Test Generation Framework for Evaluation of Wireless MAC Protocols

Shamim Begum
Electrical Engineering, University of Southern California
Email: sbegum@usc.edu

Abstract

The MAC protocol is the main determinant of the efficiency of sharing the limited communication bandwidth of a wireless channel. While new protocols and architectures are being designed to improve efficiency, no systematic approach has been proposed for testing these protocols. Traditional performance evaluation approaches typically evaluate average performance but do not capture the worst cases, nor do they expose the breaking points.

We propose a novel framework to critically analyze a given MAC protocol with respect to its correctness criteria and performance metrics. The framework is composed of error generation and test scenario generation algorithms. The error generation algorithm generates a set of conditions that meet the study objective. The test generation algorithm then generates complete test scenarios that satisfy our objective, e.g., minimize the value of a particular performance metric. The test generator employs efficient forward and backward search techniques to construct the sequence of events that satisfy our objective. We apply heuristics to prioritize the search alternatives that might lead to the worst case performance scenarios within a reasonable time frame.

We demonstrate the effectiveness of our approach by using our framework to analyze the worst case performance, in terms of throughput, energy efficiency and fairness, of IEEE 802.11 for adhoc networks. Using our framework we generate library of scenarios that cause worst performance. For example, we generate scenarios where all nodes in the network suffer from zero throughput resulting in zero network throughput. Some of our scenarios degrade throughput and energy efficiency by a factor of 12 and 8, respectively, when compared to randomly generated scenarios. Empirical analysis shows that the complexity of our algorithms is practical.

The class of protocols we aim in developing our framework is single channel wireless MAC protocol based on CSMA/CA schemes. We propose extensions of our basic framework to incorporate the performance analysis of the extensions of CSMA/CA protocols for quality of service and power control. We perform initial analysis of the case studies on each of the extensions and identify previously unknown issues and protocol problems. Based on the analysis of the identified problems, we propose classes of protocol modifications and design of new protocols that are expected to improve the performance. The extensions of our framework and the respective case studies exhibit the robustness as well as importance and application of our framework to a broader class of MAC protocols.

1 Introduction

In the OSI reference model, medium access is the function of the layer 2 sub-layer called the Medium Access Control (MAC) layer. Sharing a limited communication bandwidth efficiently among all nodes in the network is the main objective of wireless MAC protocols. A wireless MAC protocol must address the hidden terminal and exposed terminal problem in order to meet its main objective. Wireless network can be generally classified into two: (1) centralized, and (2) distributed networks. In centralized (infrastructure) wireless networks, a central entity, or base station (BS), acts as an interface between the wireless and infrastructure wireline networks as well as assigns time slots (single channel protocols) or channels (multiple channel protocols) among all nodes for efficient channel utilization of the wireless network. In the distributed (infrastructure-less) wireless network, there is no central administration and all nodes in the network achieve efficient channel utilization in a distributed manner. The scope of our performance evaluation framework is in the distributed wireless network. However, this is not a limitation as we focus on distributed wireless network because it is more challenging due to absence of a central administration.

We divide the distributed wireless networks into two classes: (1) wireless adhoc networks, and (2) wireless sensor networks. The common objectives of MAC protocols designed for wireless adhoc and sensor networks are to achieve efficient channel utilization and power (energy) efficiency. Wireless adhoc network is required where a fixed communication infrastructure, wired or wireless, does not exist or has been destroyed. The goal of the network is to allow a group of communicating nodes to setup and maintain connection among themselves without the support of a base station. In recent years, the adhoc networks have been very useful in coordination of large scale emergencies, crisis response, military applications, and conference meetings. Medium access control (MAC) is the main determinant how efficiently and fairly a network utilizes the scarce wireless medium. Sharing a limited bandwidth efficiently and fairly among all nodes are hence the main objectives of MAC protocols for wireless adhoc networks. Protocols have been designed to address these issues, however, very few of them have been tested systematically for their performance. Providing a test generation framework for worst case performance evaluation of a broad class of protocols motivates this work.

Recent advances in technology have empowered sensor devices with certain processing, memory and communication capabilities. Networking these sensor devices serve a wide varieties of applications including environmental, structural, factory and seismic monitoring, and target tracking. In many applications, a sensor node can operate as long as its battery. Therefore, achieving high channel utilization as well as energy efficiency have been the major goals of MAC protocols designed for wireless sensor networks. We can use our framework in the context of sensor network as well. Furthermore, we plan to incorporate topology synthesis in our framework using which we expect to generate meaningful and important results in sensor MAC protocols.

Test generation (TG) is mainly based on search techniques that search for valid sequences of protocol events that expose weaknesses or errors in the design of a protocol. Traditional test generation approaches target verification and are based on forward search methods where the entire search space is exhaustively enumerated for test scenarios [1, 2]. Formal verification approaches use high-level system description languages to model and to analyze network protocols [6, 7] and practical systems [5], and involve determining the set of all reachable states of the models.

We propose a test generation framework that, instead of adopting the validation approach, uses a "falsification approach" and directly targets the protocol conditions that adversely effect the protocol performance objective. Our main formalism is that of a finite state machine (FSM) which we use to represent protocol events and states of network nodes. We also integrate time relations between the events and the state transitions into the abstraction. A **scenario** is defined in terms of network node states, protocol events and time relations between the events and the state transitions. If one or more components of a scenario are partially specified, it is called a **partial scenario**. Given a protocol performance objective (e.g., throughput) and the expression that defines it, we first identify our study objective (e.g., minimize throughput) and a set of target events that meet our study objective. We then use error generation algorithms to generate a set of conditions and protocol transition rules that lead to the target events. We apply heuristic to select the condition that maximizes our study objective, and refer to the condition as the **target error**, which is specified as a partial scenario. We then use a mix of forward and backward search and implication techniques to generate test scenarios that can create the target error. We formulate this as a branch-and-bound search. We use heuristics to prioritize alternatives during this step and typically generate the worst scenario at practical run-time complexity. Finally, we use extensive network simulations to validate the test scenarios as well as to evaluate the performance for realistic applications.

Our proposed performance evaluation framework is applicable to all single channel wireless adhoc MAC protocols that use handshaking as the basic mechanism to reserve the channel among all the users [11, 13, 14, 15, 16, 17, 18]. We have applied our method to several well-known protocols, e.g., IEEE 802.11 [11], MACA [13] and MACAW [14]. We only present the results for IEEE 802.11b and MACAW in this proposal. Using our framework we have generated test scenarios in which network throughput degrades by a factor of 12 compared to random scenarios. In the scenarios generated for worst fairness, some nodes in the network starve while others achieve very high throughput, leading to short-term unfairness in IEEE 802.11 networks. Short-term fairness is important in many contexts, e.g., smooth acknowledgment flow for TCP and low jitter for real-time audio and video applications [21]. The scenarios generated for worst energy efficiency exhibit degradations by a factor of 8 compared to random scenarios. The scenarios we generate are likely to arise due to a pattern of synchronized transmission events and severely affect the performance of these applications.

Our basic framework is designed for the worst case performance evaluation of single channel wireless MAC protocols based on CSMA/CA schemes. The class of these protocols usually uses handshaking as the basic chan-

nel access mechanism. The extension of the basic CSMA/CA scheme to incorporate the quality of service as well as power control has been recently studied [30, 33, 34, 35, 36, 37, 38, 39]. We propose to extend our basic framework to carry out performance analysis of variations of CSMA/CA protocols that consider quality of service and power control. Based on an initial analysis, we propose a systematic approach for protocol modification and design of new protocols. In particular, we propose classes of protocol modifications to CSMA/CA protocols for power control.

The main contributions of this work are as follows. First, we present a representation of a system that uses a **partial scenario** to describe protocol events, network node states and their temporal relations. This representation is semantically general for a broad class of protocols. The novelty of our algorithms is in its use of implication rules in specifying the partial scenario and identifying the components to prune an invalid scenario early in the search process. Second, using our framework we generate scenarios that lead to very low throughput and extreme unfairness in the protocol under study. Such extreme behaviors of a protocol can not be exposed using approaches based on analytical methods or random simulations. Our systematic test generation approach provides a library of scenarios for simulation that expedite the process of testing various mechanisms throughout the evolution of a protocol. Finally, the extensions of our basic framework in broader classes of protocols exhibit the richness of our framework as well as its robustness.

The rest of the proposal is organized as follows. Section 2 presents related work. Section 3 describes an overview of our proposed performance evaluation framework. The main two components of our framework, namely, error generation algorithms and test scenario generation algorithms are presented in Sections 4 and 5, respectively. The case studies on IEEE 802.11b and MACAW are presented in Section 6. In Section 7 we analyze our framework. Section 8 presents a detailed description of our proposed research. Section 9 presents a summary of the proposal and outlines the proposed future research.

2 Related Work

Conformance testing deals with verifying that a specific implementation conforms to a specification, whereas our testing technique evaluates a design (as opposed to an implementation). For brevity we refer only the most related and recent literature on the formal verification of network and communication protocols. The main formalism of these works are that of a *probabilistic timed automata* which is an extension of *timed automata* [3]. **PRISM** [4] is a tool for the automatic formal verification of probabilistic systems that uses a high-level system description language to exhaustively analyze the set of all reachable states. It has been used for model checking and for accurate computing of numerical properties of network protocols [6, 7] as well as practical systems [5]. [6] uses PRISM to verify properties of IEEE 802.11 wireless LAN referring both to the likelihood of repeated transmission collision and to the probability that a node sends packet correctly within a certain deadline. Our work is best positioned with [6] which formally verifies certain properties of IEEE 802.11, whereas we provide a library of scenarios that exposes the worst performance of the protocol. Both these approaches are search based, and therefore are exhaustive. However, we use implication algorithms to reduce the search space by eliminating invalid branches early in the search process and heuristics to guide the search to generate worst performance scenarios at practical complexities. Finally, our framework can be extended to synthesize topology, whereas in formal verification approaches, topology must be an input.

Our work is based on STRESS [8] which provides a framework for protocol design and verification. STRESS uses search techniques to synthesize test scenarios given a protocol description, and its correctness criteria. The result is a set of network topologies, network failures, and protocol event sequences that violate the correctness criteria. One of the approaches STRESS uses is FOTG (fault oriented test generation)[9] in which the low level fault (e.g., packet loss) is modeled in the form of correctness criteria. Our approach is **error oriented**. An error is a high level (e.g., MAC layer) anomalous behavior and is defined in terms of protocol events, network node states and time relations between protocol events and state transitions. Our approach is closest to FOTG with the following major differences:
(1) differences in the model: we adopt an atomic representation of protocol messages and states with the semantics of time in order to model behaviors (e.g., collision) at MAC layer that leads to the high level **partial scenario**, whereas, FOTG adopts a non-atomic abstraction with no semantics of time,
(2) differences in algorithms: we use implication algorithms that are used to deal with prohibited entries in a

partial scenario, and
(3) differences in application: incorporation of semantics of time enriches our TG framework to deal with **partial scenarios** and for the first time allows use of the approach to generate scenarios for wireless network protocols.

[10] presents a survey of 34 MAC protocols for wireless adhoc networks ranging from industry standards to research protocols. Channel access and separation is one of the features on which these protocols are classified. Our framework is designed for single channel MAC protocols where the medium is shared between all nodes in the network for both data and control packets. Handshaking is the basic access mechanism in this class of protocols that includes IEEE 802.11 [11], MACA [13], MACAW [14], FAMA [15], MACA-BI [16], PS-DCC [17], RIMA-SP [18], and so on. Our approach is applicable to all these protocols, however, as we present detailed results primarily for IEEE 802.11b, we refer to the work related to this protocol.

A number of studies have been conducted to analyze average performance of IEEE 802.11b. To the best of our knowledge, no previous approach systematically generates scenarios that expose worst-case performance - which is what we enable. Using a *markov chain* model for performance analysis [20] concludes that though the protocol is robust under moderate condition of hidden terminals and node mobility, the performance (e.g., throughput) degrades under extreme condition of hidden terminals and node mobility. The fairness issues we have identified by analyzing the results of our test generation framework are closest to the fairness issues addressed in [22]. However, [22] discusses the short-term fairness of IEEE 802.11b in wireless LAN (WLAN or infrastructure mode), whereas we address the same problem in the context of adhoc network (i.e., infrastructure-less mode). Among the causes of unfairness identified in [22], we show that silent drop of RTS packets is the dominant one.

We perform initial analysis on the case studies of our proposed extensions for quality of service and power control. IEEE 802.11e [30] is an industry standard that is an extension of the legacy 802.11b to provide quality of service. [31] shows that the protocol provides quality of service by assuring that the higher priority class achieves more throughput compared to a lower priority class. We generate scenarios in which a lower priority class achieves more throughput compared to a higher priority class. We also perform initial analysis on the case studies of the proposed extensions for transmission power control (TPC) and directional antennas. There has been significant amount of work done in previous [30, 33, 34, 35, 36] that has identified various issues of the basic TPC protocols. However, to the best of our knowledge, the results and issues regarding the throughput and fairness of the TPC protocols that we have identified have not been addressed or identified by any previous work. The issues we identified in protocols that use directional antennas for power control are the same as the issues identified in [39].

3 Framework Overview

Given a wireless MAC protocol or a set of underlying schemes, our objective is to evaluate the worst case performance of the protocol with respect to a given set of protocol performance metrics. Figure 1 presents the block diagram of our performance evaluation framework. The framework is composed of three main blocks: (1) Error generation algorithms, (2) test generation algorithms, and (3) simulation. Inputs to the framework are: (1) the protocol under study, (2) an objective function in terms performance of the protocol, and (3) a network topology. Given a protocol performance objective function, for example, throughput, the error generation algorithms use the protocol transition table to generate a set of **wanted transitions** and **wanted conditions** that meet our study objective. These transitions and conditions as well as the topology are input to the test generation algorithms which generate a set of test scenarios that maximize or minimize our objective. The test scenarios are then simulated in a simulation environment (we use ns-2) to achieve the actual performance provided by our scenarios.

Following is a list of key performance metrics of MAC protocols for wireless adhoc networks that we study using our framework. Wireless MAC protocols try to optimize one or more of these metrics, sometimes compromising others.

Throughput: Throughput is defined as the fraction of time the channel is used to successfully transmit payload bits [27]. Let ρ be the amount of payload (data) successfully transmitted in time γ, then **throughput** is defined as follows:

$$throughput = \rho/\gamma. \tag{1}$$

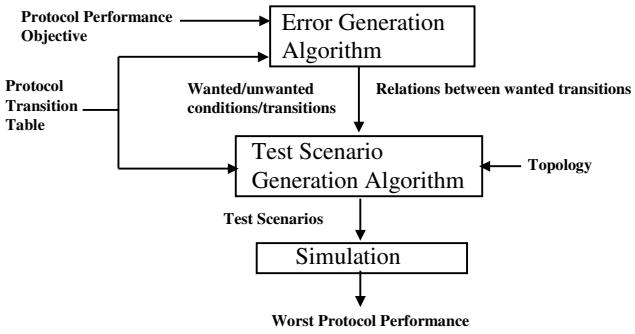
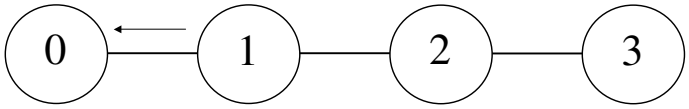


Figure 1: An overview of our framework.



