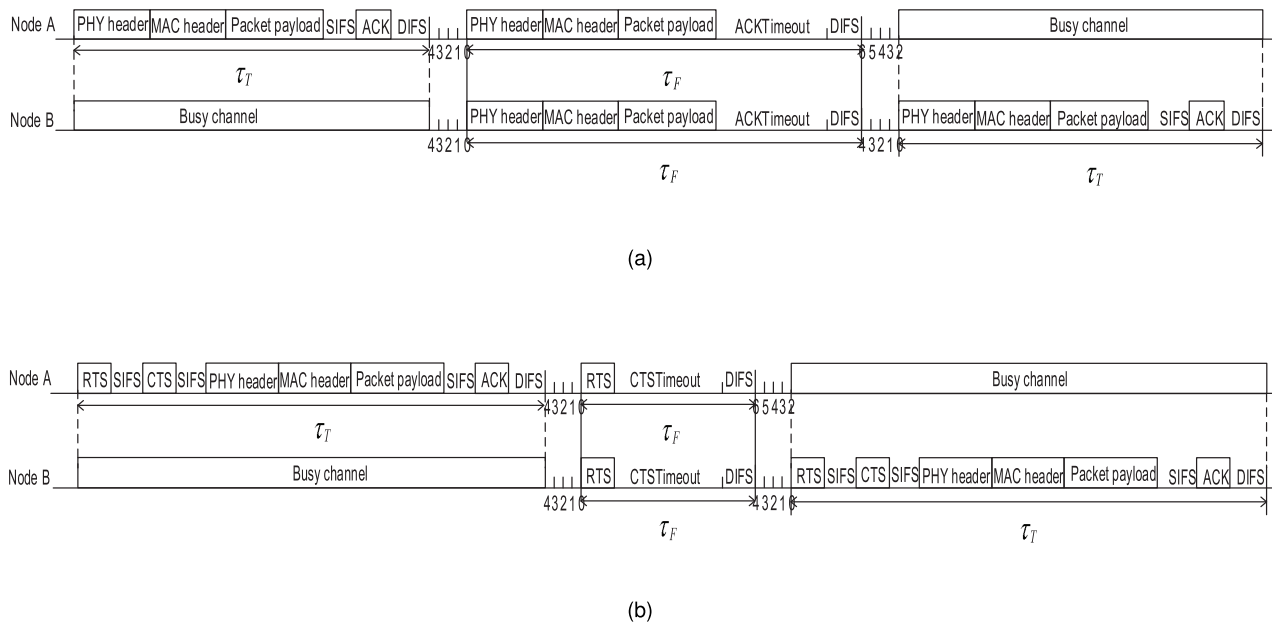
FIGURE 4. Holding time representation in collision states and successful transmissions in IEEE 802.11 DCF networks (a) basic access (b) RTS/CTS mechanism.

waiting another duration of SIFS. If the CTS frame is not received in the CTS time-out period the node will enable the backoff process and reserve the channel again. In the RTS/CTS mode the holding times for successful transmission and collision states are derived as below

$$\tau_T^{rts} = \frac{\frac{8PL}{R_D} + \frac{8MH}{R_D} + 4PH + \frac{8(RTS+CTS+ACK)}{R_B} + 3SIFS + DIFS}{\sigma} \tag{8}$$

and

$$\tau_F^{rts} = \frac{\frac{8RTS}{R_B} + PH + CTSTimeout + DIFS}{\sigma} \tag{9}$$

respectively, where RTS and CTS are measured in the units of bytes.

By comparing Eq. 6 and 7 to Eq. 8 and 9, it can be observed that the basic access has a shorter holding time in the successful transmission state than the RTS/CTS mechanism, i.e., $\tau_T^{rts} > \tau_T^{ba}$, due to shorter overhead in the frame exchange process of the basic access. On the other hand the RTS/CTS mechanism enjoys less duration collisions because the RTS frame is usually shorter than the data packet, i.e., $\tau_F^{rts} < \tau_F^{ba}$. Therefore the basic access is expected to achieve less network throughput as compared to RTS/CTS mechanism, when the packet payload length *PL* is larger.

B. OPTIMAL RTS THRESHOLD FORMULATION
In the RTS threshold analysis the RT^* is defined as the optimal RTS threshold and is measured in the units of bytes. RT^* corresponds to a decision where the RTS/CTS mechanism can replace the basic access when the packet length is larger than the RT^*, i.e., $\hat{D}^{rts} > \hat{D}^{ba}$ if $PL > RT^*$. By integrating

Eq. 5-9, the optimal RTS threshold for the IEEE 802.11 DCF networks can be acquired as

$$RT^* = \frac{\frac{RTS}{R_B}(1-p_A) - (\frac{CTS}{R_B} + \frac{SIFS}{4} + \frac{PH}{4})p_A \ln p_A}{1 - p_A + p_A \ln p_A} \cdot R_D - MH, \tag{10}$$

where p_A denotes the steady state point as given in Eq. 3, and is a function of initial backoff window size W, cutoff phase K and the number of nodes n as seen in Eq. 3. Typical values of the overhead are stated by IEEE 802.11 standard [5] and summarized in table 5. By using these values, the RTS threshold can be rewritten as

$$RT^* = \frac{\frac{20}{R_B}(1-p_A) - (\frac{14}{R_B} + 9)p_A \ln p_A}{1 - p_A + p_A \ln p_A} \cdot R_D - 36. \tag{11}$$

After substituting the system parameters values, we observe from Eq. 11 that the RTS threshold RT^* relies on the cutoff phase K, initial backoff window size W, the basic rate R_B, the transmission rate R_D and the number of nodes n.

By looking at the Fig. 5 we see how the RT^* changes with the system parameters. Figs. 5a and 5b show that the RT^* decreases as the number of nodes n increase and the initial backoff window size W decrease. As the collision time in basic access is higher than the RTS/CTS mechanism, i.e., $\tau_F^{rts} < \tau_F^{ba}$, so for small packet lengths, when more collisions occur in the network, the RTS/CTS mechanism can outmatch the basic access. This happens when more wireless nodes connect to the network, or these nodes embrace small initial backoff window sizes. Figs. 5c and 5d illustrate that RT^* decreases with a small transmission rate or when the basic rate rises, i.e., when the RTS frame overhead time
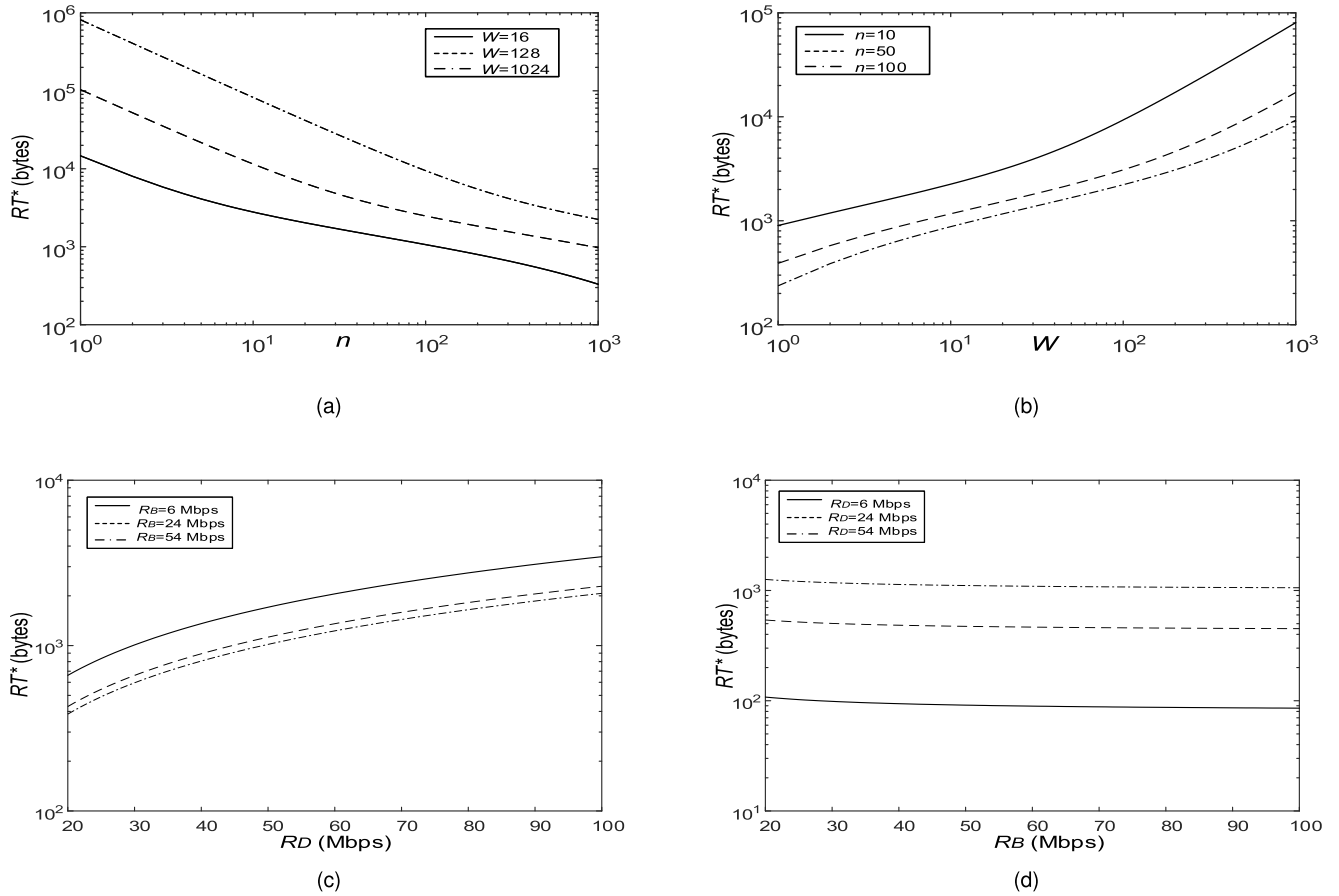
FIGURE 5. RTS Threshold RT^*. (a) RT^* against the number of node n. $K = 6$. $R_D = 54$ Mbps. $R_B = 6$ Mbps. (b) RT^* against the initial backoff window size W. $K = 6$. $R_D = 54$ Mbps. $R_B = 6$ Mbps. (c) RT^* against the transmission rate R_D. $n = 20$. $W = 16$. $K = 6$. (d) RT^* against the basic rate R_B. $n = 20$. $W = 16$. $K = 6$.

