

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/221453528>

# Overhaul of IEEE 802.11 modeling and simulation in NS-2

Conference Paper · October 2007

DOI: 10.1145/1298126.1298155 · Source: DBLP

CITATIONS

299

READS

501

6 authors, including:



[Marc Torrent-Moreno](#)

Eurecat, Barcelona

39 PUBLICATIONS 1,831 CITATIONS

[SEE PROFILE](#)



[Luca Delgrossi](#)

Daimler

55 PUBLICATIONS 1,850 CITATIONS

[SEE PROFILE](#)



[Hannes Hartenstein](#)

Karlsruhe Institute of Technology

214 PUBLICATIONS 8,532 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



SUPERHUB [View project](#)

All content following this page was uploaded by [Marc Torrent-Moreno](#) on 27 April 2017.

The user has requested enhancement of the downloaded file.

# Overhaul of IEEE 802.11 Modeling and Simulation in NS-2

Qi Chen \*, Felix Schmidt-Eisenlohr <sup>+</sup>, Daniel Jiang \*,

Marc Torrent-Moreno <sup>+</sup>, Luca Delgrossi \*, Hannes Hartenstein <sup>+</sup>

\* DaimlerChrysler Research, Engineering  
and Design North America Inc.

{qi.chen, daniel.jiang, luca.delgrossi}  
@daimlerchrysler.com

<sup>+</sup> Institute of Telematics  
University of Karlsruhe (TH)

{fschmidt, torrent, hartenstein}  
@tm.uni-karlsruhe.de

## ABSTRACT

NS-2, with its IEEE 802.11 support, is a widely utilized simulation tool for wireless communications researchers. However, the current NS-2 distribution code has some significant shortcomings both in the overall architecture and the modeling details of the IEEE 802.11 MAC and PHY modules. This paper presents a completely revised architecture and design for these two modules. The resulting PHY is a full featured generic module able to support any single channel frame-based communications (i.e. it is also able to support non-IEEE 802.11 based MAC). The key features include cumulative SINR computation, preamble and PLCP header processing and capture, and frame body capture. The MAC accurately models the basic IEEE 802.11 CSMA/CA mechanism, as required for credible simulation studies. The newly designed MAC models transmission and reception coordination, backoff management and channel state monitoring in a *structured* and *modular* manner. In turn, the contributions of this paper make extending the MAC for protocol researches much easier and provide for a significantly higher level of simulation accuracy.

## Categories and Subject Descriptors

I.6 [Computing Methodologies]: Wireless Communications Simulation

## General Terms

Design, Verification

## Keywords

IEEE 802.11, Simulator, NS-2

## 1. INTRODUCTION

### 1.1 NS-2 overview

Network Simulator, NS-2 [3], was originally developed for networking research with substantial support for simulation of TCP, routing, and multicast protocols over wired networks. The latest NS-2 distribution package is version 2.31, which was

released on March 10, 2007. The default implementation for simulation of IEEE 802.11 within NS-2 is strongly influenced by Sun Microsystems and the UC Berkeley Daedalus and CMU Monarch projects. It has been widely cited by wireless communications and networking researchers in all fields as the underlying simulation platform for their work.

However, there have long been recognitions among simulator developers and users that NS-2's IEEE 802.11 modeling has deficiencies. As described in the next section, several teams have tried to improve the NS-2 implementation recently. This paper presents the work produced by collaborations between DaimlerChrysler REDNA and University of Karlsruhe. Instead of patching up the existing NS-2 implementation, this work is focused on a complete redesign of IEEE 802.11 modeling.

### 1.2 Content and scope

This work is focused on the modeling of IEEE 802.11 PHY and MAC DCF (i.e. the CSMA/CA) mechanisms. Upper MAC layer features such as beacons, association, etc. are outside the scope of this work. The design, however, is friendly to further implementation of such features. The descriptions in this paper point out how such future update could be supported in each module.

This revised design of MAC and PHY is generic and meant to simulate IEEE 802.11a, b, g, and the currently being drafted p standards, given proper parameter settings. However, its suitability for IEEE 802.11n study needs to be further evaluated.

While the choice of RF propagation modeling method has strong influences on the wireless communications studies, it is contained in an independent module outside of the MAC and PHY. Due to space constraints, this paper does not discuss RF propagation modeling. Interested readers should see e.g. [11].

## 2. PREVIOUS WORK

Several publications ([8], [9], [10]) have shown a lack of accuracy in NS-2 when looking at the models used for packet interferences and packet reception, stating that the assumptions for simplistic models do not cover the effects occurring in a real world setup. It is also shown that using more appropriate models changes the characteristics of simulation results enormously in many cases. There exist approaches to overcome these weaknesses ([12], [13]) although these ones state that the current structure of NS-2 hardly allows improvements in a great manner. In [4], [5], [6], [7], the two groups involved in this work presented improvements towards a higher-accuracy version of the NS-2 simulator for wireless applications. This work also includes the models proposed for capture effect [14]

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MSWiM'07, October 22–26, 2007, Chania, Crete Island, Greece.

Copyright 2007 ACM 978-1-59593-851-0/07/0010...\$5.00.

and follows the design of IEEE 802.11 given in the standard documentation, [1] and [2].

### 3. IEEE 802.11 TRANSMISSION AND RECEPTION OVERVIEW

This section describes details of the IEEE 802.11 transmissions and receptions at a level that is meaningful for simulator designs. This overview is intended to demonstrate the importance of proper preamble and PLCP header handling in the simulation design, which is not modeled in the current NS-2.

While these details are no mystery to wireless engineers, they are often not well understood by NS-2's users, many of which are usually trained in layer 3 and above and only have an overly abstract view of the PHY layer. On the other hand, the IEEE 802.11 CSMA/CA mechanism at the MAC level is well described in the literature and will not be covered here.