$G = \{ G_0 = \{1\}, G_1 = \{0, 2\}, G_2 = \{1, 3\}, G_3 = \{2\} \}$

Figure 2: Topology I: A wireless network of 4 nodes.

Energy efficiency: Energy efficiency is defined as the number of packets delivered per unit energy spent [29].
Short-term fairness: Short-term fairness of a protocol refers to its ability to provide equitable access to all the open connections over short period of time [28].

Given a protocol performance metric, for example, throughput, our study objective is to analyze the worst case performance by generating the set of scenarios that lead to the worst case value for the metric. We achieve this objective using the following steps. First, we transform the protocol performance metrics in a form that reflects our study objective. For example, given throughput, we transform it into a **protocol condition** that we minimize or maximize in later steps in order to generate scenarios that lead to minimum throughput in a given topology. We refer the **conditions** that meet our study objective as **wanted conditions** or **errors**. The **error generation algorithm** transforms the protocol performance objective into wanted conditions. Second, given a set of wanted conditions and a topology, we use **test generation algorithm** to generate scenarios that maximize or minimize our study objective, which in turn, minimize (or maximize) the given protocol performance metric. Section 3.1 presents a brief description of our basic models.

3.1 Base Models

Our overall model consists of (1) a network topology model, (2) models of network node states and protocol messages, (3) a protocol model and (4) model of a scenario.

(1) Network topology: Topology is modeled in terms of transmission range of each node in the network. Transmission range of node i is a set G_i containing the nodes who hear its transmission. Figure 2 presents a wireless network of 4 nodes where $G_0 = \{1\}$, $G_1 = \{0, 2\}$, $G_2 = \{1, 3\}$, and $G_3 = \{2\}$.

(2) Models of network node states and protocol messages: We need to model each protocol message in a way that captures meaningful state transitions the message causes. Transmission of a message m effects the state of transmitter while reception of m effects the states of intended receiver and the receivers that overhear the message. Therefore, we decompose a protocol message into its transmission and reception. A node transmitting the message m changes its state when it starts the transmission, remains in the state as long as it transmits and changes its state again when it finishes the transmission. Therefore, we further decompose transmission and reception into corresponding *start event* and *end event*. Semantics of an event e from source node i to destination node j is: $e_{i,j}(\gamma)[\sigma]$ where e is the event ID, γ represents its relative timestamp which is the delay of e from the

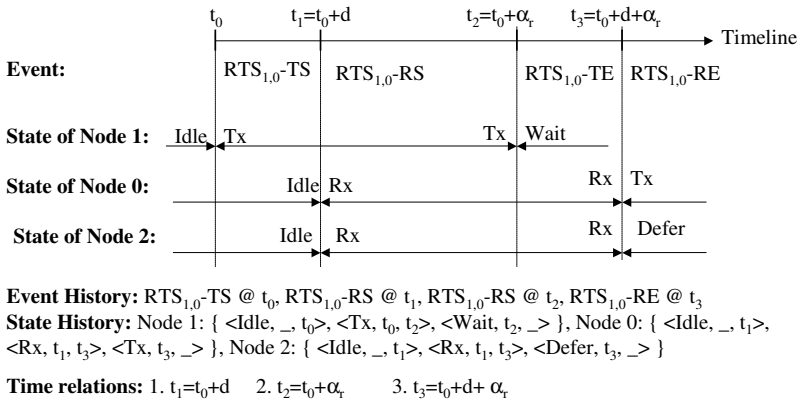


Figure 3: Semantics of a protocol message.

transition that creates e, and σ represents the set of nodes affected by e. The state s of a node i during period $[t_u,t_v]$ is denoted by $i: <s, t_u, t_v>$.

For example, an RTS (Request-To-Send) message from node i intended for node j is modeled as following events: $RTS_{i,j}$-Transmit-Start ($RTS_{i,j}$-TS) at time t_0, $RTS_{i,j}$-Transmit-End ($RTS_{i,j}$-TE) at $t_2 = t_0+\alpha_r$, $RTS_{i,j}$-Receive-Start ($RTS_{i,j}$-RS) at $t_1 = t_0 + d$ and $RTS_{i,j}$-Receive-End ($RTS_{i,j}$-RE) at $t_3 = t_0 + d + \alpha_r$. d denotes message propagation delay and α_r denotes the time it takes to transmit/receive an RTS message. Figure 3 illustrates the semantics of RTS events and corresponding state transitions for topology I (Figure 2). At time t_0 node *1* initiates RTS/CTS exchange with node *0*. The events are represented as: $RTS_{1,0}$-TS(0)[{1}], $RTS_{1,0}$-RS(d)[G_1], $RTS_{1,0}$-TE(α_r)[{1}] and $RTS_{1,0}$-RE($d+\alpha_r$)[G_1]. Node *1* changes its state from *Idle* to *Tx* at t_0 when it starts transmitting the RTS message. At t_1, nodes 0 and 2 start receiving RTS message and change their states from *Idle* to *Rx*. At t_2, node 1 finishes its transmission and changes its state from *Tx* to *Wait*. Node 0 finishes receiving the message at t_3 when it changes its state from *Rx* to *Tx*. Node 2 changes its state from *Rx* to *Defer* at t_3.

The state of a node is one of the following two types: (1) single state, or (2) *combined* state. In single state, a node performs only one function while in *combined* state it performs more than one function. For example, *Rx* is a single state while *WCTSARx* (wait-for-CTS-and-receive) is a combined state in which a node waits and receives at the same time. A protocol event is one of the following two types: (1) non-timer, and (2) timer. A timer event is an event for which a corresponding timer is running within a node. We model two types of timers: (1) *non-suppressive* timers, and (2) *suppressive* timers. A suppressive timer is a wait timer which is scheduled to wait for an event and is suppressed when the event occurs. We refer to network node state or protocol event in a scenario using a common term *entity*.

(3) Protocol Model: We use the above notion of node state and protocol event to model the protocol. The protocol is specified using a transition table F. Each row of transition table defines state transitions of network nodes for a protocol event. Semantics of an entry in the transition table is given by: $<s_{in}, e_{j,k}, s_{out}, \{ e1_{j,l1}(\gamma_1)[\sigma_1], e2_{j,l2}(\gamma_2)[\sigma_2] ,... \} >$. It describes the following transition: event $e_{j,k}$ at any given time t changes the state of node i, where i is either the same as j or $i \in G_j$, from s_{in} to s_{out} and triggers events $e1_{j,l1}$ at time $t+\gamma_1$, $e2_{j,l2}$ at time $t+\gamma_2$ and so on. $e1_{j,l1}$ effects a set of nodes σ_1, $e2_{j,l2}$ effects σ_2 and so on.

(4) Model of a Scenario: A **scenario** is defined by the following: history of network node states (denoted by H_s), history of protocol events (H_e), time relations or system of time inequalities (SOI), and prohibited lists. The first three describe the scenario while the prohibited lists describe network node states (PL_s) and protocol events (PL_e) that should not occur as well as the time relations (PL_{SOI}) between prohibited and history entities. The scenario in Figure 4 presents a description of collision at node 1 in topology I. In this scenario, node 1 is in state *BOCOL* (backoff on collision) during interval [t_0, t_1] where the relation between the times, $t_0 < t_1$ is specified in SOI.

We describe our algorithms using a protocol P. Tables 1, 2 and 3 present the description of some states, events

and time variables of the protocol, respectively. Event types *NT*, *ST*, and *NST* denote non-timer, suppressive-timer, and non-suppressive-timer, respectively. Note that *Idle* is an initial state and *Pkt* is an initial event. Among the states, *WCTSARx* is a combined state. *WCTS* timer is a suppressive timer as it gets suppressed by reception of the corresponding *CTS*. Table 4 presents a part of transition table of the protocol *P*. It works as follows: upon receiving a packet from higher layer for node *j*, node *i* transmits an *RTS* destined for node *j*, schedules a *WCTS* timer to wait for the *CTS* from *j*. Upon receiving the *RTS* destined for it, node *j* transmits a *CTS* (Clear-To-Send) to node *i*. On overhearing the *RTS*, a node *k* defers access to the channel by scheduling a timer for a period of θ_r. If node *i* receives the *CTS* from *j* before its *WCTS* timer expires, it goes to *Tx* state to transmit the data, otherwise, it goes to backoff-on-failed-transmission (BOFT) state after which it retransmits the *RTS*. The type of a timer can be determined from the transition table.

Table 1: States of protocol P.

State name	**State description**	**Type**
Idle	Does nothing	Initial
Tx	Transmitting	Single
Rx	Receiving	Single
WCTS	Waiting for CTS	Single
Defer	Deferring access to channel on RTS	Single
WCTSARx	WCTS and receiving	Comb.
BOCOL	Backoff on collision detection	Single
BOFT	Backoff on failed transmission	Single

Table 2: Events of protocol P.

Event name	**Event description**	**Type**
Pkt	Packet from higher layer	Initial
RTS-TS	RTS Transmission Start	NT
RTS-RS	RTS Reception Start	NT
RTS-TE	RTS Transmission End	NT
RTS-RE	RTS Reception End	NT
CTS-TS	CTS Transmission Start	NT
CTS-RS	CTS Reception Start	NT
CTS-TE	CTS Transmission End	NT
CTS-RE	CTS Reception End	NT
WCTST-TS	Wait for CTS Timer Start	ST
WCTST-TE	Wait for CTS Timer End	ST
DeferT-TS	Defer Timer Start	NST
DeferT-TE	Defer Timer End	NST
BOCOLT-TS	BOCOL Timer Start	NST
BOCOLT-TE	BOCOL Timer End	NST
BOFTT-TS	BOFT Timer Start	NST
BOFTT-TE	BOFT Timer End	NST

Table 3: Time variables of protocol P.

Time var.	**Time var. description**
α_r	Time to transmit/receive RTS
α_c	Time to transmit/receive CTS
α_a	Time to transmit/receive ACK
d	Propagation delay
θ_r	Defer period on RTS
w_c	WCTS period
B_c	Backoff period

4 Error Generation Algorithm

Figure 5 presents the block diagram of error generation algorithm. Input to the algorithm is a function or expression of the protocol performance metric under study. It has two parts. The first part identifies or generates a set of target events from the input performance objective function/expression using the protocol transition table **F**. The second part uses the transition table to generate a set of conditions from the target events that meet our study objective. We define these conditions as **error conditions** or **wanted conditions**. While generating

Table 4: Transition table of protocol P.

	Start state	Input event	End state	Output event
1	Idle	$\text{Pkt}_{i,j}$	Tx	$\text{RTS-TS}_{i,j}(0)[\{i\}]$, $\text{RTS-RS}_{i,j}(d)[\text{G}_i]$, $\text{RTS-TE}_{i,j}(\alpha_r)[\{i\}]$, $\text{RTS-RE}_{i,j}(d+\alpha_r)[\text{G}_i]$
2	Tx	$\text{RTS-TE}_{i,j}$	WCTS	$\text{WCTST-TS}_{i,i}(0)[\{i\}]$, $\text{WCTST-TE}_{i,i}(w_c)[\{i\}]$
3	Idle	$\text{RTS-RS}_{j,i}$	Rx	
4	Rx	$\text{RTS-RE}_{j,i}$	Tx	$\text{CTS-TS}_{i,j}(0)[\{i\}]$, $\text{CTS-RS}_{i,j}(d)[\text{G}_i]$, $\text{CTS-TE}_{i,j}(\alpha_c)[\{i\}]$, $\text{CTS-RE}_{i,j}(d+\alpha_c)[\text{G}_i]$
5	WCTS	$\text{CTS-RS}_{j,i}$	WCTSARx	
6	WCTSARx	$\text{CTS-RE}_{j,i}$	Tx	$\text{Data-TS}_{i,j}(0)[\{i\}]$, $\text{Data-RS}_{i,j}(d)[\text{G}_i]$, $\text{Data-TE}_{i,j}(\beta)[\{i\}]$, $\text{Data-RE}_{i,j}(d+\beta)[\text{G}_i]$
7	Idle	$\text{RTS-RS}_{j,k}$	Rx	
8	Rx	$\text{RTS-RE}_{j,k}$	Defer	$\text{DeferT-TS}_{i,i}(0)[\{i\}]$, $\text{DeferT-TE}_{i,i}(\theta_r)[\{i\}]$
9	WCTS	$\text{WCTST-TE}_{i,i}$	BOFT	$\text{BOFTT-TS}_{i,i}(0)[\{i\}]$, $\text{BOFTT-TE}_{i,i}(B_c)[\{i\}]$
10	Rx	$\text{RTS-RS}_{j,i}$	BOCOL	$\text{BOCOLT-TS}_{i,i}(0)[\{i\}]$, $\text{BOCOLT-TE}_{i,i}(B_c)[\{i\}]$
11	BOFT	$\text{BOFTT-TE}_{i,i}$	Tx	$\text{RTS-TS}_{i,j}(0)[\{i\}]$, $\text{RTS-RS}_{i,j}(d)[\text{G}_i]$, $\text{RTS-TE}_{i,j}(\alpha_r)[\{i\}]$, $\text{RTS-RE}_{i,j}(d+\alpha_r)[\text{G}_i]$

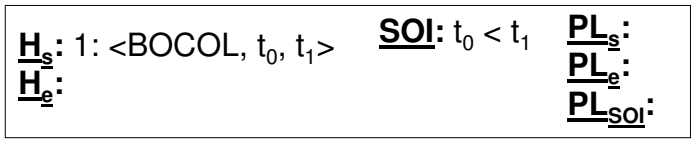


Figure 4: A scenario describing a collision.

the wanted conditions, the algorithm generates a set of transitions (called **wanted transitions**) and the relation between these wanted transitions. We first present the terminologies and a high level description of the algorithms. We use the expression of throughput in Equation 1 as an example.

Target event: The first step of the error generation algorithm is to identify the target events based on the protocol performance objective function or expression using the protocol transition table.

Wanted conditions and transitions: Given the target events, we define the transitions necessary to meet our study objective as wanted transitions. The associated conditions are defined as wanted conditions.

Unwanted conditions and transitions: Given the target events, we define the transitions that nullify our study objective as unwanted transitions. The associated conditions are defined as unwanted conditions.

Consider an example where our study objective is to generate scenarios to minimize throughput. Target events are identified as *successful data reception* and *successful acknowledgment reception* based on ρ and γ in Equation 1, respectively. The transitions necessary to trigger these target events nullify our study objective (instead of meeting the study objective). Therefore, the transitions necessary to trigger the target events, in this case, are in fact the unwanted transitions. Thus, the transitions necessary to trigger the target events either meet our study objective, or nullify it. In the first case, we generate the wanted transitions directly. In the second case, we generate the wanted transitions indirectly through the unwanted transitions. Wanted conditions are the output

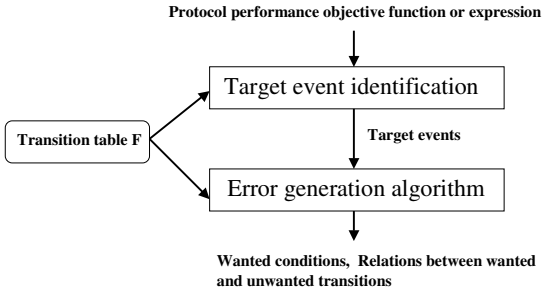


Figure 5: Block diagram of error generation algorithm.