in RTS/CTS mechanism is less as compared to the packet transmission time in the basic access.

An explicit expression for RTS threshold RT^* is obtained in order to maximize the network throughput performance. The analysis show that the RT^* increase with an increase in transmission rate R_D and rise in initial backoff window size W, but steadily decrease when the basic rate R_B or the number of nodes n increase.

C. INFERENCE ON NETWORK DESIGN

The analysis regarding the RT^* in this article, helps in the selection of appropriate access mode along with the practical network design for IEEE 802.11 DCF networks. The default value of RT^* is fixed at 2347 bytes in the current IEEE 802.11 WLANs [5], which makes it difficult for the network to switch between the most appropriate access mode and has to bear rate loss. The RT^* needs to be tuned adaptively in reference to the number of nodes in order to guarantee that the network always switches to the access mode that achieves the highest throughput according to the Eq. 10. By analyzing the MAC header, the AP can extract information regarding the number of nodes which leads to the calculation of the RT^* as given in Eq. 10.

TABLE 4. Typical Basic Rate R_B and Data Rate R_D Values showing the Optimal RTS Threshold RT^* (bytes) for IEEE 802.11 DCF Networks. $K = 6$. $W = 16$. $n = 50$.

R_D \ R_B	6	9	12	18	24	36	48	54
6	126	207	287	449	611	934	1258	1419
9	101	169	238	374	511	785	1058	1195
12	88	151	213	337	461	710	958	1083
18	76	132	188	300	411	635	859	971
24	70	122	175	281	386	597	809	914
36	63	113	163	262	362	560	759	858
48	60	108	156	253	349	542	734	830
54	59	107	154	250	345	535	726	821

The basic rates and the transmission rates are depicted in Table 4 for the optimal RTS threshold RT^* for the current standard settings of backoff parameters, i.e., $K = 6$ and
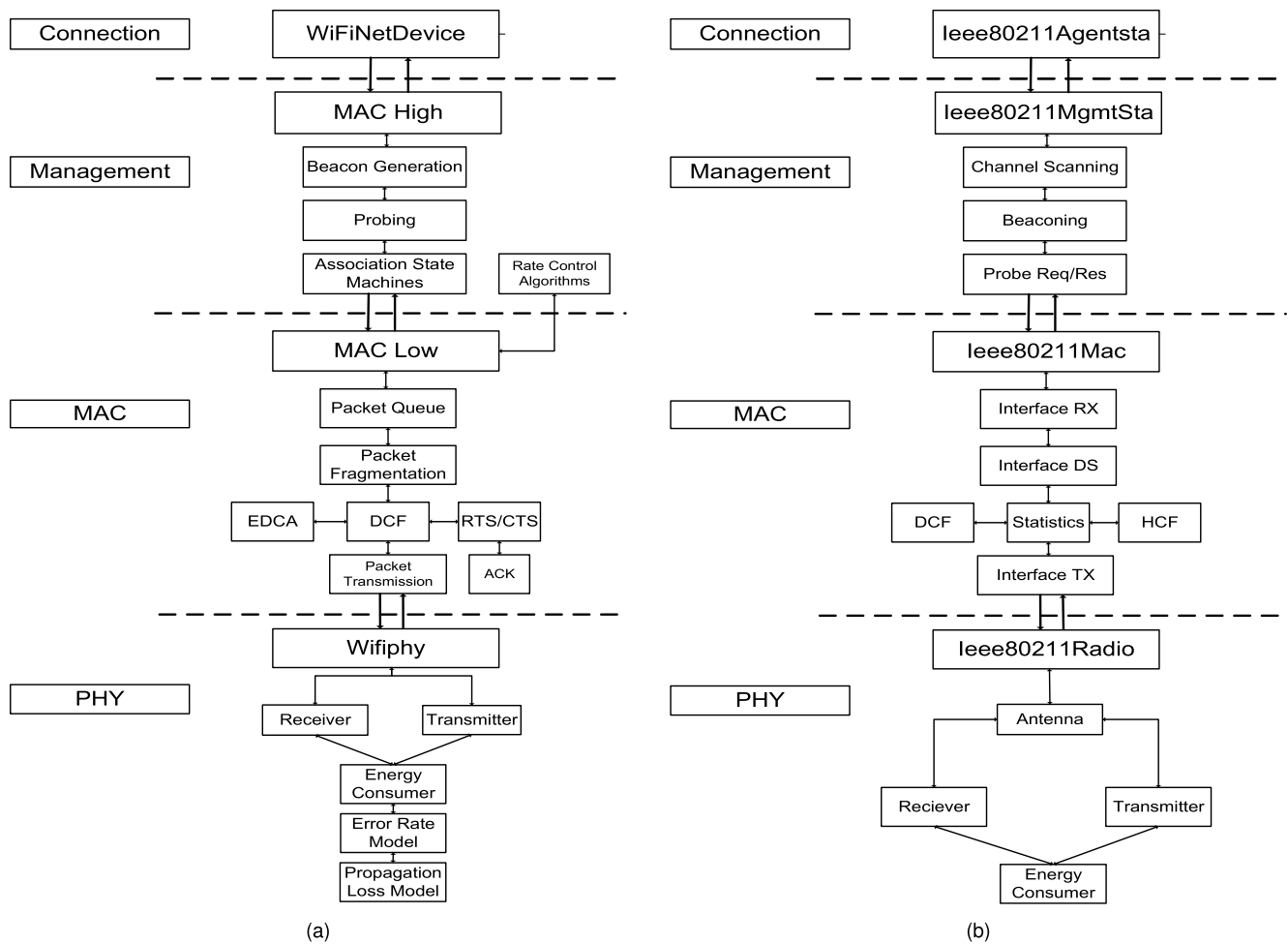
FIGURE 6. Layered representation of WiFi Modules (a) WiFiNetDevice architecture in NS-3. (b) Ieee80211 network interface card in OMNeT++.

$W = 16$. It is observed that the RT^* values are usually less than the standard setting value which is fixed at 2347 bytes. It is understood that for a broader range of packet lengths the RTS/CTS mechanism can lead to higher network throughput than the basic access.

VI. NS-3 AND OMNeT++ WiFi MODULES

This section introduces the WiFi modules in NS-3 and OMNeT++.

A. NS-3 WiFi MODULE

The WiFiNetDevice is used as a WiFi module in NS-3 [48] as depicted in Fig. 6a. The module models a network interface controller based on IEEE 802.11 standards and has four layers of submodules, where the top layer is the connection layer which acts as an interface to the management layer. When a transmission is initiated by an application the WiFiNetDevice acts as an interface and sends the packet to the MAC high module in management layer which handles many MAC functions such as beacon, association, probing and association state machines. The MAC high module is similar to

Ieee80211MgmtSta module in OMNeT++ with some extra features added such as a set of rate control algorithms which are called by the MAC low layer. There are a number of rate control algorithms available in NS-3. Real algorithms from real devices have helped creating these rate control algorithms. Some of the algorithms are ConstantRateWifiManager, ArfWifiManager, OnoeWifiManager and MinstrelWifiManager [48]. MAC high module is classified into four types such as

1) ns3:RegularWiFiMac: which uses QoS supported configuration and act as a common parent for the three below mentioned MAC models.
2) ns3:ApWiFiMac: used for an infrastructure mode with AP.
3) ns3:StaWifiMac: used for active probing and association states.
4) ns3:AdhocWiFiMac: used for ad-hoc mode.

The MAC layer has the MAC Low module which perform three duties 1) ns3:MacLow: which take care of transactions involving RTS/CTS and DATA/ACK. 2) ns3:DcfManager and ns3:DcfState: which manages the DCF functions.