#### 3.1 Modulation and coding rate

The perceivable data rates of IEEE 802.11 radios are determined by modulation and coding rate used in transmissions. Modulation is the method in varying a periodic waveform in order to convey some information. As shown in Table 1, which lists all the defined transmission options for IEEE 802.11a radios, the BPSK modulation method is able to convey 1 bit of information per periodic waveform per sub-carrier. Because IEEE 802.11a radios transmit data over 48 sub-carriers concurrently in its OFDM design, BPSK allows each OFDM symbol to carry 48 bit of information. Coding rate describes the fraction of the total carried bits that is used for actual data bits. The rest are redundancies used to correct errors in the reception process. For a 20 MHz channel, each periodic waveform (plus a guard interval to prevent inter-symbol interference) takes 4  $\mu$ s to send. Therefore, the combination of BPSK modulation with 1/2 coding rate results in the 6 Mbps baseline data rate supported by IEEE 802.11a radios.

Table 1, IEEE 802.11a modulations and coding rates

Modulation	Coded bits per sub-carrier	Coded bits per OFDM symbol	Coding rate	Data bits per OFDM symbol	Data rate for 20MHz channel
BPSK	1	48	1/2	24	6 Mbps
BPSK	1	48	3/4	36	9 Mbps
QPSK	2	96	1/2	48	12 Mbps
QPSK	2	96	3/4	72	18 Mbps
16-QAM	4	192	1/2	96	24 Mbps
16-QAM	4	192	3/4	144	36 Mbps
64-QAM	6	288	2/3	192	48 Mbps
64-QAM	6	288	3/4	216	54 Mbps

While a higher level of modulation and coding rate combination provide a higher data rate, the reception of such a frame is more likely subject to errors. In other words, it needs a cleaner and stronger received signal to successfully receive such a frame.

#### 3.2 PHY layer frame structure

A transmitter may select any defined modulation and coding rate combination to transmit a frame at any time as regulated by the MAC protocols. A receiver, however, needs to be able to

know if the signal on the channel is a frame instead of noise, how long the frame duration is and what modulation and code rate combination is used, in order to have an opportunity to successfully receive a frame.

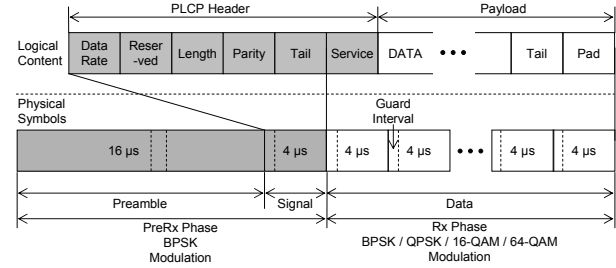


Figure 1, IEEE 802.11a PPDU frame format

As shown in Figure 1, the PHY layer frame always starts with a known training sequence called preamble to notify receivers on the arrival of a frame and assist them to lock on to the signal. The preamble is followed by a PLCP (Physical Layer Convergence Procedure) header, which contains details on the frame body including modulation and coding rate, frame length, etc. The preamble and the front part (i.e. Signal portion) of the PLCP header are both modulated by BPSK in almost all IEEE 802.11 radio configurations. While the preamble has no coding rate, the Signal part of the PLCP header is coded at 1/2 rate. The frame payload follows the PLCP header and is demodulated and decoded by the receivers according to the information contained in the Signal portion of the PLCP header.

#### 3.3 Frame reception process

A group of interacting IEEE 802.11 radios shares one channel for communications. In general, wireless radios are not able to send and receive simultaneously in the same channel. Therefore, a radio is always in one and only one of three conditions: listening on the channel to look for incoming frames, transmitting a frame when commanded by the MAC, and (attempting in) receiving a frame.

As a radio listens on the channel, it continuously looks for the known pattern of the preamble by demodulating the received signal according to the BPSK method. If such a pattern is found, the receiver will attempt to decode the Signal portion of the PLCP header. If this attempt is again successful, the receiver is then committed to demodulate the received RF waveforms for the frame duration according to the information recovered in the Signal portion of the PLCP header. From this point on to the end of the frame duration, the receiver treats all received signal as belonging to the incoming frame and attempts to demodulate it. The resulting raw bits will be given to the MAC for CRC check to finally determine if a frame is successfully received.

If a frame arrives when the node is in transmission, it will not be able to hear this frame. Even if its transmission state would end very soon after, this radio would not be able to receive the incoming frame because it would have missed the critical preamble and PLCP header and have no information to receive the frame body.

Similarly, if a node is already in the frame body reception process, it will not be able to receive an incoming frame because the node would treat the signal from the new frame as from the frame currently being received. (There, however, is an

exception to allow such a new frame to be received, which is described in section 3.3.2 below.) Since the node is not looking for the preamble of a new frame, it will not realize that a new frame has arrived. If the new frame has strong enough signal strength, it will collide with the existing frame and prevents its successful reception.

Because the frame body may be transmitted with a modulation and coding rate combination different from the BPSK and 1/2 coding rate used for the Signal portion of the PLCP header, it is possible that a receiver is able to receive and decode the preamble and PLCP header correctly yet still fail the subsequent reception of the frame body itself, depending on the signal quality and interferences.

### 3.3.1 Preamble capture

The IEEE 802.11 radios are designed to be very robust in the process of searching for and decoding the preamble and PLCP header. That is: if a new frame arrives when the receiver node is still receiving the preamble and PLCP header of an earlier frame, the new frame can be picked and locked onto by the receiver node if it has sufficiently higher power to be heard above the earlier one. This preamble capture feature is generally present in the IEEE 802.11 chips but is not widely modeled in the simulators.

### 3.3.2 Frame body capture

It is also possible to capture a new incoming frame during the frame body reception process of another one. Frame body capture is not part of the IEEE 802.11 standard but is implemented in some chips as an optional feature.

This feature works by having the PHY continuously monitor the received signal strength over the air. If there is suddenly a sharp rise (e.g. by 10 dB), the receiver assumes this is caused by the arrival of a new and much stronger frame. It then immediately abandons the previous frame and attempts to decode the preamble and PLCP header of the new frame. If successful, it then starts the reception process for the new frame.