Performance metric:
Throughput = α / β

(a) Input to target event identification

Target event:
Data-RE

(b) Output of target event identification

Wanted transitions: Successful Data (S-Data, row 6), S-CTS (row 4), S-RTS (row 1)

Unwanted transitions: U-RTS, U-CTS, U-Data, D-RTS

(c) Wanted and unwanted transitions

Wanted Transition	Description	Precondition/ Unwanted condition	Weight	Wanted Condition	
				Transmitter	Receiver
U-RTS	Unsuccessful RTS	Pkt	1	BOFT	BOCOL
U-CTS	Unsuccessful CTS	S-RTS (Successful RTS)	2	BOCOL	BOFT
U-Data	Unsuccessful Data	S-RTS, S-CTS	3	BOFT	BOCOL
D-RTS	Silent drop of RTS	Pkt	4	BOFT	Defer

(d) Output of error generation algorithm

Figure 6: Example of error generation using protocol P.

states of the wanted transitions as these are the conditions that meet our study objective.

4.1 Algorithm Details

The error generation algorithm has two parts as presented in Figure 5. Given a protocol performance objective function, the target event identification is currently a manual step in which the protocol designer identifies a set of events that either meet the study objective or nullify it. If the target events nullify the study objective, the automated error generation algorithm directly generates the unwanted transitions. Otherwise the error generation algorithm directly generates the wanted transitions. The error generation algorithm also generates a relation between the wanted and unwanted transitions. In this section, we present a 3-step algorithm for the case when the target events nullify our study objective. The other case is more straightforward.

1. **Generate unwanted transitions from target event:** Given a target event, the algorithm creates its predecessors using transition table until it reaches the initial event. As the target event nullifies our study objective, each of the transitions (rows of the transition table) thus visited is *unwanted* for our study objective. All unwanted transitions for all target events are combined to obtain the set of all unwanted transitions.

2. **Generate wanted transitions from an unwanted transition:** Given an unwanted transition, the algorithm generates all wanted transitions using three steps. First, the algorithm uses backward implication to generate the precondition of the unwanted transition. It then uses forward implications to generate *all* transitions caused by the precondition. Note that the first and second steps respectively use backward and forward implications. Therefore, the unwanted transition is one of the transitions generated in second step. In the third step, the algorithm outputs all transitions generated in second step, except the unwanted transition, as wanted transitions.

3. **Generate wanted conditions:** Given a wanted transition, the output state of the transition is the wanted condition. Given a set of target events, the algorithm thus identifies *all* wanted conditions which are the output states of *all* wanted transitions.

Figure 6 presents an example of error generation algorithm using protocol P. Let our study objective be to minimize throughput. Inputs to the algorithm are the transition table of the protocol P (Table 4) and the performance metric of interest, namely throughput (Figure 6.(a)). The target event is identified as *successful data*

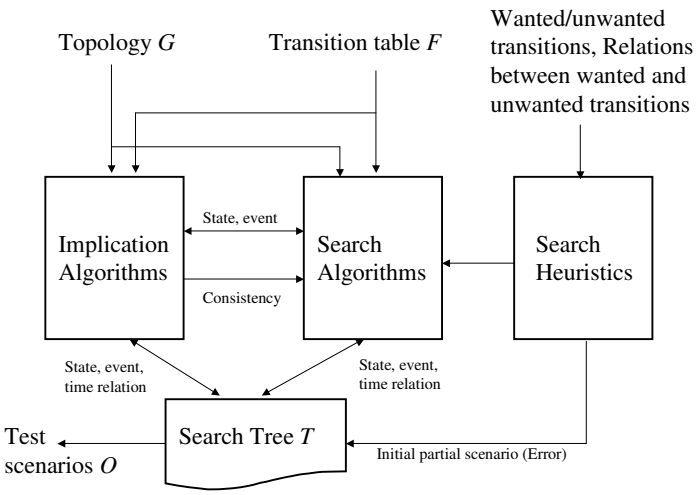


Figure 7: The proposed test generation algorithm.

reception or Data-RE (Figure 6.(b)). In this case, the target event identified nullifies our study objective. Hence, the algorithm first generates the unwanted transitions and then generates the wanted transitions. Figure 6.(c) presents these transitions. For example, S-Data (successful Data), S-CTS and S-RTS are generated as unwanted transitions all of which must be satisfied to trigger the target event Data-RE. U-RTS (unsuccessful RTS), U-CTS and D-RTS (drop of RTS) are generated as wanted transitions any of which is sufficient to meet our study objective of minimizing throughput. Figure 6.(d) presents the output of our error generation algorithm. The table in Figure 6.(d) presents all wanted transitions with their associated unwanted and wanted conditions. The column denoted by **Weight** in Figure 6.(d) represents weight of a wanted transition in meeting our study objective. We discuss the weight of a wanted transition in Section 5.3. Note that in this example, the error generation algorithm generates three wanted conditions, namely, *BOCOL*, *BOFT* and *Defer*. Any of these conditions must be satisfied in order to meet our study objective. In Section 5.3 we explain how we apply heuristics in choosing *the* wanted condition that is deemed more likely to lead to the scenario that meets the study objective.

5 Test Generation Algorithm

Figure 7 presents the block diagram of our test generation framework. Inputs to the framework are the transition table F, network topology G, the set of wanted conditions and transitions, and the relations between the wanted and unwanted transitions. The last two components are outputs of error generation algorithm. The test generation algorithm outputs test scenario that meets our study objective. The framework has three building blocks: (1) search, (2) implication, and (3) heuristics. Search algorithm enumerates child nodes and creates and maintains the search tree. Implication algorithm derives node state and protocol event history which can be uniquely determined from the states, events and time relations represented by a tree node and checks consistency of the tree node. Inconsistent tree nodes are pruned. The implications are used to reduce the complexity of search. Nevertheless, worst-case runtime complexity of any search algorithm that generates **all scenarios** (among which we need to find the worst case) is always impractically high. For this reason, we develop a set of heuristics to increase the likelihood that the search algorithm generates the worst case scenario in practical time.

We present an overview of the test generation algorithm in this section. Details of search, implication and heuristics are presented in the Sections 5.1, 5.2, and 5.3, respectively. An illustrative example using the protocol P is presented in Section 5.4. Given a set of wanted conditions and transitions, we first convert them to a partial scenario that represents a network condition in the given topology. In this step, we apply heuristics to guide the search to generate conditions to meet our study objective. We then apply search and implications that either lead to the worst (or best) case scenario or prune because of inconsistency. Our search and implication algorithms work as follows to generate the worst scenario. Given an entity in a partial scenario, we attempt to derive unique information using backward and forward implications of the entity. Since the derived information might already

exist in the scenario (due to some other causal relations), we check whether the newly generated information already exists in the scenario. We add the information in the scenario if we can confirm that it does not already exist in the scenario. While adding the information to the scenario, we check for any conflict or inconsistency between existing and newly added information. If a conflict exists, we prune the scenario. Otherwise, given an entity, we check if any of its predecessors exist in the scenario. Existence of a predecessor justifies the entity in the scenario. If the entity does not have any predecessor in the scenario, we must justify it. To do so, in our search tree, we create child nodes of the scenario by creating every possible choice of the predecessor for every unjustified entity in the scenario. Thus the process of implication and enumeration continues until all entities in the scenario are justified. In the process of exploring choices in search, we use heuristics to reduce the search space to meet the study objective. Details of search and implication procedures are presented in our technical report [24].

5.1 Search Algorithms and Concepts

Predecessor of an entity: If entity y is sufficient to create an entity x, entity y is said to be a predecessor of x. Note that an entity (other than an initial state or an initial event) may have one or more predecessors.
Justification of entities: An entity x in tree node T_n is **justified** if at least one of its predecessors exists in the scenario T_n. The entity is **unjustified** if none of its predecessors exists in the scenario. If all entities of a scenario are justified, it is called a **fully specified scenario**. If one or more entities of the scenario are unjustified, it is called a **partial scenario**.
Checking existence of an entity in a scenario: Two events are said to be *incompatible* if their event IDs are different, or sources are known and different, or destinations are known and different. For example, RTS-$RS_{1,0}$ and RTS-$RS_{1,unknown}$ are compatible, but RTS-$RS_{1,0}$ and RTS-$RS_{1,2}$ are incompatible. Two states are *compatible* if their node IDs as well as state IDs are identical. Given two compatible entities, y at time t_y and z at t_z, where z exists in scenario T_n, the existence of y in T_n (with respect to z) can be determined using time stamp test **time-stamp-test** presented in Table 5. Procedure **check-existence** checks the existence of an entity y in a scenario T_n. The procedure first finds all entities of T_n that are compatible to entity y. It then uses the **time-stamp-test** with each of these compatible entities to determine the existence status as "Yes", or "No", or "May-exist" in which case it returns the choices which could potentially be identical to entity y.

Table 5: **time-stamp-test** to test existence of y w.r.t. z.

$\mathbf{t}_y$=$\mathbf{t}_z$ **Test**	$\mathbf{t}_y$<$\mathbf{t}_z$ **Test**	$\mathbf{t}_y$>$\mathbf{t}_z$ **Test**	**Output**
Unsolvable	don't care	don't care	y New
Solvable	Unsolvable	Unsolvable	y Old
Solvable	Unsolvable	Solvable	y May-be-old
Solvable	Solvable	Unsolvable	y May-be-old
Solvable	Solvable	Solvable	y May-be-old

Lemma 1. *Test* **time-stamp-test** *accurately determines whether entity entity y at t_y is identical to a compatible entity z at t_z that exists in a given scenario T_n.*

Proof. Assume that z is the only entity that exists in T_n and is compatible to y. Therefore, for z to be identical to y, t_y must be the same as t_z, leading to a solution for $\mathbf{t}_y$=$\mathbf{t}_z$ test. Furthermore, for y to be determined uniquely as being identical to z, no solution must exist to $\mathbf{t}_y$<$\mathbf{t}_z$ and $\mathbf{t}_y$>$\mathbf{t}_z$ tests. A solution either to $\mathbf{t}_y$<$\mathbf{t}_z$ or $\mathbf{t}_y$>$\mathbf{t}_z$ test leads to inconclusiveness. Now, consider the case when $\mathbf{t}_y$=$\mathbf{t}_z$ test does not have a solution. The non-solution of the test uniquely determines that entity y is not identical to entity z. □

Enumeration procedure: Given an entity x in T_n, we can use procedure **create-pred** to create P_x, predecessor set of x and then check existence of each of these predecessors using procedure **check-existence**. If at least one of the predecessors in P_x exists in the scenario, x is deemed justified. If none of the predecessors of P_x exists in the scenario, x is deemed **unjustified**. Given a scenario T_n, we consider every unjustified entity in the above manner to create its child nodes in our search tree. Procedure **enumerate** takes the cross product between predecessor sets of all unjustified entities to create child nodes of T_n.

Lemma 2. *Given a tree node T_n, procedure* **enumerate** *enumerates all child nodes rooted at T_n.*

Proof. Given an entity x in T_n, procedure **create-pred** creates all its predecessors from the transition table. It directly follows from Lemma 1 that we can determine whether x is justified in T_n accurately. Procedure **enumerate** only considers unjustified entities. Cross product of the predecessor sets of all unjustified entities enumerates all possible combinations of predecessors, and therefore, all child nodes rooted at T_n. □

Test scenario generation procedure: Given a partial scenario, we generate test scenarios using procedure **create-tree**. It applies a depth-first search (DFS) technique and uses procedure **enumerate** to enumerate child nodes. The search continues until (1) we reach at a *leaf node* at which all entities are justified, or (2) the scenario is pruned using implication rules presented in Section 5.2. According to Lemma 2, at every node, the procedure generates all possible predecessors for every unjustified entity. Hence, the search algorithm generates all test scenarios leading to the initial partial scenario. We then compare the generated scenario with a previously generated scenario, using our study objective (e.g., minimize throughput), to identify the scenario that meets our study objective. Generating the worst case scenario using a search algorithm is almost impractical in realistic time. To generate the worst case scenario in a practical time we use heuristics presented in Section 5.3.

Lemma 3. *All entities are justified in a leaf node, i.e., a leaf node of a search tree is indeed a fully specified scenario.*

Proof. Let x be unjustified in leaf node T_n. From the definition of justification, no predecessor of x exists in T_n. Procedure **enumerate** must create its predecessors to enumerate child nodes of T_n that contradicts with the initial assumption that T_n is a leaf node. □

Lemma 4. *Given an error scenario described in E, procedure* **create-tree** *is guaranteed to generate a test scenario if it leads to E. It is guaranteed to prune a scenario if it does not lead to E.*

Proof. See Section 7. □

5.2 Implication Algorithms

Implication is the process of determining network node state and event history that can be uniquely identified as a consequence of a given partial scenario. If an entity x in scenario T_n can uniquely[1] be created from entity y, then **backward implication** of x results y. Entity y is added to the scenario as a result of backward implication. If x uniquely creates entity z, i.e., z can uniquely be created from x, then **forward implication** of x adds z to T_n. We also perform unique implication of combination of multiple entities. For example, the implication rules 5 and 7-9 presented in this section uniquely generate prohibited entities from implication of multiple entities. We use implications to further specify a partial scenario.

While specifying a scenario, our implication rules generate **prohibited entities** that are precluded from the scenario. For example, consider a partial scenario presented in Figure 10.(a) that describes a collision of $\text{RTS}_{2,3}$ and $\text{RTS}_{0,1}$ at node 1 in topology I (Figure 2). In Figure 10.(b), implication of event $\text{RTS-RE}_{2,3}$ generates prohibited entries in PL_e and PL_{SOI} using implication rules we describe in this section. The entries in PL_e are precluded from the scenario as long as they satisfy the time relation presented in PL_{SOI}. In the process of implication, whenever an entity generated to be added to the history (H_s or H_e) is determined to exist in the **prohibited list** (PL_s or PL_e), or vice versa, we prune the scenario. We also prune using a state consistency check presented later in this section. Thus the function of implication is to specify a partially specified scenario as well as to prune inconsistent branches of the search tree.

Following is a complete list of implication rules. Rules 1-4 are used to specify a partial scenario by creating necessary conditions. Rules 5, 7, 8 and 9 are used to generate prohibited entities, and to check their existence in the history in order to eliminate invalid scenarios. Therefore, correctness of these rules ensures the correctness as well as the completeness of our algorithms. For this reason, we present these rules using Lemma 5, 6, 7, and 8, respectively. In [24], we present the procedures to generate the prohibited entities, to check their existence, and to eliminate invalid scenarios according to these rules.

1. Transmission-Reception rule: Reception of a message m implies the transmission of m. A reception-start

[1] The only way in which entity x can be created in T_n is by entity y.

(RS) event implies (backward) a transmission-start (TS) event. Assuming no (explicit) loss[2], a TS event implies (forward) corresponding RS event.