3) ns3:DcaTxop and ns3:EdcaTxopN: which is used for handling packet queues, packet fragmentation, packet transmission/re-transmission. The PHY layer has the WiFi-Phy module which is used for the modeling of transmission and reception of frames, it is also responsible for tracking the energy consumption. The packet reception is related to three components which are,

1) Every received packet is checked to have a probability of success or failure. The probability depends on the state of the physical layer (while transmission or sleeping state no reception is made), signal to noise ratio and the type of modulation.

2) To compute the correct interference power, there is a track made for all the received signals which eases in reception decision.

3) Error models are used to compute the probability of the packets received successfully. The error rate model provides an additional feature as compared to OMNeT++ PHY layer, of calculating the probability of successful receptions. The calculation of interference power helps in reception decision. Error models confirms the probability of success and failure. PHY model also consults the propagation loss models to make sure a loss free transmission of packets.

B. OMNeT++ WiFi MODULE

OMNeT++ uses an INET framework for the TCP/IP and the protocols related to internet. The simulation model suite supports IPv4, IPv6, TCP, SCTP, UDP protocols, link layer models such as (PPP, ethernet, IEEE802.11), MPLS modeling with RSVP-TE and LDP signaling. The network interface card (NIC) used in OMNeT++ is configurable with several interfaces that are listed below:

1) Ieee80211Nic: a generic and configurable network interface card.

2) Ieee80211NicAdhoc: used for the ad-hoc mode.

3) Ieee80211NicAP: used for the access point functionalities.

4) Ieee80211NicSTA: used for the infrastructure mode with AP.

The IEEE 802.11 network interface card (NIC) is used as a WiFi Module in OMNeT++ [49] and has four submodules which are connection, management, MAC and PHY layer as shown in Fig. 6b. The connection layer issues instruction to management layer to perform channel scanning, beaconing and probe request/response. The results of these instructions are returned back to the Ieee80211AgentSta module. The dynamic behavior of the nodes can be changed by modifying or replacing the Ieee80211AgentSta module for example while implementing different handover strategies. The management layer having Ieee80211MgmtSta module is used to exchange management frames via MAC with entities such as nodes and APs. This layer is also responsible to switch channels periodically during scanning and gathers the information from the received probes and beacons. MAC layer has Ieee80211Mac submodule which is responsible to transmit and receive frames through interface TX and interface RX respectively, following the rules specified by CSMA/CA protocol. In OMNeT++ the MAC has built in policies such as ACK policy, RTS/CTS policy, TXOP policy, fragmentation policy, DCF policy, HCF policy and statistics policy. The physical layer has the Ieee80211Radio submodule which deals with the modeling of transmission and reception of frames and as there are no error rate model algorithms, so to determine whether the frames are received correctly, factors such as bit errors due to low signal powers or interference in radio channel are considered. Energy consumption is also modeled in this layer.

VII. EXPERIMENTAL SETUP

This section provides the detailed information of the simulation setup and the testbed setup.

A. SIMULATION SETUP

NS-3-dev (ns-3.26) and OMNeT++ 5.4.1 along with INET framework version of 4.1.0 is used for the simulations. Network modes such as infrastructure mode and ad-hoc modes are considered in this study so that a detailed comparison can be made between the mathematical model of IEEE 802.11 DCF networks and the simulation results. The infrastructure and ad-hoc mode are shown in Fig. 7. In the ad-hoc mode each nodes act as a receiver and a transmitter where as in infrastructure mode the nodes get the access to the network through an AP by following beacon transmission, active scanning and authentication.
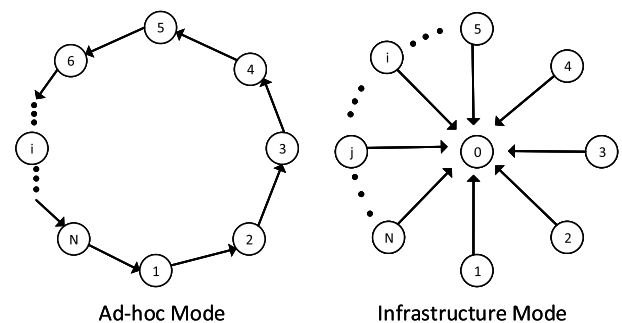

FIGURE 7. Ad-hoc mode and infrastructure mode.

The mathematical model focuses on the packet payload transmission ignoring the association effects of infrastructure mode, hence it is expected that the mathematical model will show closer results to the ad-hoc mode. In the ad-hoc mode each node sends and receives a packet at a data rate of 54 Mbps from the neighboring nodes and the number of nodes is increased from 5 to 50 with an increment of 5 nodes each time. The total throughput $\hat{D}$ is the sum of the data rate of each nodes.

In the simulation experiments we choose the AdhocWiFiMac and ApWiFiMac in NS-3 as mac layer for ad-hoc and infrastructure mode respectively. In OMNeT++ we use Ieee80211NicAdhoc and Ieee80211NicSTA as the mac layer for ad-hoc and infrastructure mode respectively. The physical

layer and the default WiFi channel are utilized from YANS model [50]. IEEE 802.11 ''a'' standard is considered in the simulation setup with the basic data rates of 6 to 9, 12, 18, 24, 36, 48 and 54 Mbps. The nodes are separated by a minimum distance of 0.001 *m*. The time for data packet generation is set to 200 μs. The system parameters are summarized in Table 5. Finally, all our experiments are performed on a HP PC, a core i7 2.4 GHz CPU and 16 GB of RAM.

TABLE 5. System Parameter Settings [5].

PHY Header (PH)	20 μs	ACKTimeout	69 μs
MAC header (MH)	36 bytes	CTSTimeout	69 μs
ACK	14 bytes	DIFS	34 μs
RTS	20 bytes	SIFS	16 μs
CTS	14 bytes	Slot Time	9 μs
CW_{min}	15	CW_{max}	1023

B. TESTBED SETUP

A real world testbed is constructed to perform identical experiments such as in NS-3 and OMNeT++, with multiple nodes and one AP as shown in Fig. 8. The testbed validates the Dai's model [1] and the simulation results for the infrastructure mode, where the throughput is measured against the initial backoff window size W for different number of nodes n.
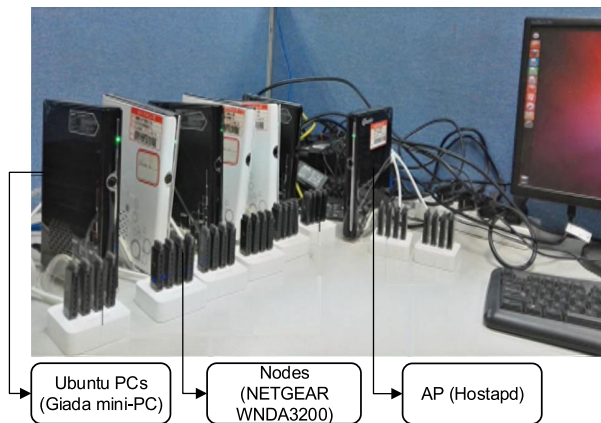

FIGURE 8. Testbed setup.

The testbed built is Linux-3.2.71 kernel based with mac80211 driver support. NETGEAR WNDA3200 USB wireless cards based on the Atheros AR9002U-2NX chipsets, are used as wireless nodes. AR9002U-2NX chipsets use the Ath9k drivers. The iw [51] configure tool is used to communicate to mac80211 module. The protocol drift between the iw and the Ath9k driver is depicted in Fig. 9. The configuration of APIs for the IEEE 802.11 network devices is provided by the cfg80211 module. The nl80211 module acts as an interface bridge between iw and cfg80211 module. The configuration details of cfg80211 is provided by nl80211. The iw source file nl80211.h, contains the *nl*80211_*commands*
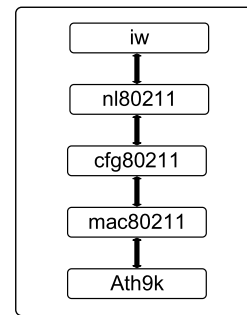

FIGURE 9. The protocol drift between Ath9k driver and iw.

among which the utility $NL80211\_CMD\_SET\_WIPHY$, allows to the change the DCF parameters. The iw employs a *nl*80211_*txq_attrstruct* where the window sizes are defined. The window sizes are changed by adding an extension to the iw tool. This extension *handle_txq*() is appended to the source file *phy.c* to modify *nl*80211_*txq_attr*.