Frame body capture is a useful feature for the typical WiFi usage cases in homes and offices. The weak frame is likely to come from another BSS (Basic Service Set) at some distance away, while the strong frame is likely to be from a node nearby and within the same BSS of the receiver. As such, frame body capture could improve reception rate of these meaningful transmissions from the receiver's perspective. Similarly, frame body capture could also make an impact for ad hoc communications and networking studies, if the hidden terminal effect is strong.

## 4. CURRENT IEEE 802.11 MODELING IN NS-2

The design of the default NS-2 wireless implementation is shown in Figure 2. It consists of several layered modules that are interconnected by simple interfaces for passing packets up and down along the layers. Design and responsibilities of each of the modules are briefly described below.

- The **WirelessChannel** interconnects all wireless nodes in a simulation scenario and interchanges frames among them. Its only function is to pass frames from each sender to all possible receivers, creating an exclusive copy for each of

them. It does not handle interferences, collisions, or path loss calculations.

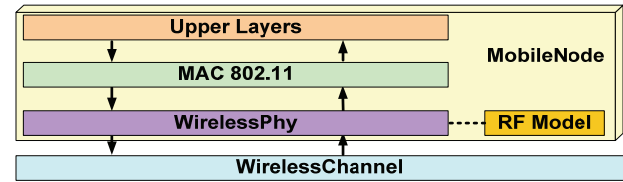


Figure 2, NS-2 wireless simulation framework

- The **WirelessPhy** module takes the frames coming from the wireless channel and requests for each of them the actual reception power  $P_r$  from the RF model object described in next section. If  $P_r$  is greater than the *Carrier Sense Threshold*, the frame is simply passed to the Mac802.11 module where all further handling is done. When a packet comes from the MAC, it is just passed to the wireless channel.
- In the **RF model** a specific RF propagation model is taken to determine the reception power for each packet. The individual power depends on the selected model, and is, among other factors, depending on the distance between transmitter and receiver. For detailed discussion of propagation models we refer the reader to our earlier work.
- The **MAC 802.11** module contains most of the complexities in IEEE 802.11 modeling. It not only handles the coordination of channel access for transmissions (which is appropriate for a MAC module), but also determines physical packet reception and collision management (which should be handled by the PHY module).

The biggest drawback of this implementation is structural in nature. Most of the PHY functionalities are mixed up in the MAC. As a result, it is very difficult, if not impossible, to model everything correctly at both the physical and logical levels. The overly complex MAC module is also a big challenge for the users to understand and extend in their researches. Beyond this, there are also many instances of over-simplification or inaccuracies in the IEEE 802.11 modeling. Some of them are listed below but not discussed due to space constraints. Please see our prior work (e.g. [4], [5]) for details.

- Wrong collision handling
- No preamble and PLCP header modeling
- No cumulative SINR implementation
- Wrong backoff handling
- Misusage of the network allocation vector (e.g. EIFS)
- Incomplete support for capture

## 5. OVERHAULED IEEE 802.11 MAC AND PHY MODULES

### 5.1 Architecture

The revised IEEE 802.11 modeling directly fits into the overall NS-2 wireless simulation framework as shown in the previous section. That is: all nodes are attached to a common Wireless Channel object through the PHY; whenever a node transmits, the channel notifies all attached nodes of the transmission event

with location, power, etc. details; each node's PHY also communicates with a RF Model object, which computes received signal strength for each incoming transmission using a selected RF propagation model and the node's own position.

The architecture of the IEEE 802.11 MAC and PHY modeling, as shown in Figure 3, is now completely different from the default NS-2 implementations. Instead of putting everything inside the MAC, all functionalities of the IEEE 802.11 radio are now cleanly and properly separated between the MAC and PHY.

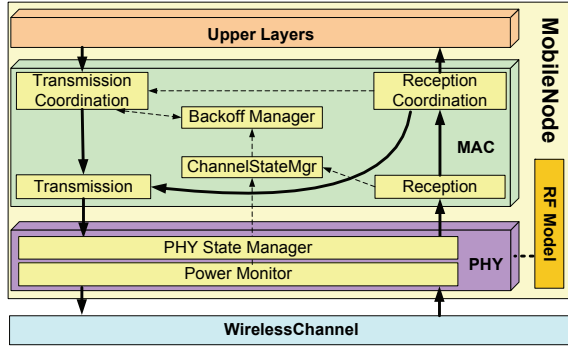


Figure 3, Revised IEEE 802.11 simulation architecture

The MAC module now only operates at the logical level. It depends on the PHY to handle actual transmissions, receptions and physical channel sensing. The focus of the MAC design is to correctly and cleanly model all the complexities in the IEEE 802.11 CSMA/CA mechanism which is also known as DCF (Distributed Coordination Function). A key purpose of this new design is to make it easy for users to understand and extend the MAC for their researches.

The PHY module handles all physical layer related issues, such as channel sensing, SINR tracking and PLCP state management. While the work on the PHY is part of the overall IEEE 802.11 modeling, its design is sufficiently generic so that it is able to support the implementation of different MAC designs on top.

In the following subsections, all modules within the MAC and PHY are described. The appendix section at the end of the paper provides more details in forms of SDL diagrams of these modules.

## 5.2 PHY modeling

As shown in Figure 3, there are two modules placed in the PHY. The power monitor module keeps track of received RF signals from all the transmission events over the air. The PHY state manager is responsible to maintain PLCP states.

### 5.2.1 Power monitor

The power monitor module corresponds to the PMD (Physical Media Dependent) sub-layer within the PHY. PMD is the only sub-layer that directly interacts with the analog RF signals. Therefore, all information on received signals is processed and managed in this module.

The power monitor module keeps track of all the noise and interferences experienced by the node individually for their respective durations. Whenever the cumulative interference and noise level rises crosses the carrier sense threshold, it signals the MAC on physical carrier sense status changes. It should be

noted that a node's own transmission is treated as carrier sense busy through this signaling interface as well.

### 5.2.2 PHY state manager

The PHY state manager models the PLCP sub-layer, which is the logic sub-layer of the PHY. As shown in Figure 4, this state manager keeps track of how the PHY switches among four operating modes.