2. Event start-end rule of a message: An end event implies (backward) the start event for the corresponding message. If we encounter a transmission-end (TE) event, it is implied that a transmission-start (TS) event was encountered earlier. Similarly, an RE event implies corresponding RS event. Assuming the wireless nodes do not fail, a TS event implies (forward) a TE event. Note that an RS event does not *always* imply (forward) corresponding RE event because a receiver receiving the message may receive other message causing a collision and hence a garbled reception.

Note that the abstraction of RS event (forward implication, rule 1) assumes that the RS event is triggered even when the corresponding message collides with an ongoing reception. The physical layer indicates the start of a collision to MAC layer whenever it starts receiving another message. In our collision model, we therefore, consider the RS event of the second message to be the start of collision. Consequently, no RE events of the messages under collision are triggered in that case. The RE event of a message, therefore, is modeled as an indication of a successful reception (rule 2).

3. State creation rule: End state.output events ← Start state.input event. This rule states that if node i is in **start state** when it encounters the **input event**, it changes its state to **end state** and triggers the **output events**. This rule is used to create state history from states and events.

4. Rule of neighborhood: When a reception event is encountered by a node j where the transmission is from node i (and j is an element of the set G_i), the same reception event must be encountered by other nodes of the set G_i assuming no explicit message loss.

5. Successful reception rule: A reception of a message in a scenario T_y is confirmed as *successful* when the corresponding RE event is generated in T_n. Note that according to rule 2, forward implication of an RS event does not generate the corresponding RE event. Therefore, an RE event is generated either by backward implication or by **create-pred** procedure, and always is implied as *successful*. We present the implication of a successful reception using Lemma 5.

Lemma 5. *Given a successful reception of a message m defined by m-$RE_{i,j}$ at time t_v at node k and corresponding m-$RS_{i,j}$ at time t_u in a tree node T_y, there must not be any event n-$RS_{p,q}$ at time t_w or n-$RE_{p,q}$ at time t_x in T_y such that n is a message of the protocol, k is an element of the sets G_i and G_p, and $t_u \leq t_w \leq t_v$ and/or $t_u \leq t_x \leq t_v$.*

Proof. Let there exist an event n-$RS_{p,q}$ at time t_w in the scenario T_y during the successful reception interval [t_u, t_v] . From the definition of collision, at time t_w the receiving node k must change its state to the collision state $BOCOL$ leading to a garbled reception that contradicts our initial assumption of successful reception. □

A successful reception of message m from node i to j at node k for a reception interval of [t_u, t_v] implies that there was no other reception in the neighborhood that node k hears during the interval. If any such event does exist in the scenario, we prune the branch. Procedure **succ-rx** generates these prohibited entities in PL_s, PL_e and PL_{SOI}. It also checks existence of these prohibited entities in the scenario T_y. If any prohibited entity exists, the procedure returns to prune the branch.

6. Timer expiration rule: A timer expiration event implies a timer set event. If a timer expires at time t, it is implied that it was set at time t-ω where ω is the period for which the timer was scheduled. While, on the other hand, if a timer is set at time t, it does not imply that it would expire at time t+ω. The forward implication depends on the type of timer. For a non-suppressive timer, forward implication holds as it expires when the scheduled period (ω) is over. In contrast, a suppressive timer may get suppressed before t+ω.

7. Suppressive timer rule: A suppressive timer is scheduled to wait for some event to occur and is suppressed when the event occurs. We present the implication of a suppressive timer expiration using Lemma 6.

Lemma 6. *Given a suppressive timer mT with the timer set event mT-$TS_{i,i}$ at time t_u and the timer end event mT-$TE_{i,i}$ at time t_v at node i in a tree node T_n, there must not be any event $e_{j,i}$ at time t_w in T_n such that e is the event of the protocol for which the timer mT is waiting, $i \in G_j$, and $t_u \leq t_w < t_v$.*

[2] We define the low-level (e.g., physical layer) loss as *explicit* loss. The loss internally incurred at MAC layer is defined as *implicit* loss.

Proof. Let there exist an event $e_{j,k}$ at time t_w in T_n during interval [t_u, t_v] for which the timer was waiting. From the definition of suppressive timer, the timer must suppress at time t_w ($< t_v$) that contradicts the scheduled timer expiration event at time t_v. □

Expiration of timer mT at time t_v, which was set at time t_u for an event $e_{j,k}$ to occur during interval [t_u, t_v], implies that the event $e_{j,k}$ did not occur during the interval. If any such event does exist in the scenario, we prune the branch. Procedure **supp-timer** generates these prohibited entities in PL_s, PL_e and PL_{SOI}. It also checks existence of these prohibited entities in the scenario T_n. If any prohibited entity exists, the procedure returns to prune the branch.

8. Non-suppressive timer rule: Generally in MAC protocols, non-suppressive timers are designed to control the access to the channel. For example, NAV (we refer it as defer timer) and backoff timers are designed to defer access to the channel. We present the implication of a non-suppressive timer using Lemma 7.

Lemma 7. *Given a non-suppressive timer mT defined by timer expiration event mT-$TE_{i,i}$ at time t_v and set event mT-$TS_{i,i}$ at time t_u in a tree node T_n, there must not be any event n-$TS_{i,j}$ at time t_w or any event n-$TE_{i,j}$ at time t_w in T_n such that n is a message of the protocol, j is an element of the set G_i, and $t_u \leq t_w \leq t_v$.*

Proof. Let there exist an event n-$TS_{i,j}$ at time t_w during the interval [t_u, t_v] in T_n. From the definition of non-suppressive timer, the event must have occurred either before the interval or after the interval that contradicts our initial assumption of the event n-$TS_{i,j}$. □

A non-suppressive timer mT at node i for an interval [t_u, t_v] implies that there was no transmission from node i to any of its neighbor during the interval. If any such event does exist in the scenario, we prune the branch. Procedure **non-supp-timer** generates these prohibited entities in PL_s, PL_e and PL_{SOI}. It also checks existence of these prohibited entities in the scenario T_n. If any such entity exists, the procedure returns to prune the branch.

9. Rule of state consistency: The state representation in our model is such that a node remains in only one state (single or combined state) during an interval of time. Therefore, existence of two different states of a node during any interval leads to a state inconsistency. We present this rule as Lemma 8.

Lemma 8. *If a node i is in state s_1 from time t_1 to t_2 and in state s_2 from time t_3 to t_4 such that $s_1 \neq s_2$ (i.e., s_1 and s_2 are two different state symbols), then it leads to inconsistency if these time intervals overlap.*

Proof. If the intervals overlap, the node remains in both states s_1 and s_2 during the period the intervals overlap and therefore contradicts our basic combined state representation. □

Table 6 presents **test-interval-overlap** test to determine if two given intervals overlap. Procedure **state-cons** checks the consistency of state history in a given scenario. The intervals [t_1, t_2] and [t_3, t_4] are guaranteed to overlap if **both** of the non-overlapping tests NOT1 ($t_2 < t_3$) and NOT2 ($t_2 < t_3$) do not have any solution.

Implication procedures: We first present the basic idea of implication using an example of protocol P

Table 6: **test-interval-overlap** to test whether intervals $[t_1, t_2]$ and $[t_3, t_4]$ overlap.

NOT1 Test ($t_2 < t_3$)	**NOT2 Test ($t_4 < t_1$)**	**Result**
Solvable	Don't care	May not overlap
Don't care	Solvable	May not overlap
Unsolvable	Unsolvable	Must overlap

described in Table 4. Let us perform backward implication of the event RTS-$TS_{0,1}$ at t_0 in the scenario presented in Figure 8. Event RTS-TS is output in row 1 and row 11 in transition table in Table 4. Hence we have two predecessors, p_1 and p_2 of the event. In order for the predecessor p_1 to be the one that triggers this event, node *0* must (1) be in state *Idle* for a period [t_a, t_b] such that $t_a < t_0$ and $t_b \geq t_0$, and (2) not be in state $BOFT$ for a period [t_u, t_v] such that $t_u < t_0$ and $t_v \geq t_0$. Condition 1 confirms the necessary state in which the input event *Pkt* can trigger the state transition to generate the RTS-TS event while condition 2 rules out the predecessor p_2. Lemma 9 formally states this process of unique implication.

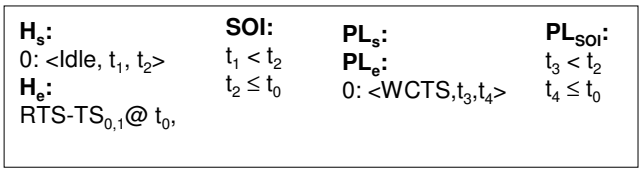


Figure 8: An example partial scenario of protocol P.

Lemma 9. *Given an entity x that has N predecessors, it is guaranteed that the predecessor p_k uniquely creates x if only p_k satisfies the condition of existence, and all other N-1 predecessors either satisfy the condition of non-existence or are ruled out.*

Proof. Condition of existence for only predecessor p_k is a necessary condition for unique implication. The condition of non-existence for all other entities rules out other predecessors to be derived by implication, and therefore is sufficient in order for p_k to be the unique implication of x. □

In the above example, backward implication of RTS-$TS_{0,1}$ generates event $Pkt_{0,1}$ at t_0 and state 0: $< Tx, t_0, unknown >$ because of the unique implication. If more than one predecessors satisfy the condition of existence, then we prune the branch of the test generation search tree. If none of the predecessors satisfies the condition, we continue without any implication at this point. While performing implication for an entity, we first carry out the process of elimination described above (procedure **elim-rows** to eliminate predecessors. If number of predecessors left after elimination is more than one, we check if there exists an entity common to all the predecessors that we can derive uniquely (procedure **row-intersection**). Procedure **bk-implication** presented in technical report [24] performs the backward implication of a given entity.

Scenario specification using implications: Given a scenario presented in a tree node T_n, procedure **specify-scenario** applies both backward and forward implications on all entities of the scenario as long as as the implication procedures generate new information. In other words, it stops performing implication at step (or round) k if no new information is generated in the k^{th} implication step. Note that procedure **fw-implication** is similar to procedure **bk-implication** with the exceptions that it starts with the rows of the transition table where the entity is an input.

Table 7: **find-relation** to determine relationship of time stamps.

EQ Test ($t_u = t_v$)	**SLT Test ($t_u < t_v$)**	**SGT Test ($t_u > t_v$)**	**Output**
Solvable	Unsolvable	Unsolvable	$t_u = t_v$
Solvable	Solvable	Unsolvable	$t_u \leq t_v$
Solvable	Unsolvable	Solvable	$t_u \geq t_v$
Solvable	Solvable	Solvable	Unknown
Unsolvable	Unsolvable	Unsolvable	Unknown
Unsolvable	Solvable	Unsolvable	$t_u < t_v$
Unsolvable	Unsolvable	Solvable	$t_u > t_v$
Unsolvable	Solvable	Solvable	Unknown

5.3 Heuristics

Our search-based framework covers the whole search space and hence is guaranteed to generate the worst case scenario provided sufficient run-time is allowed. To increase the possibility of generating the worst scenario in practical time we have developed heuristics. The heuristics are based on the protocol performance objective and the characteristics of the protocol or the class of the protocols under study. Following is a list of heuristics we have developed to generate worst/best case scenarios considering three performance metrics, namely throughput,

energy efficiency and fairness.

Heuristic 1. Weights for wanted transitions: We assign relative weights to wanted transitions depending on our study objective. In this case, higher weight is assigned to a transition that is deemed more likely to lead to the scenario that meets the study objective.

We use this heuristic to select one of the possible wanted transitions (and the associated wanted conditions) in the first step of test generation to construct the initial partial scenario.

Heuristic 2. Message selection: Among alternative messages, choose the message with the longest duration if the message is meant to overlap with another (for example, for a collision). Choose the message with the shortest duration if the message is meant not to overlap.

Heuristic 3. Network node selection (message success): Among alternative receiving nodes, choose the node with the lowest node degree[3] if the reception at the node is meant to be successful. Choose the node with highest degree if the reception is meant not to be successful or if a message is desired. Among alternative transmitting nodes, choose the node with the lowest node degree if the transmission or the channel access is meant to be successful.

Choosing the receiver with the lowest node degree increases the probability of a successful reception at the node as there are the least number of nodes interfering with the reception. Choosing the transmitter with the lowest node degree increases the probability of getting access to the wireless channel as there are the least number of nodes to contend for the channel.

Heuristic 4. Network node selection (message overlapping): Among alternative transmitting nodes, choose the node with a neighbor of the highest node degree if a message overlap is desired.

Choosing the transmitter that has a neighbor with the highest node degree (among all neighbors of all transmitters to choose from) increases the probability of message overlap. Consider two transmitters A and B from which we can choose one. The node A is connected to node C which has one neighbor E. The node B is connected to node D which has three neighbors E, F and G. Node A can potentially overlap with the transmission from node C to E. Node D can potentially transmit to any of the three neighbors with which the transmission from B can overlap with. Therefore, the probability of the message overlap is higher if we choose the transmitter B in this case. Note that heuristic 3 chooses the transmitter considering the node degree of the alternative transmitters while heuristic 4 chooses the transmitter considering the node degree of all 1-hop neighbors of the alternative transmitters.

Heuristic 5. Network node selection (throughput of a network): When the study objective is to minimize network throughput, among alternative transmitter-receiver pairs, choose the pair that maximizes the number of **victim** nodes in the network. Choose the pair that minimizes the number of victim nodes when the study objective is to maximize the throughput.

A transmitting node is defined as a **victim** if the transmitter is in back-off state because of a failed transmission. A receiving (intended recipient of a transmission) node is defined as a **victim** if the receiver is in back-off state because of a collision. A node which is neither a transmitter nor a receiver is defined as a **victim** if it cannot transmit whenever it wishes because it is deferring access to the channel. Maximizing the number of victims in a given network heuristically minimizes the network throughput as number of failed transmission and reception is maximized in this case. Note that the definition of **victim** is specific to the performance metric. For example, a node deferring access to the channel is a victim with respect to throughput, however, it is not a victim with respect to energy efficiency as the node saves energy while in defer state.

Heuristic 6. Network node selection (fairness of a network): To minimize fairness of a network, choose a set of transmitter-receiver pair that maximizes the number of **gainers** while maximizing the number of victims in the network.

A transmitting node is defined as a **gainer** if the transmitter is successful in gaining access to the channel. A receiving node is defined as a **gainer** if the receiver successfully receives the transmission.

Heuristic 7. Retransmission: If there are alternatives to choose from a transmission and a retransmission, always choose the retransmission if the study objective is directly related to increasing delay or energy waste.

Heuristic 8. Message selection (timer state): If there are alternatives to choose message triggering a timer state, choose the message depending on the type of timer (periodic vs. suppressive vs. non-suppressive timer) and the relation of the timer with the performance metric. If the study objective is directly related to delay, then for suppressive and non-suppressive timer states, choose the message that triggers the state to be longest.

[3]The number of 1-hop neighbors.

For example, for throughput minimization choose defer state (non-suppressive timer state) triggered by RTS (compared to defer state triggered by CTS).