A WiFi network with no encryption is created with the help of hostapd [52]. Hostapd.conf file in the AP serve as a platform to configure the network configuration such as channel parameters and SSID. Ubuntu 14.04 using a USB hub allows multiple USB wireless cards to act as a node, these wireless cards are configured with unique IP addresses, which serve as multiple WiFi nodes in the testbed network. To connect to the network, the nodes use the extended tool *iw* created by the hostapd. These distributed nodes are controlled efficiently by a controlling host in the testbed with the help of a tool called ClusterSSH. The nodes are considered as stationary so the transmission rate and RSSI values are constant for each node. Throughput is measured using iPerf. The testbed topology is shown in Fig. 10 where the wireless stations are connected to the AP. The testbed is developed as a test run to measure the throughput against the initial backoff window size W, for a large number of nodes, by using the command *iwdevwlanXsettxqW*.
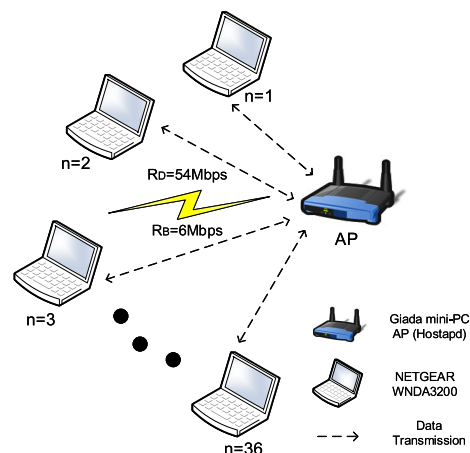

FIGURE 10. Testbed topology.

VIII. PERFORMANCE EVALUATION

This section presents the simulation and testbed results for DCF experiments, RTS threshold, and performance comparison of NS-3 and OMNeT++ in terms of computational times, memory usage and CPU utilization.

A. DCF EXPERIMENTS

In this section we report the list of DCF experiments performed on both the simulators and the testbed to see the comparison between Dai's theoretical model and the simulation results achieved by WiFi modules of both the simulators and the testbed. Results are obtained by varying the aggregate traffic input rate $\hat{\lambda}$, number of nodes n, window size W and the cutoff phase K. Multiple simulation runs are made in each simulator and the mean output value is considered for performance evaluation.

1) NETWORK PERFORMANCE AGAINST TRAFFIC: UNSATURATED TO SATURATED

In both the simulators the network state is shifted from unsaturated to saturated by increasing the overall input rate $\hat{\lambda} = n\lambda$ where λ is the probability of generating a new packet every τ_T time slots.
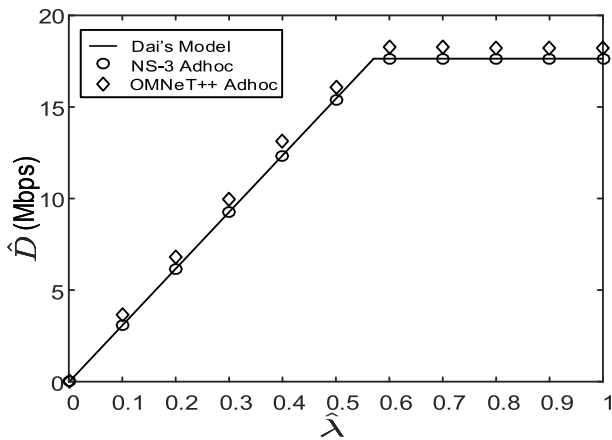
FIGURE 11. Throughput $\hat{D}$ against the aggregate input rate $\hat{\lambda}$ in IEEE 802.11 DCF network with basic access mechanism. *PL* = 1023 bytes. *n* = 50. *W* = 16. *K* = 6. R_D = 54 Mbps. R_B = 6 Mbps.

Increasing the input rate $\hat{\lambda}$ gradually increases the total throughput $\hat{D}$ and eventually the network becomes saturated as shown in Fig. 11. NS-3 exhibits a slightly close behavior to the mathematical model as compared to OMNeT++ simulator. When each node has a low λ value the network is unsaturated, which means that each HOL packet can be transmitted successfully. As $\hat{\lambda}$ increases the network becomes saturated as each node still has a packet to transmit, at this point the $\hat{D}$ remains constant as $\hat{\lambda}$ increases and is decided by system backoff parameters.

2) THROUGHPUT AGAINST KEY SYSTEM PARAMETERS

By looking at Eq. 3 and 5 the total throughput $\hat{D}$ of a saturated IEEE 802.11 DCF network depends upon the cutoff phase K, the initial backoff window size W and number of contending nodes n.

The first comparison for the throughput $\hat{D}$ is made against the number of nodes n as shown in Fig. 12. The simulation results show a close behavior to the theoretical results except when the value of n is equal or less than 5. The reason for this is that $\hat{D}$ in the mathematical model is determined by the successful transmission of HOL packets p, where p is obtained under an assumption that n is sufficiently large. Therefore the theoretical network throughput rate may deviate from the simulation results when $n < 5$.
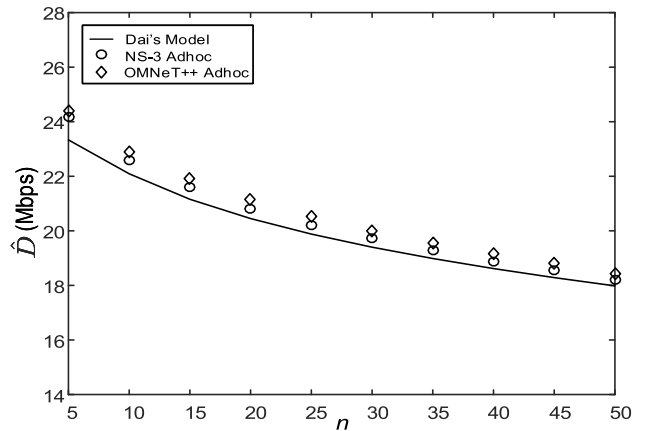
FIGURE 12. Throughput $\hat{D}$ against the number of nodes n in a saturated IEEE 802.11 DCF network with basic access mechanism. *PL* = 1023 bytes. *W* = 16. *K* = 6. R_D = 54 Mbps. R_B = 6 Mbps.

The next comparison made is between the throughput $\hat{D}$ and the initial backoff window size W in an infrastructure mode as shown in Fig. 13a. This experiment is done twice by changing the number of nodes n. In the first experiment n is kept as 9 and the normalized throughput is taken as the performance metric shown in red color. The normalized throughput is calculated by the mathematical formula given as $\frac{X-\mu}{\alpha}$, where X is the data set, μ is the mean value and α is the standard deviation. By observing the curves for the normalized throughput, we see a similar trend in throughput variation for the Dai's model, simulators and the testbed. For $n = 9$ and $W = 32$ a maximum normalized throughput is obtained for the testbed, simulators and the model.

Due to the complexity of making a testbed a very few studies have reported the prototype testbed implementation with number of nodes more than 15 in the literature. We created a low-cost testbed by utilizing USB WiFi cards that helped us in creating a large number of nodes such as 36.

In the second experiment, the nodes are increased from 9 to 36. We compare the theoretical and simulation results to the testbed results and notice a sufficient performance gap. The performance gap is a result of an uncontrolled environment for testbed where the physical medium has interference from external sources and the distance between nodes and AP cause fading. NS-3 simulation results however show more accuracy in comparison to OMNeT++ and testbed results when compared to theoretical results. In the start when W increases the nodes attain a larger window size hence
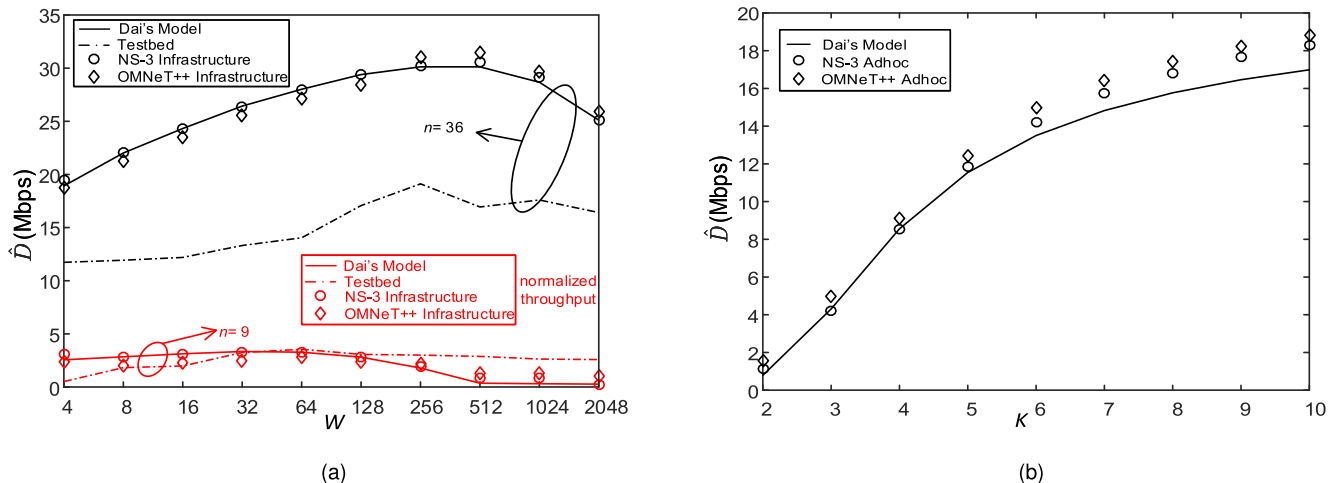
FIGURE 13. Throughput $\hat{D}$ against the backoff parameters. (a) $\hat{D}$ against the initial backoff window size W in IEEE 802.11 DCF network with basic access mechanism in infrastructure mode. n = 9 and 36. PL = 1023. R_D = 54 Mbps. R_B = 6 Mbps. (b) $\hat{D}$ against the cutoff phase K in a saturated IEEE 802.11 DCF network with basic access mechanism. n = 50. PL = 1023. R_D = 54 Mbps. R_B = 6 Mbps.

avoiding collisions resulting in higher throughputs. When W continue to increase the nodes backoff duration increase keeping the channel idle for longer time hence decreasing the throughput. We hope to validate Dai's model [1] and the simulation results, considering the remaining key system parameters through a testbed in the near future and have a deeper investigation about the factors impacting the testbed performance.