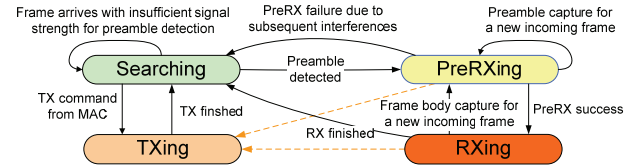


Figure 4, PHY state machine

The PHY is in the Searching state when it is neither in transmission nor reception of a frame. Within this state, the PHY evaluates each transmission event notification from the Wireless Channel object it is attached to for potentially receivable frames. If a frame arrives with sufficient received signal strength for preamble detection (i.e.  $\text{SINR} > \text{BPSK threshold}$ ), the PHY moves into the PreRXing state.

The PHY stays in the PreRXing state for the duration of preamble and Signal portion of the PLCP header. If the SINR of this frame stays above the BPSK and 1/2 coding rate reception threshold throughout this period, the PHY moves into the RXing state to stay for the frame duration. If a later arriving frame from the channel has sufficient received signal strength to prevent proper preamble and PLCP header reception for the current frame, the PHY moves back to the Searching state. However, if this later frame has sufficiently higher signal strength for its own preamble to be heard above others, it will trigger preamble capture, which means the PHY stays in the PreRXing state with a reset timer for the new frame.

Within the RXing state, the PHY handles the reception of the body of the current frame. It monitors the SINR throughout the frame body duration. If the SINR drops below the threshold required by the modulation and coding rate used for the frame body at any time while in this state, the PHY marks the frame with an error flag. After RXing timeout, the PHY moves back to the Searching state. It also passes the frame to the MAC, where the error flag is directly used for the CRC check decision.

If the frame body capture feature is enabled, then it is possible for a later arriving frame to trigger the PHY to move back to the PreRXing state for the new frame in the manner described in section 3.3.2. Otherwise, the later arriving frame has no chance of being received and is only tracked by the power monitor as an interference source.

A transmit command from the MAC will move the PHY into the TXing state for the duration of a frame transmission regardless what the PHY is doing at the moment. The expiration of the transmission timer ends the TXing state. If a frame comes in from the channel when the PHY is in the TXing state, it is ignored and only tracked by the power monitor as interference.

Usually the MAC will not issue a transmit command while the PHY is in the PreRXing or RXing state because of the carrier sense mechanism. However, the IEEE 802.11 standard mandates the receiver of a unicast data frame addressed to itself to turn

around after SIFS and transmit an ACK frame regardless of the channel condition. Similarly, the receiver of a RTS frame, if it has an empty NAV (Network Allocation Vector), will wait for SIFS and then transmit a CTS frame regardless of the channel condition. Such scenarios are represented by the two dashed lines shown in the state machine. The PHY is designed to drop and clean up the frame it is attempting to receive and move into TXing state when this happens.

The MAC, however, should never issue a transmit command when the PHY is still in the TXing state. The new frame has little chance of being received within its intended audience because others in general have no means to tell that a new frame is suddenly started. Therefore, the PHY is designed to issue an error and halt the simulator when this event happens because it means the MAC above has a critical error in design or implementation.

### 5.2.3 Frame drop reasons breakdown

The PHY now models frame reception failures in very detailed manner. In turn, it is able to produce traces that provide much needed insights for researchers when they analyze simulation results. The table below lists all the frame drop reasons at a receiving node. Please see the SDL of the PHY at the end of this paper for precise details and case numbers.

**Table 2, Frame drop reasons breakdown**

Case #	Frame Drop Reasons at a Receiving Node
1	Received signal too weak for the frame to be detected above noise/interference
2	PreRX failed due to subsequent interferences
3	Signal strength not enough for preamble capture
4	Node already in PreRX for another frame
5	Signal too weak for frame body decoding
6	Interrupted by node's own transmission command
7	Interrupted by node's own transmission command
8	Node already in reception of another frame
9	Frame body reception failed due to subsequent interferences
10	Signal not enough for frame body capture
11	Node already in reception of another frame (capture OFF)
12	Node is in transmission of its own

### 5.2.4 Future extensions

This section describes a few features to be extended in the future. Their design approaches are also briefly discussed.

- **Bit Error Rate (BER) based frame reception** decision is a more sophisticated method than the current design that uses a threshold for frame reception decision directly. Using a lookup table, the PHY finds the BER for a frame based on the SINR value. It in turn calculates the Packet Error Rate (PER) using the BER and the frame length. This method is able to model the impact of a frame's size.
- **Sleeping (i.e. power saving) mode** allows a radio to be turned off (and back on) in power constrained wireless communications systems. This is particularly useful for

protocol studies of various mobile ad hoc and/or sensor networks.

- **Channel switching** supports simulations of multiple channels concurrently used by different radios. A node may choose to switch from one channel to another dynamically. This feature is useful for researchers in the fields of cognitive radios, VANET, etc.

Of these three features, the BER support is the easiest to implement since it involves mostly substituting of the current frame reception decision code locally. The other two features, however, require more complicated updates to the PHY design.

The sleeping mode implies the addition of a sleeping state, which is accessible from the Searching state. The question is: should the power monitor in the PHY keep tracking of the activities on the channel while in the sleeping state? Obviously, keep tracking of signals on the channel enables correct behaviors at the immediate moment when the PHY returns back from the sleeping state. On the other hand, turning off power monitor (i.e. delisting a node from the channel) could greatly reduce the computational load of simulations, especially in scenarios in which many nodes stay in sleeping mode most of the time.

The channel switching feature also faces a similar question: should each node monitor activities on channels other than its current one? If it does not, how could it know the local channel condition as it switches to a new channel?

There are two potential approaches in addressing the problems described above. The channel object could be updated so that it tracks all live transmissions and provides such information when queried by any node's PHY module. Or there could be one more "transitioning" state introduced to the PHY design. A PHY exiting the sleeping state or channel switching is required to go to the transitioning state and stay for one or two ms. While in the transitioning state, the power monitor is turned in its tracking of the channel activities. This transitioning state can be justified as modeling the overhead/latency involved by the PHY before settling down in the searching state.