5.4 An Example

In this section, we present an example scenario generation using protocol P (Table 4) in topology I (Figure 2). Our goal is to generate the scenario that leads to worst case throughput at link 0-1 in the topology. The inputs to the test generation algorithm are the topology, the transition table and the wanted transitions/conditions and unwanted transitions, and relation between them (Figure 6). We first construct an initial partial scenario (error) that we input as the root node of the search tree. In constructing the error, we use heuristics to choose the initial scenario that is deemed likely to lead to the worst case. We also use heuristics in various steps of our search algorithm to choose network nodes and protocol messages and reduce the run time complexity. To demonstrate the effectiveness of heuristics we present the example without and with the use of the heuristics in Sections 5.4.1 and 5.4.2, respectively.

5.4.1 Without Heuristics

Given the inputs, the test generation algorithm first constructs all possible initial partial scenario. Figure 9 presents the choices of the error scenario from which the algorithm proceeds with the first choice. The algorithm systematically uses the output of error generation (Figure 6.(d)) and topology to construct the initial partial scenario defined as **error**. For example, there are six alternative choices of receivers to be in one of the three wanted conditions when the study objective is to minimize throughput at link 0-1 of the topology. Among these choices, each BOCOL state of the receiver is further enumerated as BOCOL state due to unsuccessful RTS (U-RTS) and unsuccessful Data (U-Data). Among all these choices, the algorithm proceeds with the first choice which is the case when node 1 is in BOCOL state due to U-RTS. At this point, the algorithm further enumerates based on the fact that a collision needs at least another message to overlap with. This leads to 3 initial error scenarios in which the $\text{RTS}_{0,1}$ collides with $\text{RTS}_{2,3}$, $\text{CTS}_{2,3}$, and $\text{Data}_{2,3}$ defined as E_1, E_2, and E_3, respectively. In the figure, the choices in **bold** denotes the first choice at every enumeration step that the algorithm takes in order to construct the first error scenario E_1 that we use to illustrate the example in this section. Note that here we assume that the algorithm does not use any heuristic in constructing the error. We use search and implication algorithms to generate all valid scenarios leading from **all** initial errors (E_1, E_2, E_3,....). Finally, we compare **all** valid scenarios to select the scenario that leads to worst case throughput at link 0-1. For better understanding of the algorithms, we present the search and implication steps taken while generating a scenario given E_1 as initial error scenario.

Given a scenario, we first perform implication starting with the known events and states that continues as long as the implication procedure adds new information into the scenario. Implication of an entity may add an entity (event, state, time relation) into the scenario if it does not exist in the scenario. The scenario is pruned if the new entity to be added to the scenario exists in the corresponding prohibited list with certainty (i.e., if the **check-existence** procedure for that entity returns a "Yes" to check against the prohibited list). Figure 10 presents the scenario generation using implications in detail. Consider the error description E_1 presented in Figure 10.(a) that describes the collision of $\text{RTS}_{2,3}$ and $\text{RTS}_{0,1}$ at node 1 of topology I. We start by copying the error description E_1 into T_0, the root of the search tree as presented in the figure.

Implication of the events in the scenario adds the corresponding transmit-start (TS) events into H_e, timer relations into SOI, and prohibited entries into PL_e and PL_{SOI} as shown in Figure 10.(b). In the figure, we only show the entities that are added to the scenario (i.e., the entities that are generated by implication, but are not added to the scenario are not shown). For example, event entry $\text{RTS-RE}_{2,3}$ at t_0 is a successful reception at node 3 and by Lemma 5, there should be no other reception event that node 3 can potentially hear (or overhear) during the successful reception interval $[t_1, t_0]$. Since node 3 has only one neighbor (node 2), the two events are added in the prohibited list in Figure 10.(b). Whenever an entry is added to the prohibited list, our procedure checks if the entry exist in the history (H_e and H_s). If the prohibited entry exists in the history, the scenario is pruned. In this example, the prohibited entries do not exist in the history. Now, events $\text{RTS-RS}_{2,3}$ at t_1 and $\text{RTS-RS}_{0,1}$ at t_2 generate $\text{RTS-TS}_{2,3}$ at t_3 and $\text{RTS-TS}_{0,1}$ at t_4 respectively using implication rule 1. Figure 10.(c) and 10.(e) present the scenario after the first and second rounds of implications, respectively.

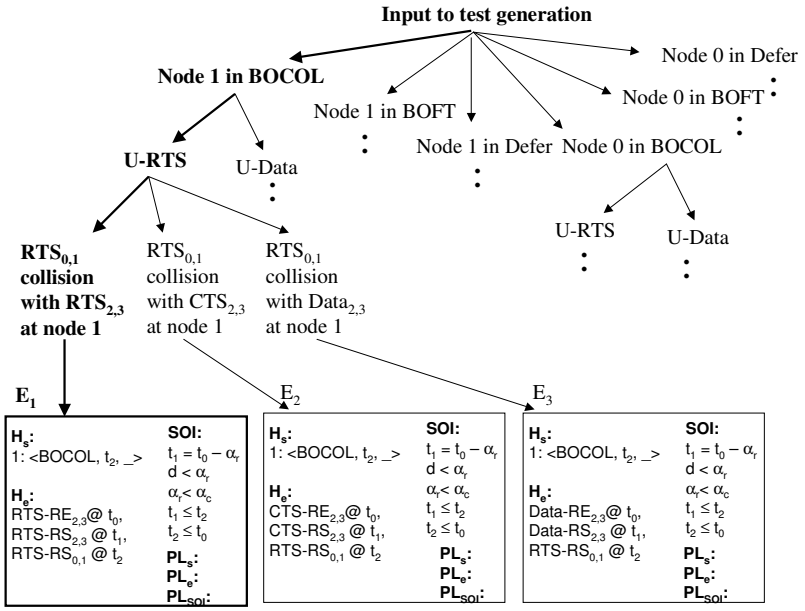


Figure 9: Constructing initial partial scenario.

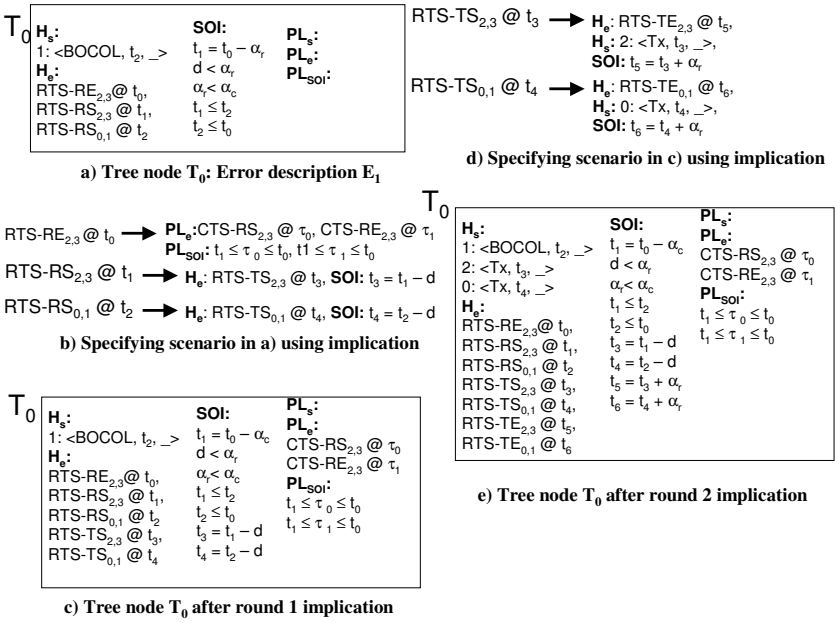


Figure 10: Scenario generation using error scenario E_1.

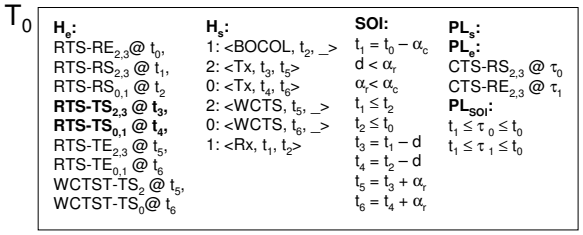


Figure 11: The scenario E_1 after 3 rounds of implications.

Figure 11 presents the scenario after it is specified using 3 rounds of implications at which point no new information is generated by the implication procedure. Therefore, we enumerate the tree node T_0 at this point. Note that the enumeration procedure returns all child nodes of T_n, however, we continue with one branch at a time to perform a DFS search. Table 8 presents the justification status of event history of the scenario presented in Figure 11. The column denoted by **Total predecessor** presents the number of predecessors of the event as derived from the transition table in Table 4. The column denoted by **check-existence** presents the existence status of the predecessor of the event as returned by the **check-existence** procedure. For example, RTS-$RE_{2,3}$ at t_0 has only one predecessor (RTS-$RS_{2,3}$ at t_1) that exists in the scenario, and hence it is justified. Events RTS-$TS_{2,3}$ at t_3 and RTS-$TS_{0,1}$ at t_4 are unjustified (**bold** in Figure 11) as none of their predecessors exists in the scenario.

Figure 12.(a) presents the predecessors of these unjustified entities as returned by **create-pred** procedure.

Table 8: **Justification status of event history of scenario in Figure 11.**

Event	Total predecessor	check-existence	Justification status
RTS-$RE_{2,3}$ at t_0	1	Yes	Justified
RTS-$RS_{2,3}$ at t_1	1	Yes	Justified
RTS-$RS_{0,1}$ at t_2	1	Yes	Justified
RTS-$TS_{2,3}$ at t_3	2	No	Unjustified
RTS-$TS_{0,1}$ at t_4	2	No	Unjustified
RTS-$TE_{2,3}$ at t_5	1	Yes	Justified
RTS-$TE_{0,1}$ at t_6	1	Yes	Justified
WCTST-TS_2 at t_5	1	Yes	Justified
WCTST-TE_2 at t_6	1	Yes	Justified

Note from the transition table (Table 4) that an RTS-TS has two predecessors: predecessor in row 1 represents a transmission of RTS when the node receives a packet from higher layer, and predecessor in row 11 represents a retransmission of RTS. Each RTS-TS event in the scenario T_0 has two predecessors, cross product of which results in 4 child nodes: T_{01}, T_{02}, T_{03}, and T_{04} shown in Figure 12.(b). Consider the tree node T_{01} in Figure 13. The node is a fully specified scenario as all entities in the scenario are justified, and hence it is a leaf node. A leaf node presents a test scenario that leads to the target error description.

Figure 14 presents the timing diagram of the above scenario. The timing diagram has been manually recreated using the SOI of the test scenario presented in Figure 13. Note that using this timing diagram we can regenerate the test scenario that leads to the target error and simulate the scenario in a network simulator, e.g., ns-2 [25]. We use SOI of **all** valid scenarios to compare the throughput reduction of the scenarios, and then output the scenario that leads to the worst throughput. Thus generating the worst scenario is unrealistic as it requires us to exhaustively explore the complete search space. Section 5.4.2 presents the execution of an approach for the same example using heuristics.

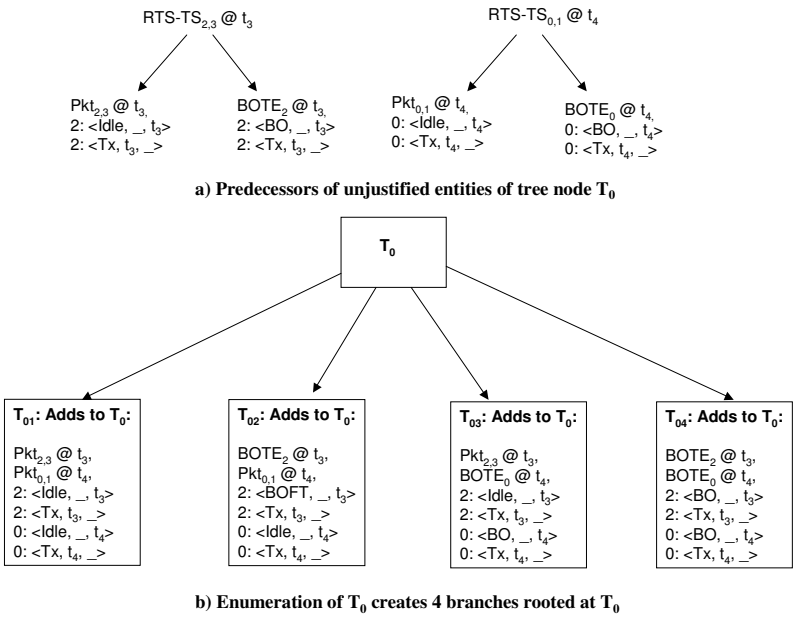


Figure 12: Enumerating child nodes.

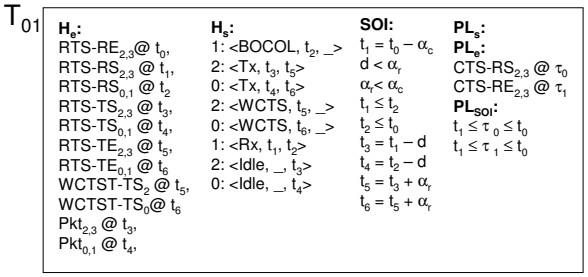


Figure 13: A leaf node: output scenario leading to error E_1 in Figure 10.(a).

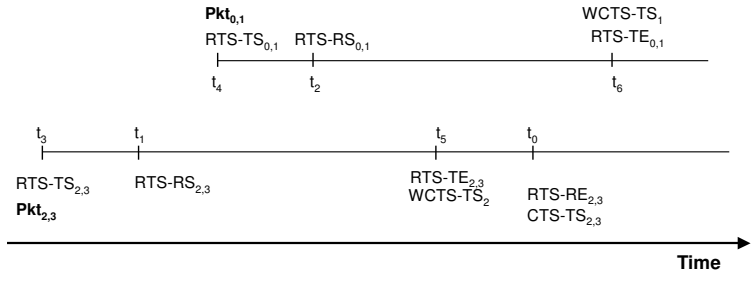


Figure 14: Timing diagram of scenario at leaf node in Figure 13.

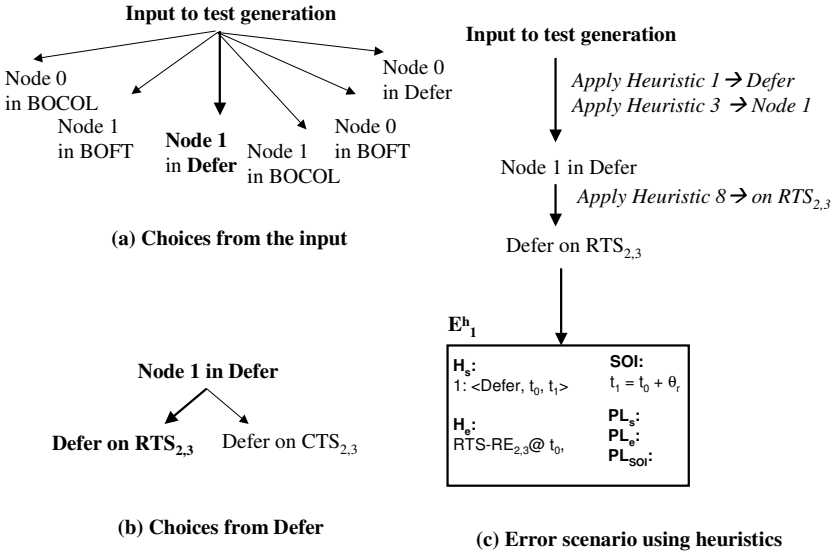


Figure 15: Constructing initial partial scenario (with heuristics).