As shown in Fig. 13b the increase in the cutoff phase K, increases the throughput $\hat{D}$ monotonically. The simulator results exhibit a close behavior to that of the theoretical analysis. When K increases the maximum backoff window size increases allowing the nodes to choose different window sizes hence avoiding collisions resulting in higher throughput due to less collisions.

3) BASIC VS. RTS/CTS ACCESS MODES
The DCF protocol comes with two access modes, one is the basic access mechanism and other is the optional RTS/CTS mechanism as explained in detail earlier in section V-A. Here we will shed some light on the basic differences of two access modes and evaluate the network performance in each of them. Basic access is a two-way handshake where the node transmits the packets after waiting a DIFS duration if it senses the channel idle, otherwise it starts the backoff process by choosing the backoff window size. If the node receives the ACK frame it means the packet is transmitted successfully, if no ACK frame is received within the ACK time-out period then the backoff process is started by node.

RTS/CTS is a four way handshake where the node first sends the RTS frame to reserve the channel. After the RTS frame is successfully received by the destination, it will send a CTS frame to all nodes so that the other nodes pause their transmission until the current transmission is over. The packet transmission then takes places which is confirmed by the

ACK frame otherwise the nodes starts the backoff process after the CTS time-out period.

The variation of the throughput $\hat{D}$ with the packet loads PL is depicted for both modes in Fig. 14. The data rate is set to 54 Mbps and 24 Mbps. By looking at Fig. 14 we observe that the throughput performance for data rate R_D = 54 Mbps is superior than the data rate R_D = 24 Mbps for higher values of packet payloads. The NS-3 simulation results closely follow the mathematical analysis trend which is a good indication that Dai's model can be served as a trustworthy model to validate the WiFi MAC layer in both NS-3 and OMNeT++.
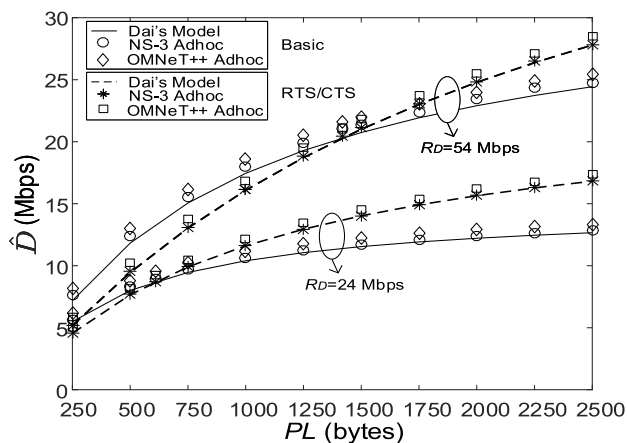

FIGURE 14. Throughput $\hat{D}$ against the packet payload PL in a saturated IEEE 802.11 DCF network with basic access and RTS/CTS. n = 50. W = 16. K = 6. R_B = 6 Mbps.

4) AD-HOC VS INFRASTRUCTURE
An ad-hoc network consists of devices that communicate to each other directly with a decentralized architecture. Each node is a transceiver, as it receives the packets and also transmits them to receiving nodes. In an infrastructure mode,

the nodes communicate with each other by first connecting to an AP making the network architecture centralized. As opposed to the ad-hoc mode the infrastructure mode has an additional channel activity because of the association which comprises of active scanning, beacon transmission etc.

Fig. 15 depicts the effect of association in infrastructure mode on the overall throughput performance. In the comparison made between the throughput $\hat{D}$ and number of nodes n for ad-hoc and infrastructure mode, we can see that the overall throughput achieved for infrastructure mode is slightly less than that of an ad-hoc mode due to the supplementary association activity in the infrastructure mode. The difference can be observed in both the simulator results, where NS-3 exhibit simulation results, that are slightly closer to the mathematical model.
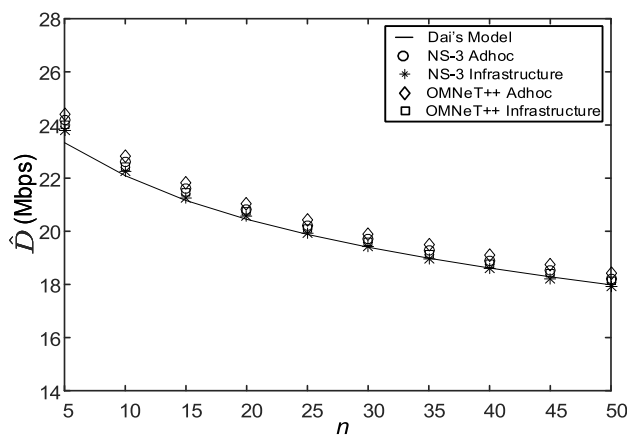

FIGURE 15. Throughput $\hat{D}$ against the number of nodes n in a saturated IEEE 802.11 DCF network with basic access in ad-hoc and infrastructure mode. $PL = 1023$ bytes. $n = 50$. $W = 16$. $K = 6$. $R_D = 54$ Mbps.

B. RTS THRESHOLD PERFORMANCE

In this section we optimize the explicit expression of RTS threshold RT^* obtained in Eq.11 by using NS-3 and OMNeT++. The analysis done for RT^* in section V-B is verified and the performance comparison of RT^* is made for standard vs the optimal setting.

1) NETWORK THROUGHPUT

The variation of the throughput against the packet length is shown in Fig.16. It can be deduced that when the data rate is higher, i.e., 54 Mbps as compared to the data rate of 24 Mbps, the RTS threshold RT^* is large. The previous analysis done in section V-B for RT^* against data rate and basic rate is verified, which confirms that as the data rates increase the RT^* also increase and vice versa for R_B. In case of basic rate as shown in Fig. 16b, we can observe that for the basic rate of 6 Mbps the RT^* is higher as compared to basic date rates of 54 Mbps. This confirms the previous analysis that when the R_B increases the RT^* should decrease.

Furthermore, by looking at Fig. 16, it is observed that when the PL is smaller than the RT^*, the throughput achieved by

RTS/CTS mechanism is lower than that of the basic access. The throughput eventually increases when PL gets higher than RT^*, such that $PL > RT^*$. In Fig. 16a with a basic rate $R_B = 6$ Mbps, the RT^* for $R_D = 54$ Mbps and 24 Mbps can be obtained from Eq. 11 as 1419 and 611 respectively, which are certified by simulation results shown in Fig. 16. The consistency of the analysis and the simulation results guarantee an accurate switching between RTS/CTS mechanism and basic access in order to achieve the optimal throughput.

The variation of the throughput $\hat{D}$ in an IEEE 802.11 WLAN with RTS/CTS and basic access mode is shown in Fig. 17 where the packet length PL is equal to RT^*. It can be concluded that when the $PL = RT^*$, no matter which access mode, either the basic access or RTS/CTS the network utilizes, the network throughput $\hat{D}$ remains identical regardless of the variation in data rates R_D or network size n as seen in Fig.17a and Fig. 17b. A good match can be seen between the analysis and the simulators verifying the accuracy of the analysis made on optimal RTS threshold RT^*, where NS-3 shows more accuracy towards the mathematical model.

2) OPTIMAL SETTING VS STANDARD SETTING

We compare the gain in the throughput of the IEEE 802.11 WLAN in optimal setting to that of the standard setting for RTS threshold RT^*. In the standard setting the RT^* is fixed at 2347 bytes which means that the network will only operate in the RTS/CTS mechanism when PL exceeds 2347 bytes. For $n = 20$ as shown in Fig. 18, when packet length $PL > RT^* = 2091$ bytes, the network switches to RTS/CTS mechanism with optimal setting. The optimal setting guarantees higher throughput rate in the RTS/CTS access mode. Fig. 18a shows that when the packet length PL is in the range between $2091 < PL < 2347$, the network achieves higher throughput with optimal setting. The gain in the throughput is further enhanced when the size of the network grows such as $n = 50$. Fig. 18b shows that when n grows to 50, a higher throughput is achieved in the network with the optimal setting of RTS threshold RT^* as compared to the standard setting for a wider range of PL, i.e., $1419 < PL < 2347$ bytes. Hence it is deduced that the adaptive optimal tuning of the RTS threshold in a large size IEEE 802.11 WLAN can lead to considerable gain in throughput performance. The performance of the NS-3 in the optimal tuning of RTS for $n = 20$ and $n = 50$ is more close to the mathematical model making it more suitable for the large scale simulations and optimization techniques.