## 5.3 MAC modeling

As shown in Figure 3, there are six modules defined within the MAC. These six are 1) transmission, 2) reception, 3) channel state manager, 4) backoff manager, 5) transmission coordination and 6) reception coordination. The solid lines in the figure show the paths for passing data and control frames while the dashed lines indicate active signaling interfaces among the modules.

The design of these six modules reflects considerations and abstractions of the extensive state diagrams contained in the IEEE 802.11 standard. However, not all decisions made in this work conform to the design in the standard. For example, the reception module in this design is simplified and some of the responsibilities (e.g. duplicate frame removal) are moved over to the reception coordination module instead.

### 5.3.1 Transmission module

The transmission module is the interface to the PHY. It passes all types of frames to the PHY for actual transmissions.

The transmission module accepts RTS and data frames from the transmission coordination module, and ACK and CTS frames



from the reception coordination module. When upper MAC management features (e.g. associations) are incorporated, it will also accept management frames from the new modules.

The transmission module has a very simple state machine, which consists of only two states: TX\_IDLE and TXing.

### 5.3.2 Reception module

The reception module completes the reception process of an incoming frame, which is started at the PHY. It applies address filtering on successfully received frames before passing them to other modules. It is also responsible to signal to the channel state manager with virtual carrier sense updates.

As a frame comes up from the PHY, the reception module performs a CRC check to see if the frame is successfully received. It does so by consulting the value of the error flag, which is set by the PHY, attached to the frame.

The standard requires a node that just received a bad or unknown frame to wait for an EIFS (Extended InterFrame Spacing) instead of the standard DIFS. The inter-frame spacing is treated in this design as the responsibility of the channel state manager. Therefore, the reception module informs the channel state manager in cases of CRC check error.

The reception module applies address filtering on all successfully received frames, and only passes a frame to the appropriate recipient module if the frame is intended for this node. When BSS or IBSS support is added in the future, the BSSID filtering process should be placed in this module.

For now, the reception module only passes data and control frames to the reception coordination module. When upper MAC layer features are added, the reception module should pass management frames to the new modules as appropriate.

Any frame that does not pass the address filtering process is examined to see if it contains a NAV before being discarded. RTS, CTS and sometimes data frames contain NAV information. If one is found, the reception module passes the NAV value to the channel state manager.

The reception module also has a simple state machine. It is in either RX\_IDLE or RXing at any time.

### 5.3.3 Channel state manager

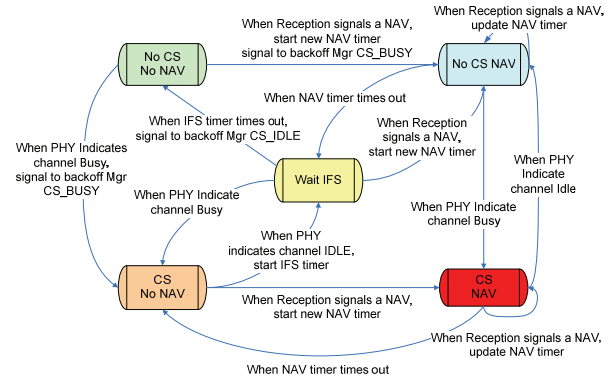
The channel state manager is responsible for maintaining both the physical and virtual carrier sense statuses for the IEEE 802.11 CSMA mechanism.

The channel state manager depends on the PHY to update the physical carrier sense status. It expects the PHY to signal channel busy when the total received signal strength rises above carrier sense threshold or when the PHY is in transmission. Similarly, it expects the PHY to signal channel clear when both conditions are gone.

The channel state manager expects signaling from the reception module for virtual carrier sense status updates as described in the above subsection. Once signaled this way, the channel state manager sets or updates the NAV for the duration specified.

As shown in Figure 5, the channel state manager has five states. The combination of both physical and virtual carrier sense statuses result in four states: NoCSnoNAV, NoCSNAV, CSnoNAV and CSNAV. Additionally, the time spent in InterFrame Spacing (IFS) waiting is also modeled as a state

within the channel state manager. This is because the IFS mechanism is essentially a self enforced NAV. As such, the channel state manager treats the Wait IFS state as channel being busy as well. The duration of IFS waiting time depends on the ifs\_value parameter, which can be set to DIFS and EIFS. When EDCA is incorporated, it will be necessary to add an additional signaling interface from the transmission coordination module to the channel state manager to advise the AIFS values. SIFS waiting, however, is handled by the transmission and reception coordination modules directly. This is because SIFS is used in a way unlike that of DIFS and EIFS. Any module that sets a SIFS timer will take action as the timer expires regardless of the channel condition during the SIFS.



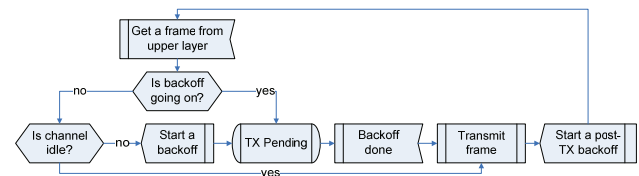
**Figure 5, Channel state manager state machine**

The channel state manager reports the joint physical and virtual carrier sense status in response to queries from any other module. That is: it reports CS\_IDLE if it is in the NoCSnoNAV state, and CS\_BUSY otherwise. It also reports the NAV status to the reception coordination module when queried, to assist the CTS decision.

The channel state manager actively signals the backoff manger whenever it moves in or out of the NoCSnoNAV state to indicate channel status changes. In turn the backoff manager resumes or pauses its backoff process if it is already in one.

### 5.3.4 Backoff manager

The backoff manager maintains the backoff counter to support the collision avoidance mechanism in the IEEE 802.11 MAC.