5.4.2 With Heuristics

Given the inputs, the test generation algorithm first construct all possible initial partial scenarios (error) by using heuristics. Figure 15 presents the first choice that the algorithm selects to generate a valid scenario. If the algorithm finds a valid scenario by exploring this choice, it reports it as the scenario leading to worst throughput at link 0-1. Otherwise, it continues with the second choice, and so on. Figure 15.(a) presents the choices from the inputs to test generation algorithm. It uses Heuristic 1 to choose the Defer state as the wanted condition, as this is the condition with highest weight among the 3 wanted conditions; and Heuristic 3 to choose node 1 as the receiving node, as this is the node with highest node degree among the choices of node 0 and node 1. Figure 15.(b) presents the choices of node 1's Defer state on $\text{RTS}_{2,3}$ and $\text{CTS}_{2,3}$ from which the algorithm uses Heuristic 8 to choose the state with longest duration. Figure 15.(c) presents E_1^h, the first input partial scenario or the error scenario which is input to the test generation algorithm.

The algorithm first copies the error scenario to T_0^h, the root of the search tree presented in Figure 16.(a). Given a scenario, we perform implication for every event and state as long as the implication procedure adds new information into the scenario. We stop when implications do not add any new information and start enumerating choices. Figure 16.(b) presents the first leaf node which is generated from the first child of the tree root node after 3 rounds of implications. In this scenario, node 2 transmits $\text{RTS}_{2,3}$ on overhearing which node 1 defers for the period of θ_r. During the defer period, node 0 sends an $\text{RTS}_{0,1}$ to node 1 which it silently discards. Note that node 0 would eventually schedule a WCTS timer, upon expiration of which the node would backoff and retransmit the RTS. This defer followed by the WCTS and backoff (BOFT) leads to the worst throughput degradation at link 0-1. Note that without the heuristics the algorithm needs to generate all valid scenario and compare them to generate the worst case scenario. While the heuristics do not guarantee that the first scenario will be optimal, we have observed this in many of the cases we have studied.

6 Case Studies

We apply our framework to evaluate the worst case performance of IEEE 802.11b, MACA, and MACAW. The performance metrics we evaluate are throughput, fairness, and energy efficiency. We first describe our basic assumptions. Then we present the results of our studies of IEEE 802.11b and MACAW in Sections 6.2 and 6.3, respectively. Based on the study results, we present a brief comparative analysis of MACAW and IEEE 802.11b with respect to the test scenarios generated in Section 6.4.

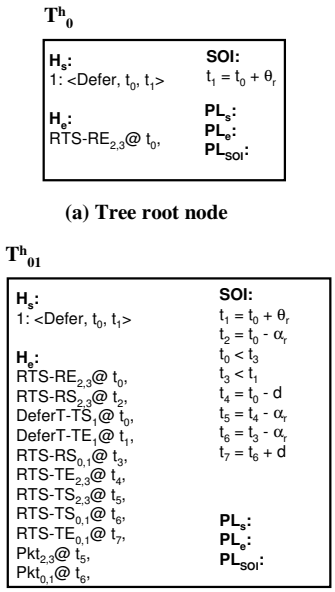


Figure 16: Scenario generation from the search tree root.

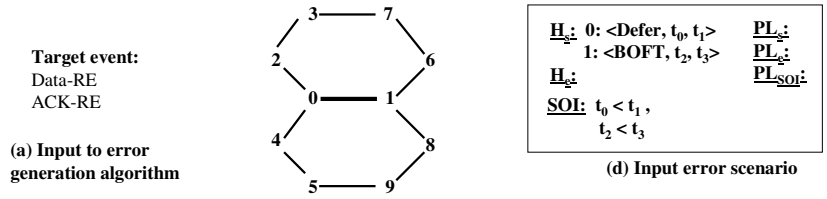
6.1 Assumptions

First, we assume that the network is static, i.e., neighborhood of a node is fixed and does not change. We could have relaxed this assumption by allowing the neighborhood to change over time as the semantics of time in our model allows the change of neighborhood at the granularity of individual message transmission or reception. However, for typical automobile speeds for a mobile node, the movement during the period of a message transmission is not significant. Therefore, we assumed a static network throughout our studies. Second, we do not model message loss explicitly, however, losses due to collision and silent drop are implicitly modeled. When a receiver is receiving transmissions from two nodes and is unable to clearly receive the signal from either node, the phenomenon is known as *collision*. When the receiver is able to clearly receive the signal from the closer transmitter, the phenomenon is called *capture*. Currently, our finite state machines are deterministic (DFSM) and model only collision and do not model capture. We can model such non-deterministic behavior of a receiver receiving two signals using a non-deterministic finite state machine (NDFSM). Finally, in the current version of our framework, we only model virtual carrier sense mechanisms.

6.2 Case Study: IEEE 802.11b

IEEE 802.11 is based on single channel multiple access schemes. The channel access is based on both physical and virtual carrier sensing mechanisms. [11] specifies the protocol[4]. In this study, we only consider the protocol mechanisms in DCF (Distributed Coordinated Function). DCF is based on *Carrier Sense Multiple Access with Collision Avoidance* (CSMA/CA). It allows channel access when both the channel sensing mechanisms indicate that the channel is idle. Each node maintains a timer called *Network Allocation Vector* (NAV) in order to monitor the channel status reserved by other nodes it hears. When node *0* of topology I (see Figure 2) wants to transmit data to node *1*, it first senses the channel status. If the channel is idle for a predefined period (DIFS), it sends RTS to *1* indicating the period *T* for which it expects to reserve the channel if RTS/CTS is successful. Node *0* schedules a wait-for-CTS (WCTS) timer to limit the period of time to wait for CTS. When node *1* receives RTS, if its NAV timer is not running, it immediately replies with a CTS packet in the next frame slot. Otherwise, it silently discards the RTS packet. While sending CTS, node *1* schedules a wait-for-data (WData) timer to limit

[4]See chapter 9 for MAC layer specification.



Wanted Transition	Description	Precondition/ Unwanted condition	Weight	Wanted Condition	
				Transmitter	Receiver
U-RTS	Unsuccessful RTS	Pkt	1	BOFT	BOCOL
U-CTS	Unsuccessful CTS	S-RTS (Successful RTS)	2	BOCOL	BOFT
U-Data	Unsuccessful Data	S-RTS, S-CTS	3	BOFT	BOCOL
U-ACK	Unsuccessful ACK	S-RTS, S-CTS, S-Data	4		BOCOL
D-RTS	Silent drop of RTS	Pkt	5	BOFT	Defer

(b) Output of error generation algorithm

Figure 17: Scenario generation of worst case throughput at a link.

the period of time to wait for data. All other nodes in the range of transmitter and/or receiver update their NAV with the value as given in the RTS/CTS packets. During the period when a node has NAV running, it defers access to the channel. When node *0* receives CTS before its WCTS timer expires, it immediately transmits data while scheduling wait-for-ACK timer to wait for ACK. If ACK from node *1* does not arrive within this period, node *0* backs off and contends for the channel again. Upon reception of data from node *0*, node *1* immediately sends an ACK back to node *0*. The sets of states, events, time variables, and the transition table of the IEEE 802.11 are a superset of those of the example protocol P presented in Section 3.1, and, therefore, are not repeated.

6.2.1 Test Scenarios Generated

We used our framework to generate scenarios to evaluate worst case performance of the IEEE 802.11 protocol with respect to throughput, energy efficiency and fairness. We present the scenario that leads to worst throughput at a given link of a given topology in detail. A summary is presented for other scenarios.

- **Worst case throughput at a given link:** Figure 17 illustrates different phases of the test scenario generation leading to worst case throughput at a given link. Figure 17.(a) presents the target events Data-RE and ACK-RE which are input to the error generation algorithm. Figure 17.(b) presents the wanted conditions of transmitters and receivers for wanted transitions, relations between wanted and unwanted transitions, and the associated weights assigned to the transitions. For example, unsuccessful data (U-Data) leads to the wanted condition *BOFT* at the transmitter and *BOCOL* at the receiver. Weights assigned to different wanted transitions represent relative weights of all the transitions with respect to the given performance objective, namely throughput. Figure 17.(c) presents the input topology in which our objective is to generate scenario leading to the worst case throughput at link 0-1.
  The topology and the relation table is input to the test generation algorithm. The algorithm applies heuristic 1 (denoted by H1) and chooses the wanted conditions *BOFT* and *Defer* for the transmitting node 1 and the receiving node 0, respectively. The partial scenario constructed from these wanted conditions with the timing relations is presented in Figure 17.(d) which is input to the test generation algorithm. The test generation algorithm uses implications until no new information is added to the scenario. Then it justifies the entities in the scenario, creates predecessors of the unjustified entities, and enumerates child nodes. In the enumeration process, it uses heuristics while choosing network nodes and messages. The algorithm

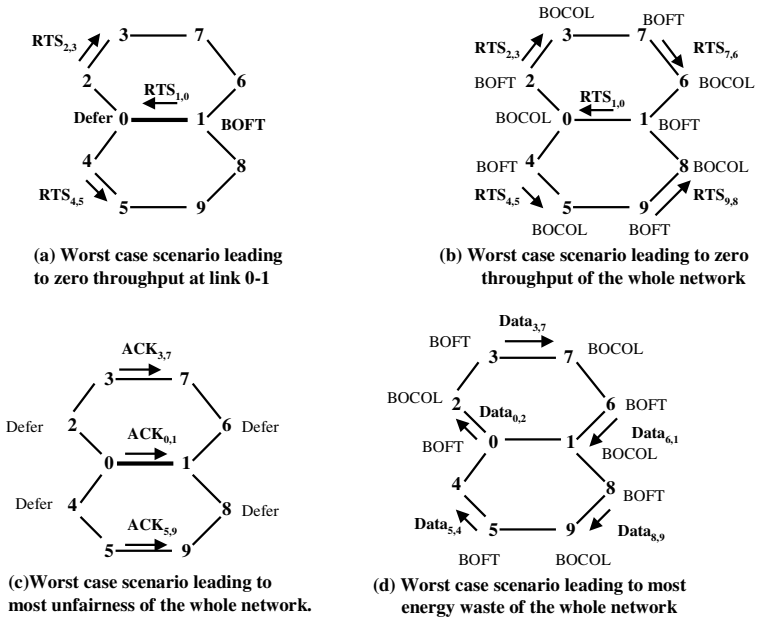


Figure 18: Worst case scenarios.

outputs the first leaf node as the scenario leading to the worst case throughput at link 0-1. Figure 18.(a) presents the output scenario in which node 0 silently drops the $RTS_{0,1}$ as it is deferring access to the channel on overhearing the $RTS_{2,3}$ and $RTS_{4,5}$ leading to **zero** throughput at the link 0-1. Here node 1 goes to $BOFT$ state after its WCTS timer is expired.

- **Worst case throughput for the whole network:** Figure 18.(b) presents the output scenario that leads to zero throughput for the entire network. The algorithm applies H5 to maximize the number of victims in the network by choosing victim transmitters and receivers in the network. The search, implication and heuristic lead to the output scenario presented in the figure. In this scenario, simultaneous RTS messages collide for which all transmitters are in BOFT state and all receivers are in BOCOL state, leading to zero network throughput.

- **Worst case fairness for the whole network:** Figure 18.(c) presents the scenario we generate that leads to the maximum unfairness in the network. The algorithm applies H6 to maximize the number of gainers in the network while maximizing the number of victims in the network by choosing the gainers (transmitter-receiver pairs) with no overlap between their transmission ranges (such that the transmissions from the gainers do not disrupt one another) and choosing the victims within the overlapping transmission range of the gainers. In this scenario, the 6 odd-numbered nodes are always transmitting or receiving and the 4 even numbered nodes are always in backoff or defer state and never get access to the channel, leading to the worst unfairness of the network.

- **Worst case energy efficiency at a given link:** Note that for energy efficiency, the weight of wanted transition D-RTS (see Figure 17.(b)) is the same as U-RTS, as deferring access to the channel saves energy. The algorithm applies H1 and H7 to construct input partial scenario from the output of error generation algorithm. In this scenario, retransmission of RTS followed by an U-ACK leads to worst energy waste at link 0-1.

- **Worst case energy efficiency for the whole network:** In the scenario presented in Figure 18.(d), retransmission of RTS in all transmitters and a collision of data at all receivers leads to the worst energy waste of the whole network.

6.2.2 Simulation Results

We simulate the scenarios we generated using ns-2 [25] to (1) verify our scenarios, and (2) use these scenarios to analyze the performance of the protocol. We also generate random scenarios for the same topologies to compare the performance degradation provided by the test scenarios we generate with respect to randomly generated scenarios.

We simulate the test scenarios we generated, namely those shown in Figure 18. The arrows in the figure represent the directions of flows in the test scenarios as generated by our framework. We use CBR sources at a rate of 0.6 MBPS. Total simulation time is 50 seconds for all scenarios. We use random scenarios in which the sources and sequences are assigned randomly. Table 9 presents the simulation results. The columns entitled

Table 9: Simulation results.

Scenario	NTh	AEE	AThG	AThV
Test	0.1478	0.59%	92%	0.167%
Random	1.81867	4.9%		

NTh and **AEE** represent the network throughput and average energy efficiency, respectively. The columns entitled **AThG** and **AThV** represent the average throughput (in %) of gainers and victims of the worst fairness test scenarios, respectively. Note that the average network throughputs for our test and the random scenarios are 0.1478 and 1.8186, respectively, representing a throughput reduction by a factor of 12 compared to random scenarios. The energy efficiency is represented by the average throughput achieved in the network per total energy spent in Joule. Our test scenarios achieve a reduction by a factor of 8 energy efficiency compared to random scenarios. Note that the gainers achieve 99.8% more throughput compared to the victims leading to extreme unfairness. These results demonstrate that IEEE 802.11 is extremely unfair in the sense that some nodes in the network starve while other nodes achieve high throughput. The results also demonstrate that the throughput can reach zero with an increase in the number of ongoing transmissions in the neighborhood. Zero throughput scenarios have been observed frequently in real deployed networks but never been formally analyzed or explained. Such short term unfairness severely affects performance of TCP and real-time applications [21]. The degradation of throughput as well as energy efficiency are significantly worse for our test scenarios compared to random scenarios.

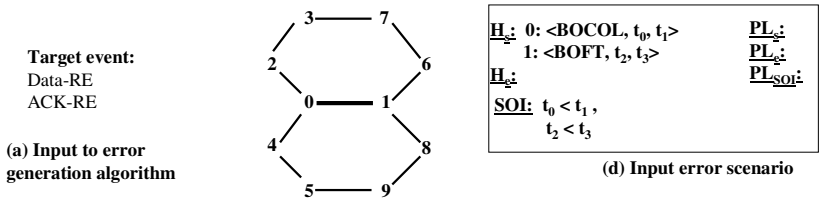
6.3 Case Study: MACAW

MACAW (Multiple Access Collision Avoidance Wireless) [14] is derived from MACA [13] to tackle the unreliability of wireless medium. The basic handshaking mechanisms of MACAW are similar to IEEE 802.11 with the following differences. First, there is no physical carrier sense in MACAW. Second, a node, after receiving a CTS in response to its RTS message, sends a short DS (Data-Send) packet before sending the data. The DS packet allows the neighborhood of the transmitter to be aware of the successful RTS/CTS handshake. Third, while deferring access to the channel on overheard RTS/CTS, a node does not drop an RTS destined for it. During the next contention period, the node sends out an Request for RTS (RRTS) message to the sender of the RTS to initiate the RTS/CTS handshake. For this case study, we first model the MACAW protocol using a transition table. We take the same topology and error description as in our case study of IEEE 802.11. We then apply our algorithms to generate test scenarios that lead to errors in MACAW. Section 6.3.1 presents example test scenarios.