C. PERFORMANCE COMPARISON OF NS-3 AND OMNeT++

This section focuses on an important question for validation process: "Have you chosen the right environment for your simulation model"? To answer this question we compared the performance of both the simulators for different number of nodes during the validation of IEEE 802.11 DCF network,
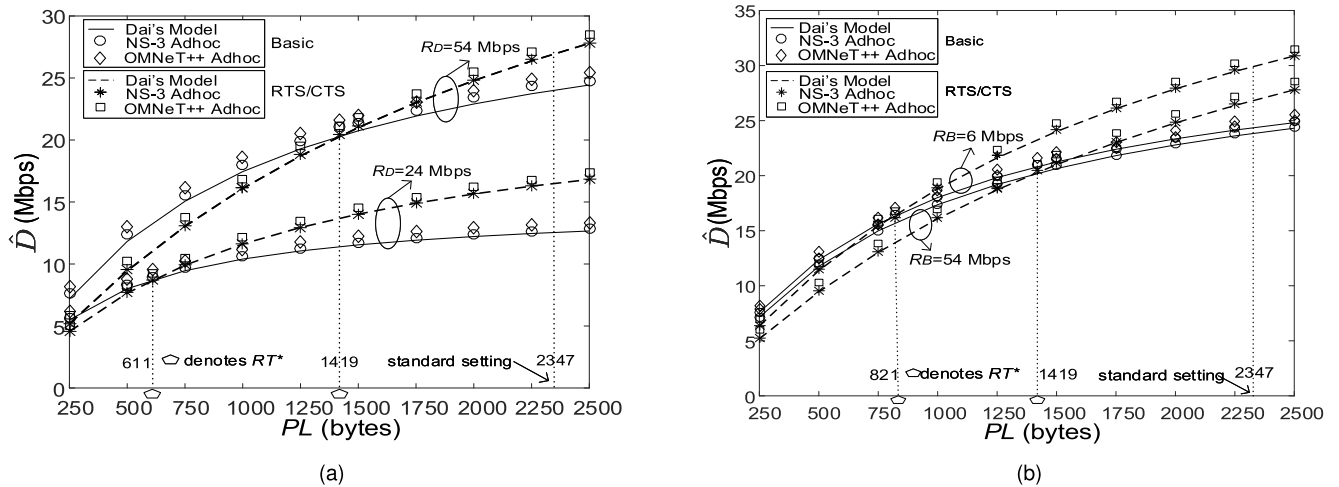
FIGURE 16. Throughput $\hat{D}$ against the packet payload *PL* in a saturated IEEE 802.11 DCF network with basic access and RTS/CTS mechanism. $n = 50$. $W = 16$. $K = 6$. (a) $R_B = 6$ Mbps. (b) $R_D = 54$ Mbps.
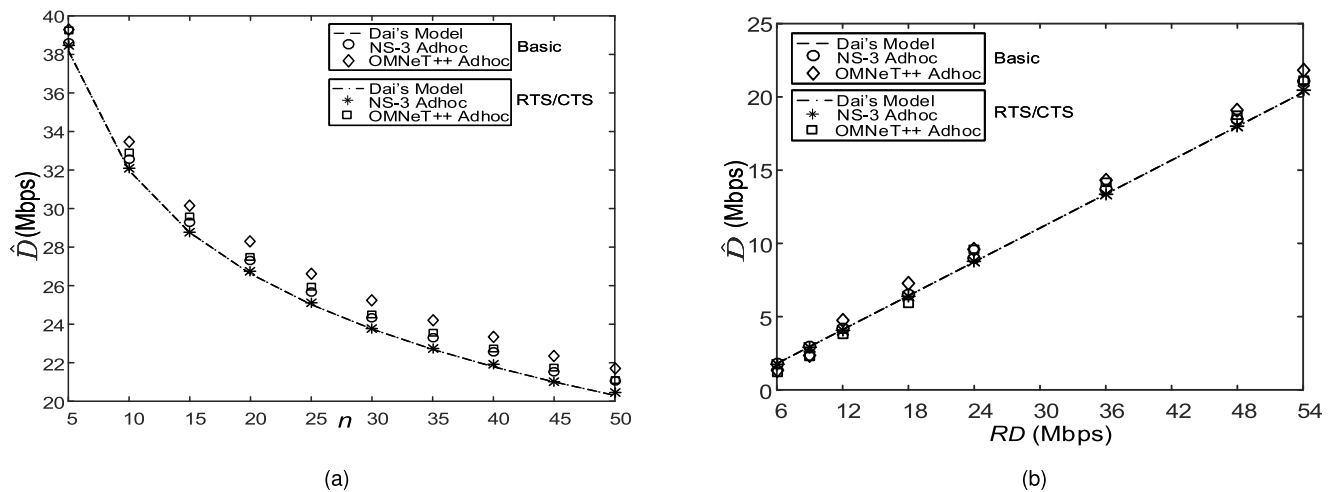

FIGURE 17. Throughput $\hat{D}$ against the number of nodes n and data rate R_D in a saturated IEEE 802.11 DCF network with basic access mechanism and RTS/CTS mechanism. $W = 16$. $K = 6$. $R_B = 6$ Mbps. $PL = RT^*$. (a) Throughput $\hat{D}$ against the number of nodes n. $R_D = 54$ Mbps. (b) Throughput $\hat{D}$ against the data rate R_D. $n = 50$.

TABLE 6. Performance comparison of NS-3 and OMNeT++.

Number of Nodes	5		15		30		45		50	
Simulator Comparison	NS-3	OMNeT++	NS-3	OMNeT++	NS-3	OMNeT++	NS-3	OMNeT++	NS-3	OMNeT++
Computational time (min)	5	7	14	18	22	34	39	45	45	60
Memory usage (%)	6	8	15	18	21	25	31	38	39	45
CPU utilization (%)	8	20	12	24	15	30	18	40	20	60

in terms of computational times, memory usage and CPU utilization as shown in Table 6. Computational time or running time is the time consumed by each of the simulator to show the output, the memory usage is the percentage of the random access memory utilized while performing simulation runs and CPU utilization corresponds to the work handled by the CPU during the simulations.

Htop, a text mode application and process reviewer, is utilized to measure the CPU and memory performance for NS-3 and OMNeT++. When the number of nodes are increased from 5 to 50, OMNeT++ shows the highest computational time, making it is less scalable as compared to NS-3. A linear increase in the memory usage of both the simulators is observed while increasing the number of nodes,
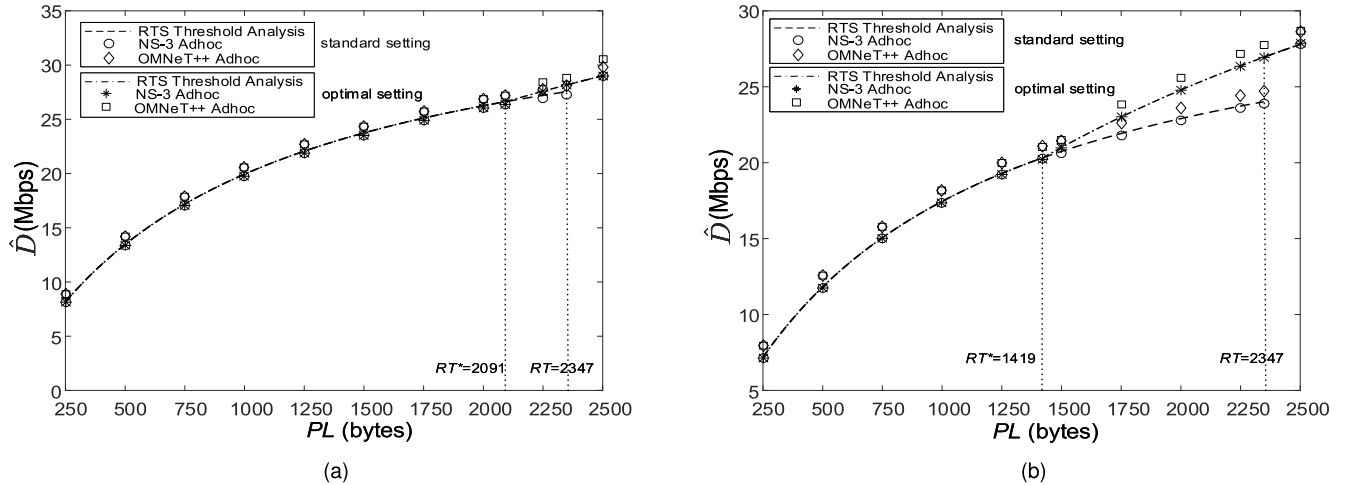
FIGURE 18. Throughput $\hat{D}$ against the packet payload *PL* in a saturated IEEE 802.11 DCF network with basic access mechanism and RTS/CTS mechanism. $W = 16$. $K = 6$. $R_D = 54$ Mbps. $R_B = 6$ Mbps. Optimal setting: $RT = RT^*$. Standard setting: $RT = 2347$ bytes. (a) $n = 20$. (b) $n = 50$.