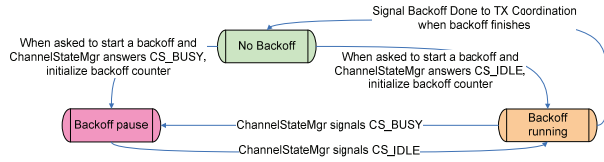


**Figure 6, Backoff manager's utilization by the transmission coordination module**

The backoff manager assists the transmission coordination module to run both the regular backoff and post-transmission backoff, but is not aware of the difference between the two. Figure 6 provides a very simplified view on how this is done.

As shown in Figure 7, the backoff manager has three states: No Backoff, Backoff Running, and Backoff Pause. It depends on the signaling of channel carrier sense state from the channel

state manager to run or pause the backoff counter. It moves back to the No Backoff state and signals Backoff Done to the transmission coordination module when the backoff counter reaches zero. When EDCA is added, it will be necessary for an additional signaling interface to be implemented between the transmission coordination module and the backoff manager. This is used to trigger a reassessment of the backoff process with a different cw (Contention Window) value. In turn, it could cause the ending of the backoff sooner in cases of higher priority pending frames.

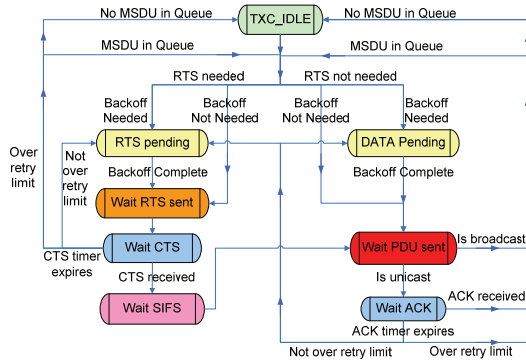


**Figure 7, Backoff manager state machine**

### 5.3.5 Transmission coordination module

The transmission coordination module manages channel access for packets passed down from the upper layer.

Figure 8 illustrates the state machine in the transmission coordination module. It is roughly divided into two sides depending on whether the RTS/CTS exchange is needed. If the data frame is a broadcast or a unicast with size less than the RTS threshold, it is entirely processed within the right side of the overall state machine. Otherwise, a RTS frame is generated and a sequence of states on the left side is involved before the data frame would be sent.



**Figure 8, Transmission coordination module state machine**

When the transmission coordination module moves out of the TXC\_IDLE state because of a packet coming down from the upper layer, it first checks if a RTS frame should be generated. Afterwards, it starts a backoff process at the backoff manager if there is not one going on already and moves into the RTS Pending or Data Pending state according to the RTS decision.

If the transmission coordination module is in the RTS Pending or Data Pending state, it instructs the transmission module to transmit the RTS or data frame respectively as soon as receiving a signal indicating Backoff Done from the backoff manager.

It is possible for these pending states to be directly bypassed. As shown in Figure 6, if the backoff manager does not have a backoff process currently going on and the channel state manager replies with a CS\_IDLE, the transmission coordination module can immediately transmit the RTS or data frame. This is

because the standard allows for a radio to start a transmission right away if it has completed a previous post-transmission backoff and the channel, both physically and virtually, has been idle for more than DIFS.

When the transmission of a RTS frame is completed, the transmission coordination module moves into the Wait CTS state and starts a timer TCTS. If the reception coordination module does not signal the arrival of a CTS frame before the timer expires, it starts another backoff process and moves back to the RTS pending state. It can repeat this process until the short retry limit is reached. If a CTS response comes back in time, then the transmission coordination module waits for SIFS before instructing the transmission module to send the data frame.

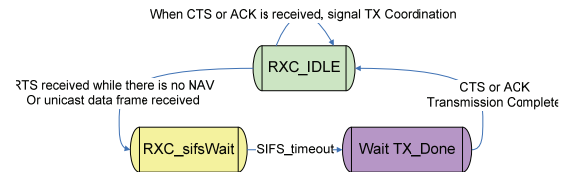
After the data frame transmission, the transmission coordination module moves into the Wait ACK state and starts a TACK timer. If it does not get an ACK reply indication from the reception coordination module before the timer expires, it starts a new backoff process and moves back to RTS Pending or Data Pending state respectively. Again, this is subject to the short or long retry limit respectively.

In cases of RTS retry or unicast retransmission, the cw parameter is updated before a backoff is requested with the new value.

The attempted transmission of a frame finishes in three possible ways: 1) it is a broadcast frame and it is transmitted once over the air; 2) it is a unicast frame and the transmission coordination module receives an ACK signaling from the reception coordination module; or 3) one of the retry limits is reached. In all three cases, the transmission coordination module resets the retry counters and the cw parameter, and starts a post-transmission backoff. Afterwards, if there is a packet waiting in the queue, it takes the packet and immediately moves into the RTS Pending or Data Pending state. Otherwise it returns back to the TXC\_IDLE state.

### 5.3.6 Reception coordination module

The reception coordination module takes from the reception module control and data frames meant for this node. It signals the transmission coordination module when CTS and ACK frames arrive. It is responsible to handle the CTS and ACK responses when RTS and data frames arrive. It also filters the data frames before passing them to the upper layer.



**Figure 9, Reception coordination module state machine**

As shown in Figure 9, the reception coordination module has only three states: RXC\_IDLE, RXC SIFS Wait, and Wait TX Done. The reception coordination module mostly spends time in the RXC\_IDLE state and awaits control and data frames from the reception module.