6.3.1 Example Test Scenarios

We used our framework to evaluate worst case performance of the IEEE 802.11 protocol with respect to throughput, energy efficiency and fairness. We present the scenario that leads to worst throughput at a given link of a given topology in detail. A very brief summary is presented for other scenarios as most of the scenarios are similar to the scenarios generated for IEEE 802.11b in Section 6.2.1.

- **Worst case throughput at a given link:** Figure 19 illustrates different phases of the test scenario generation leading to worst case throughput at a given link. Figure 19.(a) presents the target events Data-RE and ACK-RE which are input to the error generation algorithm similar to 802.11b. Figure 19.(b)



(c) Topology: $G_0 = \{1,2,4\}$, $G_1 = \{0,6,8\},\ldots,G_9 = \{5,8\}$

Wanted Transition	Description	Precondition/ Unwanted condition	Weight	Wanted Condition	
				Transmitter	Receiver
U-RTS	Unsuccessful RTS	Pkt	1	BOFT	BOCOL
U-CTS	Unsuccessful CTS	S-RTS (Successful RTS)	2	BOCOL	BOFT
U-Data	Unsuccessful Data	S-RTS, S-CTS	3	BOFT	BOCOL
U-ACK	Unsuccessful ACK	S-RTS, S-CTS, S-Data	4		BOCOL

(b) Output of error generation algorithm

Figure 19: Scenario generation of worst case throughput at a link.

presents the wanted conditions of transmitters and receivers for wanted transitions, relation between wanted and unwanted transitions, and the associated weights assigned to the transitions. Note that there are four wanted transitions: U-RTS, U-CTS, U-DATA, and U-ACK compared to five wanted transitions of 802.11b (see Figure 17.(b)). Figure 19.(c) presents the same topology where our objective is to generate a scenario leading to the worst case throughput at link 0-1.
The topology and the relation table is input to the test generation algorithm. The algorithm applies H1 and chooses the wanted conditions $BOFT$ and $BOCOL$ for the transmitting node 1 and the receiving node 0, respectively. The partial scenario constructed from these wanted conditions with the timing relations is presented in Figure 19.(d) which is input to the test generation algorithm. Note that the input partial scenario or error scenario in MACAW is different from the 802.11b error scenario (see Figure 17.(d)). The algorithm outputs the first leaf node as the scenario leading to the worst case throughput at link 0-1. Figure 20.(a) presents the output scenario in which node 0 is in BOCOL state because of collision between $RTS_{1,0}$ and $RTS_{2,3}$ leading to **zero** throughput at the link 0-1. Here node 1 goes to $BOFT$ state after its WCTS timer is expired.

- **Worst case throughput for the whole network:** Figure 20.(b) presents the output scenario that leads to zero throughput for the whole network. The scenario is the same as that for 802.11b.
- **Worst case fairness for the whole network:** Figure 20.(c) presents the scenario we generate that leads to the maximum unfairness in the network. The heuristics we apply in this case is same as 802.11b, however, the scenario is different from that we generate for 802.11b (see Figure 18.(c)). In MACAW, node defers access to the channel for the longest duration (until the Data and Acknowledgment ends) on overhearing the DS (Data Send) frame.
- **Worst case energy efficiency for the whole network:** In the scenario presented in Figure 20.(d), retransmission of RTS in all transmitters and a collision of data at all receivers leads to the worst energy waste of the whole network.

6.4 Comparison of MACAW and IEEE 802.11b

We present a comparative analysis of IEEE 802.11b and MACAW in this section based on the test results presented in Sections 6.2.1 and 6.3.1, respectively. In IEEE 802.11b, a node silently drops RTS destined for

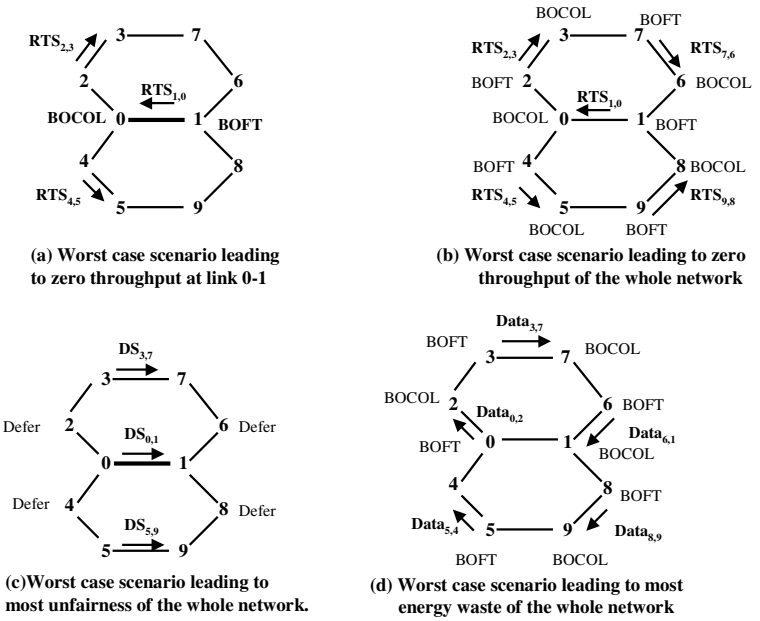


Figure 20: Worst case scenarios.

it when it defers access to channel, while in MACAW the node does not drop the RTS. Instead, it initiates the handshake by sending an RRTS message to the sender of the RTS. The advantage of this receiver-initiated handshaking mechanism of MACAW is that it allows the worst throughput and fairness scenarios to be less severe compared to 802.11b which is observed by comparing the SOI of the corresponding output scenarios. Simulating these scenarios could establish this fact more clearly, however, we did not perform simulation of MACAW.

7 Analysis of the Framework

Correctness and completeness: In this section, we prove the correctness and completeness of our framework.

Given a partial scenario at a tree node T_n, Lemma 2 guarantees that we enumerate all child nodes rooted at T_n. We first construct the input partial scenario and use it as the root of the search tree. We then apply a DFS search technique. The search and implication continue until we (1) reach a leaf node, or (2) we prune the scenario based on our implication rules. As we have shown that our algorithm detects a leaf node correctly and prunes tree nodes correctly, we can guarantee the completeness of our algorithm in the sense that it will always generate all valid scenarios that lead to the input partial scenario. According to Lemma 3 the algorithm stops enumeration at a leaf node and hence, ensures the generation of valid scenarios. According to Lemmas 5, 6, and 7 the algorithm detects the existence of prohibited event entries in the scenario, while Lemma 8 shows that any inconsistency in the state history is identified, and hence the algorithm correctly identifies all invalid scenarios to prune. Therefore, completeness is guaranteed.

Complexity: We present a brief analysis of our test generation results to report the complexity of our search and implication algorithms using the case study of IEEE 802.11. We also present run-times with and without the heuristics to show the usefulness of the heuristics in generating the worst case scenarios at lower run-times. Total number of states, events, and rows in transition table of the protocol are 15, 31, and 132, respectively. Note that we use implications to derive node state and event history, and to eliminate invalid branches. We can generate valid scenario using our search algorithm without the implication algorithm. However, in that case we will not be able to identify any invalid scenarios. For this reason, we use implication algorithm in our case study. Implication also reduces the complexity of our basic search. In one example scenario, the algorithm enumerates 4 child nodes after 7 rounds of implication. Without performing implications, it would enumerate 12 child nodes at the root node, that would increase exponentially as the search moves down the tree. Table 10 presents statistics recorded

from executions of our framework for different types of performance scenarios on various topologies. The two rows present complexity metrics for the generation of the scenario shown in Figure 17.(d) when our framework is used without and with the heuristics. The column denoted by **NCR** indicates the number of child nodes enumerated at the search tree root. The columns denoted by **H** presents the height of the tree at which the worst scenario was generated. The columns denoted by **IR**, **IC**, **exist**, and **LP** indicate the total rounds of implications, the total number of implication calls, total calls to **check-existence** procedure, and total calls to LP (linear programming tool), respectively. The column denoted by **ET** presents the total execution time for the test run on a 2.1GHz Pentium4 machine in hours. In the case of basic search without heuristics, the column **ET** denotes the execution time to generate the *first* valid scenario which may not be the worst case scenario. We ran the basic search for up to 7 days at which point we manually terminated the test. In the case of the search with heuristics, the column **ET** denotes the execution time of the worst case scenario generation. In contrast, the same search algorithm with heuristics can find a scenario that we report at much lower complexity.

Strengths and limitations: The complexity of basic framework is low because of the mix of backward

Table 10: Search and implication statistics.

Scen.	NCR	H.	IR	IC	exist	LP	ET
No heuristic	191808	15	59	9458	21793	157822	10.41+
With heuristics	1	2	7	110	234	363	2.4 x 10^-4

and forward search as well as the use of implication rules to eliminate invalid choices earlier. The complexity of generating worst case scenarios is reduced by the use of heuristics. The reduction in complexity enables the use of the framework to test large networks.

The most powerful application of our framework is in the design of network protocol. Protocol designers can use our framework to test worst performance. Based on the test results, designers can identify the problems in design, modify and re-test. The main limitation of our framework is that it does not report the likelihood of occurrence of the scenarios we generate.

8 Proposed Research

We developed a framework for worst case performance evaluation of wireless adhoc MAC protocols in terms of throughput, fairness and energy efficiency. We performed our first case study on IEEE 802.11b as it is an industry standard and has been deployed widely and rapidly for many different environments including enterprise, home, and public access network. Among the existing technologies for wireless LAN, for example, Wi-Fi[5], Bluetooth, HiperLAN, HomeRF, etc., IEEE 802.11b or Wi-Fi has received the widest market acceptance [22]. Today, IEEE 802.11 WLAN can be interpreted as a wireless version of Ethernet supporting best effort services [31]. However, recent growth of interest in wireless network supporting quality of service (QoS) has lead the 802.11 working group to enhance the 802.11 MAC protocol to support QoS [31]. We propose to extend our framework to incorporate the study of QoS in variants of CSMA/CA scheme. Section 8.1 presents our proposed framework extension to study QoS.

In designing our framework, we consider the basic handshake based CSMA/CA protocols which implicitly assume that no power control mechanism is used. Wireless nodes usually use batteries which can provide a limited amount of energy. Therefore, a major design consideration for adhoc network MAC protocol is the power consumption of individual nodes and the overall power consumption of the entire network. Absence of the centralized control in adhoc network increases the chances of collisions and channel assignment conflicts and leads to retransmission and control overhead resulting in higher power consumption[10]. Several power control schemes can be incorporated into handshake based CSMA/CA MAC schemes, e.g., transmit power control, sleep mode, battery level awareness, and reduced control overhead[10]. We propose to extend our framework to incorporate the study of power control schemes. In particular, our proposed extensions consider the transmit power control schemes based on transmission power control (TPC) and directional antennas. The proposed extensions to incorporate these newer technologies are presented in Section 8.2.

Our systematic approach of performance evaluation gives us insight that we can use to modify a given protocol

[5] Wi-Fi is based on IEEE 802.11b standard [12].

as well as to design a new protocol. Section 8.3 presents our proposed research in protocol modification and design.

8.1 Framework Extension to Study QoS

We propose to extend our basic framework to incorporate the study of QoS in variations of CSMA/CA scheme. In this section, we discuss the issues related to the proposed extension. We start our discussion with the definition of **QoS** in the domain of wireless networks.

Quality of Service (QoS): The term QoS refers to a broad collection of networking technologies and techniques. The goal of QoS is to provide guarantees on the ability of a network to deliver predictable results. Elements of network performance within the scope of QoS often include availability (uptime), bandwidth (throughput), latency (delay), and error rate [32]. In the domain of wireless network within the context of MAC layer, we define the **QoS** based on the analysis presented in [31]. It has been identified in [31] that legacy 802.11 (IEEE 802.11b) does not have any support for QoS because there is no mechanism to differentiate between stations and their traffic. Based on the analysis presented in [31], we formally define QoS as follows. Let P_1, P_2, ... P_n be n priority traffic classes in wireless nodes of a network where P_1 denotes the highest priority traffic class and P_n denotes the lowest priority traffic class. Let Th_1, Th_2, ... Th_n denote the throughput achieved for the respective priority classes. A protocol is said to be guaranteeing QoS if $Th_1 \geq Th_2 \geq ... \geq Th_n$. In other words, for two priority classes P_i and P_j such that P_i is a higher class compared to P_j, the QoS guarantees that $Th_i \geq Th_j$.

In Section 8.1.1, we discuss the extensions of our current framework required to incorporate the study of QoS. Section 8.1.2 presents a brief description of our case study on IEEE 802.11e with some initial results.

8.1.1 Extensions of the Framework

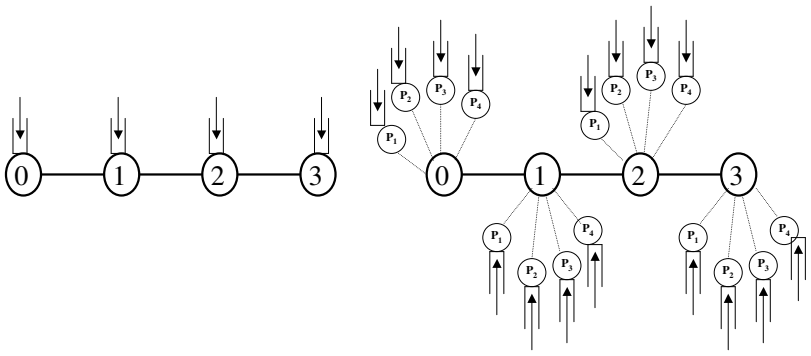
Figure 21 presents the model of a wireless node in the context of QoS support. We present the description assuming CSMA/CA MAC protocols are used for medium access. Figure 21.(a) presents the model of a wireless node when no QoS is supported. In this model, a single queue is attached to *each wireless node* in which traffic of all priority classes are serviced without any differentiation. Figure 21.(b) presents the model of a wireless node when QoS is supported. This model supports a maximum of four priority classes. One queue is attached to *each priority class* via which traffic of the corresponding class is serviced. When traffic from more than one priority class arrive at a wireless node, the higher priority class gets access to the channel before the lower priority class to assure the QoS. The access schemes between wireless nodes when multiple nodes want to access the channel at the same time, depend on the protocol under study. As our current framework is applicable to single channel multiple access protocols that use handshaking as the basic access scheme, we assume the same for the class protocols we consider in this section. Note that the channel access between wireless nodes are identical in the two models presented in Figure 21.