TABLE 7. Architectural efficiency of NS-3 and OMNeT++.

Quality Factors	NS-3	OMNeT++
Debugging support	Good	Good
Compatibility support	Good	Excellent
Integration support	Excellent	Excellent
Flexibility support	Excellent	Excellent
Modular complexity	Moderate	Low

where NS-3 performed efficiently. NS-3 shows a good CPU utilization response as the number of nodes increase. The CPU utilization response is analyzed while other applications ran in parallel, as the simulations took longer time to execute. Despite of being new and under development, NS-3 outperformed OMNeT++ by showing a quick computational response, less memory usage and a way less CPU utilization for the same number of nodes used in OMNeT++.

While performing the IEEE 802.11 DCF simulations and optimal RTS threshold optimization, we rated the architectural efficiency of both the simulators by introducing some quality factors as shown in Table 7. These quality factors are defined as follows:

- Debugging support: defines the ease by which the simulator allowed us to find and demonstrate bugs.
- Compatibility support: defines the degree of extent to which an old application can be reused in a new application.
- Integration support: refers to the ability to make individual developed components to work together without any errors.
- Flexibility support: defines the degree of adaptability by which the simulator can support the new possible extensions.

- Modular complexity: defines the difficulties involved between the interaction of modular components in the simulator.

OMNeT++ uses a well-established modular architecture with different user interfaces such as *CMDENV*, *TKENV* and *TVENV*, which reduces its modular complexity. Separate library files such as *libsim_std.a*, *libenvir.a*, etc., help in the segregation of components allowing them to be called from a separate source directory. This enhances the compatibility support in OMNeT++. In general both the simulators are flexible to handle, allowing the user to make customizations as per their requirements.

IX. CONCLUSION AND FUTURE WORK

In this article, we have performed a comparative simulation study of the unified IEEE 802.11 DCF analytical model [1] against the IEEE 802.11 MAC simulation model in NS-3 and OMNeT++ and addressed an open issue of how to achieve the optimal throughput performance in IEEE 802.11 DCF networks by accurately switching between the basic access and RTS/CTS mechanism. A Linux based testbed is also setup to see the performance of window sizes against the throughput for different number of nodes, which validates the analysis and simulation results. The optimization analysis of RTS threshold show that with the standard setting it is difficult for the network to activate the RTS/CTS mechanism which incurs rate loss. An explicit expression of RTS threshold is achieved in the study, which is adaptively tuned in NS-3 and OMNeT++ against the basic rate and data rate, number of nodes and backoff parameters such as cutoff phase and window size. Considering the implementation regarding the practical design, the AP can easily calculate the RTS threshold and decides the most suitable access mode for the network.

The study reveals that 1) NS-3 shows close trends to the theoretical analysis with improved computational times,

memory usage and CPU utilization rates as compared to OMNeT++; 2) Dai's model is a simple but a powerful tool for the performance evaluation of homogeneous IEEE 802.11 networks; 3) the RTS threshold analysis and its optimal tuning in NS-3 and OMNeT++ focus on an important aspect of network design for IEEE 802.11 DCF networks. We hope to validate the remaining simulation experiments apart from throughput vs window sizes for multiple nodes with recent and commercial IEEE 802.11 standards, through the testbed in near future.

REFERENCES

[1] L. Dai and X. Sun, "A unified analysis of IEEE 802.11 DCF networks: Stability, throughput, and delay," *IEEE Trans. Mobile Comput.*, vol. 12, no. 8, pp. 1558–1572, Aug. 2013.

[2] J. Lei, J. Tao, J. Huang, and Y. Xia, "A differentiated reservation MAC protocol for achieving fairness and efficiency in multi-rate IEEE 802.11 WLANs," *IEEE Access*, vol. 7, pp. 12133–12145, 2019.

[3] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 3, pp. 535–547, Mar. 2000.

[4] *Ns-2*. [Online]. Available: https://www.isi.edu/nsnam/ns/

[5] *IEEE Standard for Information Technology—Telecommunications and Information Exchange Between Systems—Local and Metropolitan Area Networks—Specific Requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, Standard 802.11-2016, Jun. 2007.

[6] S. Yan, Y. Zhuo, and S. Wu, "An adaptive RTS threshold adjust algorithm based on minimum energy consumption in IEEE 802.11 DCF," in *Proc. Int. Conf. Commun. Technol. ICCT*, Apr. 2003, pp. 1210–1214.

[7] A. C. B. P. Chatzimisios, "Optimisation of RTS/CTS handshake in IEEE 802.11 wireless LANs for maximum performance," in *Proc. IEEE Global Telecommun. Conf. Workshops, GlobeCom Workshops*, Nov. 2004, pp. 270–275.

[8] Z.-n. Kong, D. H. K. Tsang, and B. Bensaou, "Adaptive RTS/CTS mechanism for IEEE 802.11 WLANs to achieve optimal performance," in *Proc. IEEE Int. Conf. Commun.*, Jun. 2004, pp. 185–190.

[9] L. Zhang and Y. Shu, "RTS threshold self-tuning algorithm based on delay analysis on 802.11 DCF," in *Proc. Int. Conf. Wireless Commun., Netw. Mobile Comput.*, Sep. 2005, pp. 627–630.

[10] J. Liu, W. Guo, B. Xiao, and F. Huang, "RTS threshold adjustement algorithm for IEEE 802.11 DCF," in *Proc. 6th Int. Conf. ITS Telecommun.*, Jun. 2006, pp. 654–658.

[11] M. Ibrahim, M. A. Iqbal, M. Aleem, and M. A. Islam, "SIM-cumulus: An academic cloud for the provisioning of network-simulation-as-a-service (NSaaS)," *IEEE Access*, vol. 6, pp. 27313–27323, 2018.

[12] A. Akram and L. Sawalha, "A survey of computer architecture simulation techniques and tools," *IEEE Access*, vol. 7, pp. 78120–78145, 2019.

[13] I. Lera, C. Guerrero, and C. Juiz, "YAFS: A simulator for IoT scenarios in fog computing," *IEEE Access*, vol. 7, pp. 91745–91758, 2019.

[14] S. Manzoor, Z. Chen, Y. Gao, X. Hei, and W. Cheng, "Towards QoS-aware load balancing for high density software defined Wi-Fi networks," *IEEE Access*, vol. 8, pp. 117623–117638, 2020.

[15] *Ns-3*. [Online]. Available: https://www.nsnam.org/

[16] *OMNeT++*. [Online]. Available: https://omnetpp.org/

[17] *PlanetLab*. [Online]. Available: https://www.planet-lab.org/

[18] P. Barsocchi, G. Oligeri, and F. Potorti, "Validation for 802.11b wireless channel measurements," Consiglio Nazionale delle Ricerche, Rome, Italy, Tech. Rep., 2006.

[19] J. Liu, Y. Yuan, D. M. Nicol, R. S. Gray, C. C. Newport, D. Kotz, and L. F. Perrone, "Empirical validation of wireless models in simulations of ad hoc routing protocols," *Simulation*, vol. 81, no. 4, pp. 307–323, Apr. 2005.

[20] P. Fuxjager and F. Ricciato, "Collecting broken frames: Error statistics in IEEE 802.11b/g links," in *Proc. 6th Int. Symp. Modeling Optim.*, Apr. 2008, pp. 30–35.

[21] A. Di Stefano, A. Scaglione, G. Terrazzino, I. Tinnirello, V. Ammirata, L. Scalia, G. Bianchi, and C. Giaconia, "On the fidelity of IEEE 802.11 commercial cards," in *Proc. 1st Int. Conf. Wireless Internet (WICON)*, Jul. 2005, pp. 10–17.

[22] M. Bredel and M. Bergner, "On the accuracy of IEEE 802.11g wireless LAN simulations using OMNeT++," in *Proc. 2nd Int. ICST Conf. Simulation Tools Techn.*, Mar. 2009, p. 81.

[23] F. J. Estevez, J. M. Garcia, J. M. Castillo-Secilla, and J. Gonzalez, "Enabling validation of IEEE 802.15.4 performance through a new dual-radio Omnet++ model," *Elektronika Ir Elektrotechnika*, vol. 22, no. 3, pp. 89–94, Jun. 2016.

[24] M. Karaca, Z. Zhang, and B. Landfeldt, "An opportunistic backoff scheme for dense IEEE 802.11 WLANs," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Dec. 2015, pp. 1–6.

[25] M. Scarpa and S. Serrano, "A new modelling approach to represent the DCF mechanism of the CSMA/CA protocol," in *Proc. Int. Conf. Anal. Stochastic Modeling Techn. Appl.*, Springer, 2017, pp. 181–195.

[26] T. Qayyum, A. W. Malik, M. A. Khan Khattak, O. Khalid, and S. U. Khan, "FogNetSim++: A toolkit for modeling and simulation of distributed fog environment," *IEEE Access*, vol. 6, pp. 63570–63583, 2018.

[27] N. I. Sarkar and R. Membarth, "Modeling and simulation of IEEE 802.11 g using OMNeT++," in *Handbook of Research on Discrete Event Simulation Environments: Technologies and Applications*. Hershey, PA, USA: IGI Global, 2010, pp. 379–397.