If a RTS frame arrives, the reception coordination module queries the channel state manager for its NAV status. If the response indicates an active NAV, the RTS frame is simply



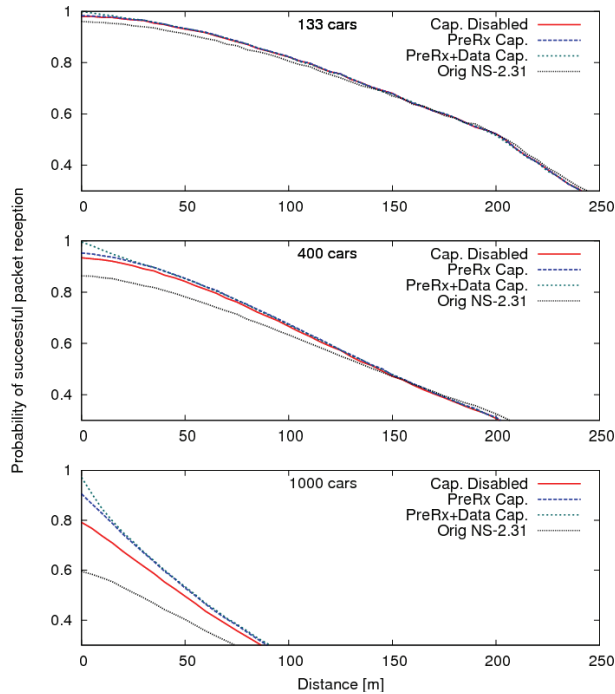
discarded. Otherwise, the reception coordination module creates a CTS frame and moves into the RXC SIFS Wait state with a SIFS timer set. When the timer expires, it immediately instructs the transmission module to transmit the CTS frame. It then moves into the Wait TX Done state for the transmission duration before returning to the TXC\_IDLE state.

If a unicast data frame arrives, the reception coordination module starts an ACK process similar to the way it handles the CTS response. However, it does not consult the channel state manager for the NAV status. If a CTS or ACK frame arrives, the reception coordination module simply signals the transmission coordination module accordingly.

The reception coordination module passes data frames to the upper layer. In this process, it is responsible to discard duplicate data frames coming in from the channel, which can be caused by the unicast retransmission mechanism. In these cases, however, it still reacts with an ACK transmission.

## 6. SAMPLE SIMULATION RESULTS

The simulation studies presented in this section are set in the VANET context. 133, 400 and 1000 car nodes are pseudo-uniformly placed on a circle with a 2000 m perimeter in three scenarios. Imagine that this circle is a road around the equator of a greatly shrunken earth. As such, the boundary effect is eliminated. Each node transmits on average 10 broadcast frames per second with a payload of 250 Byte. The transmission power has a theoretic range of 250 m. Rayleigh model is selected in the RF Model. The IEEE 802.11p [2] (which is similar to IEEE 802.11a half rated with 10 MHz channels) parameters are used.



**Figure 10, Effects of IEEE 802.11 modeling updates under different node density scenarios**

These three scenarios are equivalent to 120, 40, and 16 m per car per lane in an 8-lane highway. They represent moderate, realistically dense, and extremely dense node densities

respectively. Figure 10 shows a comparison of the average broadcast frame reception rate at various distances from the sender in these scenarios. Note that the Y-axis is truncated in its range (i.e. it does not cover from 0 to 1) for space considerations. In each scenario, one and three sets of results are obtained with the original (version 2.31) and the overhauled NS-2 simulators respectively. The overhauled simulator is used with three configurations: all capture features turned off; preamble capture turned on; and preamble and frame body captures both turned on.

### 6.1 Comparison with original NS-2

The “*Orig NS-2.31*” and “*Cap. Disabled*” lines in Figure 10 provide level comparisons between the original and the overhauled NS-2 simulators. The original NS-2 implementation lacks a cumulative SINR model. As a result, it produces higher reception rates at further distances (e.g. the *Orig NS-2.31* line shows up on the top beyond 150 m in the middle plot). At the closer distances, however, the results are reversed. This is due to the incorrect MAC collision state modeling (i.e. continuous incoming frames could extend the collision state forever) in the original NS-2. Please see [4] for detailed descriptions. It is also clear that these differences increase when the scalability of the simulation scenario increases.

### 6.2 Effect of capture features

The comparison between the “*Cap. Disabled*”, “*PreRx Cap.*”, and “*PreRx+Data Cap.*” lines in Figure 10 demonstrates differences made by the preamble and frame body capture features that are now implemented in the PHY. Both capture features strongly influence the reception rates at ranges close to the sender. These effects are more pronounced as the simulation scenario becomes denser (and introduces stronger hidden terminal factors). As such, these capture features have implications for inter-vehicular safety communications as well as the general MANET researches.

## 7. SUMMARY

This paper presents a clean and modular architecture for IEEE 802.11 simulations in NS-2. The roles, functions and signaling interfaces of all major modules in the MAC and PHY are described systematically. Microscopic design details are provided in forms of SDL diagrams. As such, the overhauled NS-2 simulator is easy to understand and extend by wireless communications researchers.

The complete and yet detailed descriptions also act as a blue print for anyone interested in building a IEEE 802.11 simulation core in simulation platforms other than NS-2.

In terms of modeling accuracy, this paper makes a strong case for the need to model the IEEE 802.11 transmissions and receptions process realistically and correctly. The addition of cumulative SINR, preamble and PLCP header handling and capture, and frame body capture features to the PHY improves simulation accuracy and provides more insights to researchers.

## 8. ACKNOWLEDGMENTS

Felix Schmidt-Eisenlohr acknowledges the support of the German Research Foundation (DFG) within the Research Training Group GRK 1194 “Self-organizing Sensor-Actuator Networks”.

## 9. REFERENCES

- [1] "IEEE Std. 802.11-2007, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications," IEEE Std. 802.11, 2007.
- [2] "IEEE P802.11p/D2.01, Draft Amendment for Wireless Access in Vehicular Environments (WAVE)," February 2007.
- [3] "Network Simulator ns-2," <http://www.isi.edu/nsnam/ns/>.
- [4] Q. Chen, D. Jiang, V. Taliwal, and L. Delgrossi, "IEEE 802.11 Based Vehicular Communication Simulation Design for NS-2," in *Proceedings of the International Workshop on Vehicular Ad Hoc Networks (VANET)*, Los Angeles, CA, USA, Sept. 2006.
- [5] F. Schmidt-Eisenlohr, J. Letamendia-Murua, M. Torrent-Moreno, and H. Hartenstein, "Bug Fixes on the IEEE 802.11 DCF module of the Network Simulator ns-2.28," Department of Computer Science, University of Karlsruhe, Tech. Rep. TR-2006-1, Jan. 2006.
- [6] F. Schmidt-Eisenlohr, M. Torrent-Moreno, T. Tielert, J. Mittag, and H. Hartenstein, "Cumulative Noise and 5.9GHz DSRC Extensions for ns-2.28," Department of Computer Science, University of Karlsruhe, Tech. Rep. TR-2006-21, Oct. 2006.
- [7] F. Schmidt-Eisenlohr, M. Torrent-Moreno, J. Mittag, and H. Hartenstein, "Simulation Platform for Inter-Vehicle Communications and Analysis of Periodic Information Exchange," in *Proc. of the Conference on Wireless On demand Network Systems and Services (WONS)*, Obergurgl, Austria, January 2007.
- [8] M. Takai, J. Martin, and R. Bagrodia, "Effects of wireless physical layer modeling in mobile ad hoc networks," in *Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing (MOBIHOC 2001)*, pages 87–94, New York, NY, USA, 2001. ACM Press.
- [9] D. Kotz, C. Newport, R.S. Gray, J. Liu, Y. Yuan, and C. Elliot, "Experimental Evaluation of Wireless Simulation Assumptions", in *Proceedings of the MSWiM'04*, Oct. 2004
- [10] M. Torrent-Moreno, D. Jiang, and H. Hartenstein, "Broadcast Reception Rates and Effects of Priority Access in 802.11-Based Vehicular Ad-Hoc Networks," in *Proceedings of the 1st ACM International Workshop on Vehicular Ad Hoc Networks (VANET)*, Philadelphia, PA, USA, Oct. 2004.
- [11] M. Torrent-Moreno, S. Corroy, F. Schmidt-Eisenlohr, and H. Hartenstein, "IEEE 802.11-Based One-Hop Broadcast Communications: Understanding Transmission Success and Failure under Different Radio Propagation Environments," in *Proceedings of the ACM/IEEE International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, Torremolinos, Spain, Oct. 2006.
- [12] T. Henderson, S. Roy, S. Floyd, and G.F. Riley, "ns-3 project goals," in *Proceedings of the 2006 workshop on ns-2: the IP network simulator*, Pisa, Italy, October 2006
- [13] M. Lacage, and T.R. Henderson, "Yet another network simulator," in *Proceedings of the 2006 workshop on ns-2: the IP network simulator*, Pisa, Italy, October 2006
- [14] A. Kochut, A. Vasan, A.U. Shankar, and A. Agrawala, "Sniffing out the correct physical layer capture model in 802.11b," in *Proceedings of the 12th International Conference on Network Protocols (INCP'04)*, Washington D.C., USA; 2004

## 10. APPENDIX

The following two figures present the design details of the MAC and PHY, as described in the paper.

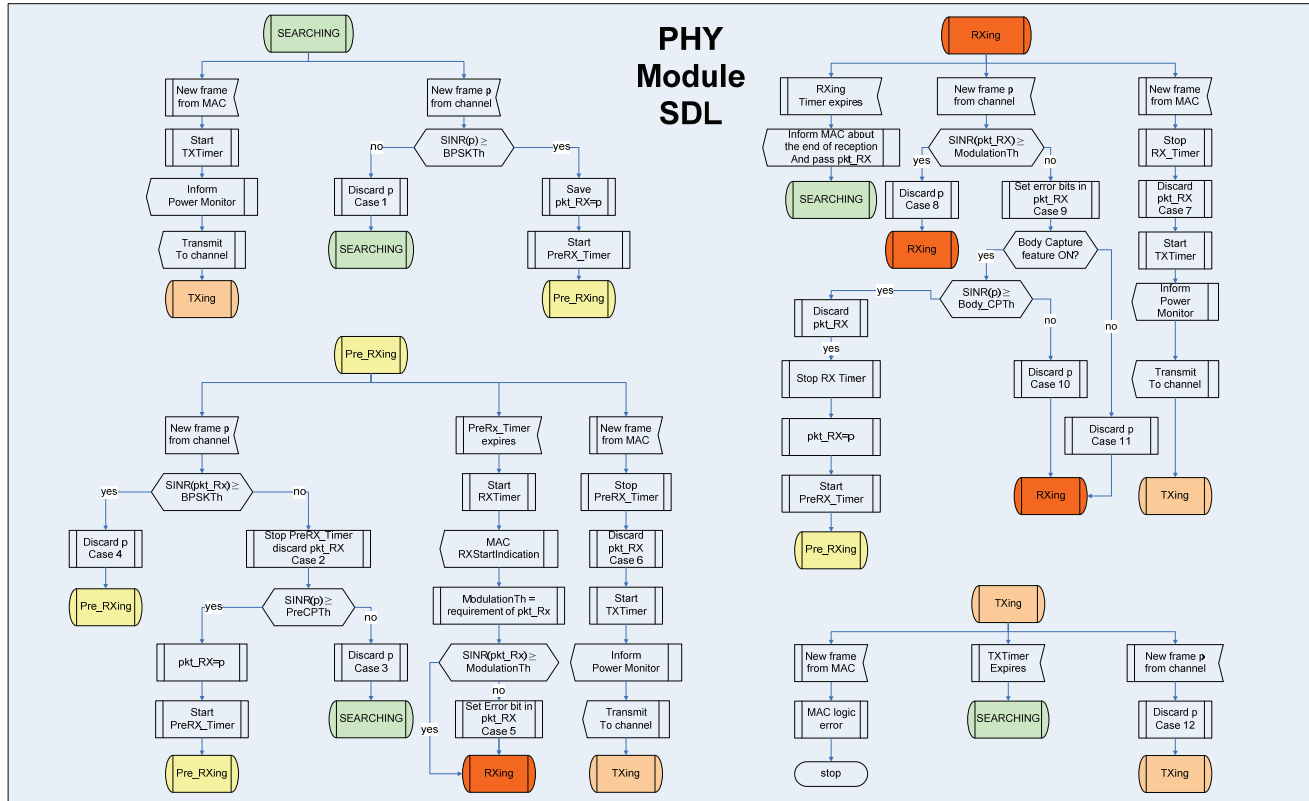


Figure 11, Design details of the PHY