[28] G. Pei and T. Henderson. (2009). *Validation of Ns-3 802.11b PHY Model*. [Online]. Available: https://www.nsnam.org/workshops/wns3-2009/talks/pei-80211phy.pdf

[29] N. Baldo, M. Requena-Esteso, J. Núñez-Martínez, M. Portolès-Comeras, J. Nin-Guerrero, P. Dini, and J. Mangues-Bafalluy, "Validation of the IEEE 802.11 MAC model in the ns3 simulator using the EXTREME testbed," in *Proc. 3rd Int. ICST Conf. Simul. Tools Techn.*, 2010, p. 64.

[30] M. Malekzadeh, A. A. A. Ghani, S. Subramaniam, and J. M. Desa, "Validating reliability of OMNeT++ in wireless networks DoS attacks: Simulation vs. testbed," *IJ Netw. Secur.*, vol. 13, no. 1, pp. 13–21, 2011.

[31] R. Patidar, S. Roy, T. R. Henderson, and M. Mehrnoush, "Validation of Wi-Fi network simulation on Ns-3," University of Washington, Seattle, WA, USA, Tech. Rep., 2017.

[32] Z. Chen, D. Fu, Y. Gao, and X. Hei, "Performance evaluation for WiFi DCF networks from theory to testbed," in *Proc. IEEE Int. Symp. Parallel Distrib. Process. with Appl. IEEE Int. Conf. Ubiquitous Comput. Commun. (ISPA/IUCC)*, Dec. 2017, pp. 1364–1371.

[33] T. Zahid, X. Hei, W. Cheng, A. Ahmad, and P. Maruf, "On the tradeoff between performance and programmability for software defined WiFi networks," *Wireless Commun. Mobile Comput.*, vol. 2018, pp. 1–12, Apr. 2018.

[34] A. Bazzi, C. Campolo, B. M. Masini, A. Molinaro, A. Zanella, and A. O. Berthet, "Enhancing cooperative driving in IEEE 802.11 vehicular networks through full-duplex radios," *IEEE Trans. Wireless Commun.*, vol. 17, no. 4, pp. 2402–2416, Apr. 2018.

[35] A. Bazzi, G. Cecchini, M. Menarini, B. M. Masini, and A. Zanella, "Survey and perspectives of vehicular Wi-Fi versus sidelink cellular-V2X in the 5G era," *Future Internet*, vol. 11, no. 6, p. 122, May 2019.

[36] J. Wang and B. Li, "The impact of RTS threshold on the performance of multi-rate network," in *Proc. IET Int. Commun. Conf. Wireless Mobile Comput. (CCWMC)*, Dec. 2009, pp. 651–656.

[37] S. M. Rifat Ahsan, M. Saiful Islam, N. Hassan, and A. Rahman, "Packet distribution based tuning of RTS threshold in IEEE 802.11," in *Proc. IEEE Symp. Comput. Commun.*, Jun. 2010, pp. 1–6.

[38] H. Singh, A. Kaur, A. Sharma, and V. Sharma, "Performance optimization of DCF-MAC standard using enhanced RTS threshold under impact of IEEE 802.11n WLAN," in *Proc. 5th Int. Conf. Adv. Comput. Commun. Technol.*, Feb. 2015, pp. 421–424.

[39] T. Sugimoto, N. Komuro, H. Sekiya, S. Sakata, and K. Yagyu, "Maximum throughput analysis for RTS/CTS-used IEEE 802.11 DCF in wireless multi-hop networks," in *Proc. Int. Conf. Comput. Commun. Eng. (ICCCE)*, May 2010, pp. 1–6.

[40] M. Mjidi, D. Chakraborty, N. Nakamura, K. Koide, A. Takeda, and N. Shiratori, "A new dynamic scheme for efficient RTS threshold handling in wireless networks," in *Proc. 22nd Int. Conf. Adv. Inf. Netw. Appl. (AINA)*, Mar. 2008, pp. 734–740.

[41] S.-T. Sheu, T. Chen, J. Chen, and F. Ye, "The impact of RTS threshold on IEEE 802.11 MAC protocol," in *Proc. 9th Int. Conf. Parallel Distrib. Syst.*, Dec. 2002, pp. 267–272.

[42] Q. Nasir and M. Albalt, "History based adaptive backoff (HBAB) IEEE 802.11 MAC protocol," in *Proc. 6th Annu. Commun. Netw. Services Res. Conf. (CNSR)*, May 2008, pp. 533–538.

[43] Y. Gao, X. Sun, and L. Dai, "Throughput optimization of heterogeneous IEEE 802.11 DCF networks," *IEEE Trans. Wireless Commun.*, vol. 12, no. 1, pp. 398–411, Jan. 2013.

[44] Y. Gao, L. Dai, and X. Hei, "Throughput optimization of multi-BSS IEEE 802.11 networks with universal frequency reuse," *IEEE Trans. Commun.*, vol. 65, no. 8, pp. 3399–3414, Aug. 2017.

[45] Y. Gao, X. Sun, and L. Dai, "IEEE 802.11e EDCA networks: Modeling, differentiation and optimization," *IEEE Trans. Wireless Commun.*, vol. 13, no. 7, pp. 3863–3879, Jul. 2014.

[46] Y. Gao, X. Sun, and L. Dai, "Achieving optimum network throughput and service differentiation for IEEE 802.11e EDCA networks," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2013, pp. 362–367.

[47] Y. Gao and L. Dai, "Optimal Downlink/Uplink throughput allocation for IEEE 802.11 DCF networks," *IEEE Wireless Commun. Lett.*, vol. 2, no. 6, pp. 627–630, Dec. 2013.

[48] *Ns-3WiFi*. [Online]. Available: https://www.nsnam.org/docs/models/html/wifi.html

[49] *OMNeT++WiFi*. [Online]. Available: https://inet.omnetpp.org/docs/users-guide/

[50] M. Lacage and T. R. Henderson, "Yet another network simulator," in *Proc. Workshop NS-2: IP Netw. Simulator WNS*, 2006, p. 12.

[51] *Iw*. [Online]. Available: https://wireless.wiki.kernel.org/en/users/documentation

[52] *Hostapd*. [Online]. Available: http://w1.fi/hostapd/

SOHAIB MANZOOR received the B.S. degree in electrical engineering from the Mirpur University of Science and Technology, Pakistan, in 2011, and the M.S. degree in electrical and electronics engineering from Coventry University, U.K., in 2014. He is currently pursuing the Ph.D. degree with the Huazhong University of Science and Technology, Wuhan, China. Since May 2012, he has been a Permanent Faculty Member with the Department of Electrical Engineering, Mirpur University of Science and Technology. His current research interests include software-defined networks, wireless LANs, and programming.

YACHAO YIN received the B.S. degree from Hubei Engineering University, Hubei, China, in 2016, and the M.S. degree in electronic and information engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2019. Since July 2019, he has been an Engineer with the Department of Algorithms, ZTE Corporation. His current research interest includes next-generation wireless networks.

YAYU GAO (Member, IEEE) received the B.Eng. degree in electronic engineering from the Nanjing University of Posts and Telecommunications, Nanjing, China, in 2009, and the Ph.D. degree in electronic engineering from the City University of Hong Kong, Hong Kong, in 2014. From October 2017 to September 2018, she visited the FuNLab, University of Washington, Seattle, USA, where she collaborated with Prof. Sumit Roy on heterogeneous networks coexistence. Since 2014, she has been with the School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan, China, where she is currently an Assistant Professor. Her current research interests include random access networks, wireless LANs, and heterogeneous coexisting networks in 5G NR unlicensed.

XIAOJUN HEI (Member, IEEE) received the B.Eng. degree in information engineering from the Huazhong University of Science and Technology, Wuhan, China, in 1998, the M.Phil. degree in electrical and electronic engineering from The Hong Kong University of Science and Technology, Hong Kong, in 2000, and the Ph.D. degree from the Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology, in 2008. From 2005 to 2007, he conducted a research visit on P2P networking with the Department of Computer Science and Engineering, NYU Tandon School of Engineering, Brooklyn, NY, USA. Since 2008, he has been with the Internet Technology and Engineering Research and Development Center, School of Electronic Information and Communications, Huazhong University of Science and Technology. He is currently an Associate Professor with the School of Electronic Information and Communications, Huazhong University of Science and Technology. He is a coauthor, together with Y. Liu and K. W. Ross, of the Best Paper in Multimedia Communications for 2008 awarded by the Multimedia Communications Technical Committee of the IEEE Communications Society. His current research interests include edge networking, intelligent healthcare, and embedded network systems.

WENQING CHENG (Member, IEEE) received the B.S. degree in telecommunication engineering and the Ph.D. degree in electronics and information engineering from the Huazhong University of Science and Technology, Wuhan, China, in 1985 and 2005, respectively. She is currently a Professor with the School of Electronic Information and Communications, Huazhong University of Science and Technology. Her current research interests include information systems and e-learning applications.

• • •