The main difference between the models of wireless nodes presented in Figure 21 are as follows. (1) There are *predefined* number of queues depending on the number of priority classes the wireless network supports to provide QoS. (2) A new mechanism is introduced to select the priority class which will access the channel when traffic from multiple classes arrive at the queues. (3) The values of contention windows and other MAC parameters are different for different priority classes. Next we show that despite these differences, we only need to add new heuristics to capture the new metric and do not need to extend the framework.

The values of contention windows and other MAC parameters, such as, interframe spaces, are modeled using various timers in our framework. Our current model supports all three types of timers, periodic, suppressive and non-suppressive timer. In our extended framework to study QoS, we need to model timer values specific to a class. For example, we need to specify $BOCOL_i$ to model the backoff (on collision) of priority class P_i as well as to specify the relation of backoff values between various classes using SOI. Thus the semantics of timers and SOI allow us to incorporate different priority classes.

As the number of priority classes are predefined and there is a predefined selection mechanism to select the class for channel access when traffic from multiple classes arrive within a node, we must incorporate new heuristics which is based on the priority class selection mechanism as well as our study objective. In this section, we present heuristics for protocol with **deterministic** selection mechanism.

Our study objective: Our objective is to generate a scenario which provide worst QoS for a given topology.



(a) Model of wireless nodes and network with no support for QoS.

(b) Model of wireless nodes and network with support for QoS: for 4 traffic classes.

Figure 21: Model of wireless nodes without and with QoS support.

Note that given two priority classes P_i and P_j such that P_i is the higher class compared to P_j, the QoS guarantees that $Th_i \geq Th_j$. Thus our study objective is to generate scenario that violates the above relation. In other words, our objective is to generate the scenarios that satisfy the relation $Th_i < Th_j$.

Active priority class: Let us define the priority class that is selected for channel access as the active priority class. The selection mechanism used by the protocol selects the active priority class when traffic from multiple classes arrive at a wireless node.

Deterministic priority class selection: When traffic from multiple classes arrive at a node, the protocol *deterministically* selects the highest priority class as the active priority class and allows it to access the channel. The access between wireless nodes are contention based.

Heuristic 9: Choose the lowest priority class of a node as a **gainer** node and the highest priority class of a node as **victim** node.

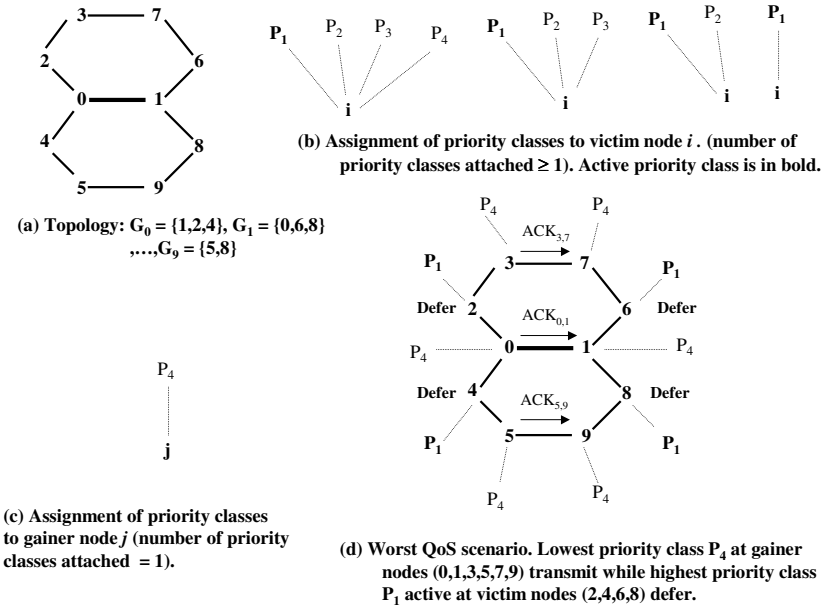
Since the assignment of priority class within a node is deterministic, we can choose the active priority class of a gainer and a victim node. Note that we apply the same concept of gainers and victims (see Section 5.3). However, we introduce a new concept of choosing the active priority class among the classes within the gainer and victim.

8.1.2 Case Study: IEEE 802.11e

IEEE 802.11e is an enhancement of the IEEE 802.11 standard MAC to support the quality of service [31]. [30] specifies the protocol. The contention based channel access is referred to as *enhanced distributed channel access* (EDCA). In this study, we only consider the QoS support mechanisms provided in EDCA. The QoS support in EDCA is provided by the introduction of *access categories* (ACs)[6] and multiple independent backoff entities. Packets are delivered by parallel backoff entities within one 802.11e node, where backoff entities are prioritized using AC-specific contention parameters called EDCA parameter set. There are four ACs and four backoff entities in every 802.11e node. The ACs are labeled according to their target application, i.e., AC_VO (voice), AC_VI (video), AC_BE (best effort), and AC_BK (background). The EDCA parameter set defines the priorities in medium access by setting individual interframe spaces, contention windows, and other parameters.

Contention based medium access is performed in backoff entity by using different parameter values for the EDCA parameter set. The backoff entity in a particular priority class in every node uses the same EDCA

[6]Same as *priority class* mentioned in previous section.



(a) Topology: $G_0 = \{1,2,4\}$, $G_1 = \{0,6,8\}$,…,$G_9 = \{5,8\}$

(b) Assignment of priority classes to victim node *i*. (number of priority classes attached ≥ 1). Active priority class is in bold.

(c) Assignment of priority classes to gainer node *j* (number of priority classes attached = 1).

(d) Worst QoS scenario. Lowest priority class P_4 at gainer nodes (0,1,3,5,7,9) transmit while highest priority class P_1 active at victim nodes (2,4,6,8) defer.

Figure 22: Worst case QoS scenario in IEEE 802.11e.

parameter [31]. During contention, when the contention window of two or more backoff entities in the same node allow the backoff entities to access channel at the same time, a virtual collision occurs. At this point, the backoff entity with the higher priority class transmits, whereas all other backoff entities act as if a collision has occurred on the channel. The channel access between different nodes of the network is the same as in IEEE 802.11 standard. In this study, we make the same assumptions as in our study of IEEE 802.11b (see Section 6.1).

Initial Results: We use the proposed extension of the framework to perform an initial analysis of IEEE 802.11e to evaluate the worst case QoS. Figure 22 presents the results of our analysis. Figure 22.(a) presents the topology. We use heuristic H9 to assign priority classes to victim and gainer nodes in the network. Figure 22.(b) presents the assignment of priority classes to victim nodes according to the heuristic. Note that the number of priority classes attached to victim nodes are ≥ 1 which must contain the highest priority class P_1. In this case, the priority class P_1 always gets access to the channel when the contention of channel occurs with other priority classes and is represented by the active priority class in **bold**. Figure 22.(c) presents the assignment of only lowest priority class P_4 to gainer nodes according to the heuristic H9. The number of priority classes attached to victim nodes is 1 with only P_4, which always gets access to the channel. Figure 22.(d) presents the scenario in which the gainer nodes (lowest priority classes) 0, 1, 3, 5, 7, and 9 transmit while the victim nodes (highest priority classes) defer because of the transmission between the gainers. Thus the aggregate throughput for the lowest priority class is higher than the aggregate throughput for the highest priority class resulting in worst QoS scenario. We plan to evaluate the above scenario using a network simulator.

Our proposed research to incorporate the study of QoS is presented in item 2 of Section 9.

8.2 Extensions to Study Power Control in MAC

We propose extensions of our basic framework to study power control in variations of CSMA/CA scheme. As mentioned earlier, we consider the transmit power control schemes for the proposed extensions. The largest source of power consumption at a wireless node is transmission power. Therefore controlling the power to appropriate level increases the energy efficiency of the MAC protocol [33]. Moreover, controlling the transmission power increases the spatial utilization of the wireless channel [33, 37, 39]. Among the transmit power control schemes, the schemes based on transmission power control (TPC) and directional antennas are widely studied. In the first case, in general, a wireless node dynamically varies the transmission power level for control and data packets to

achieve energy efficiency, to improve spatial utilization and to reduce collisions. In the second case, a wireless node directs its transmission to the intended receiver and potentially reduces the number of nodes with which its transmission interferes by using directional antennas instead of omnidirectional antennas. We have performed initial case studies using the proposed extensions to incorporate these two schemes and have identified problems in the corresponding protocols. Sections 8.2.1 and 8.2.2 present overview of the proposed extensions along with some initial results of the respective case studies.

8.2.1 Extensions to Study TPC Schemes

The basic TPC scheme in conjunction with CSMA/CA mechanisms operates as follows. The control packets RTS and CTS are transmitted using the highest power level, while Data and ACK are transmitted using the minimum power level necessary for the nodes to communicate [33]. The main objective of such a power control scheme is to increase energy efficiency and spatial utilization of the wireless network. The protocols proposed in [13, 34, 35, 36] use such a TPC based scheme to control power at the MAC level. In this section, we refer to these protocols as *basic TPC* scheme. Let P_{max} denote the maximum power at which RTS and CTS are sent. Let $P_{desired}$ denote the desired power at which the Data and ACK are sent.

In developing our basic framework, we considered a class of protocols with no power control. Hence, our framework implicitly assumes that all packets are sent at the same power level, and hence, the transmission range is the same for all packets. Next we describe the changes to our basic framework to incorporate the TPC schemes.

There are four components of our basic models: (1) network topology model, (2) model of network node states and protocol events, (3) protocol model, and (4) model of a scenario. Among these, only the topology model is changed to incorporate TPC class of protocols in our basic framework. In TPC schemes, a wireless node varies its transmission power depending on the message it transmits. Such a variation of transmission power changes the neighborhood depending on the message. The semantics of an event remain the same. However, in our basic framework the effect of a reception event when the transmission is from node i is modeled using G_i, whereas in the extended framework the effect of the event is modeled using G_i^m when m denotes the message type. The semantics of network node states and protocol events remain the same. Therefore, the semantics of the transition table as well as those of a scenario remain unchanged in all other ways.

There are three main conceptual components in our framework: (1) justification, (2) implication, and (3) prohibited list. The concept of justification is based on three main concepts: (1) predecessor, (2) compatibility of entities, and (3) checking existence of an entity. From Section 5.1, note that all these concepts are based on the model of network node states, protocol events and transition table, all of which remain fundamentally unchanged in the extended framework. The same argument applies to the concept of implication. The prohibited list is based on the implication rules using which we check the validity of a scenario. The change in the neighborhood to message level changes the rules presented in Lemma 5, 6, and 7 as all of these rules involve the reception of an event. We only discuss the changes in successful reception rule (Lemma 5). The other changes are similar.

- **Topology Model:** In TPC based schemes the amount of power used to transmit a message depends on the type of the message. For example, RTS and CTS packets are sent at P_{max} while Data and ACK packets are sent at $P_{desired}$. The power used for transmission determines the transmission range. Consider the network shown in Figure 23. Let node 2 initiate a data transmission to node 3 by sending an RTS. The $RTS_{2,3}$ is sent at P_{max} which is heard by all nodes in the transmission range of the RTS. We model such a message specific topology using the notation G_i^{msg}. For example, the transmission ranges of node i for RTS, CTS, Data and ACK messages are modeled as G_i^{RTS}, G_i^{CTS}, G_i^{Data}, and G_i^{ACK}, respectively. In Figure 23, for example, G_2^{RTS} = {0,1,3,4}, G_3^{CTS} = {1,2,4,5}, G_2^{Data} = {1,3}, and G_3^{ACK} = {2, 4}.

- **Successful reception rule:** Given a successful reception of a message m (i.e., a message of type m), denoted as m-RE$_{i,j}$, at time t$_v$ at node k and corresponding m-RS$_{i,j}$ (of type m) at time t$_u$ in a tree node T_y, there must not be any event n-RS$_{p,q}$ at time t$_w$ or n-RE$_{p,q}$ at time t$_x$ in T_y such that n is a message of the protocol, k is an element of the sets G_i^m and G_p^n, and t$_u \leq$ t$_w \leq$ t$_v$ and/or t$_u \leq$ t$_x \leq$ t$_v$.

Case Study: Basic TPC Protocols:

We use our proposed extension of the framework to perform an initial analysis of worst case performance of basic

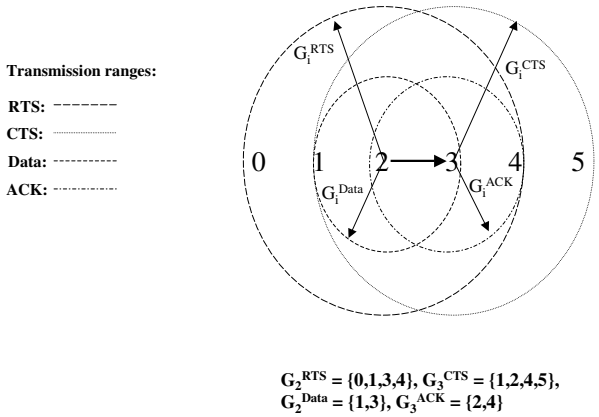


Figure 23: Model of network topology in basic TPC schemes.

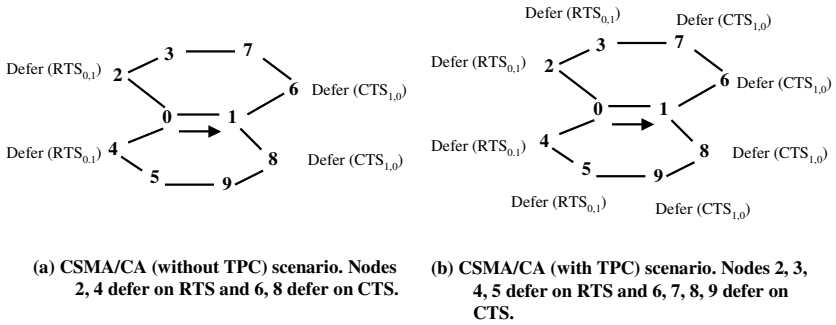
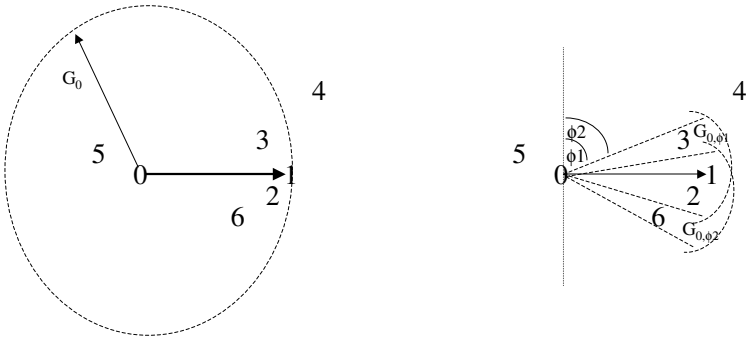


Figure 24: Worst case throughput and fairness scenario in schemes without and with TPC.

TPC schemes. Using our framework we generate scenarios which show that although the basic TPC schemes improve energy efficiency, they degrade throughput and fairness for the worst case scenarios. Figure 24 presents an overview of the worst scenario we generated. Figure 24.(a) presents the scenario when no power control scheme is used while Figure 24.(b) presents the scenario when basic TPC scheme is used. Let node 0 initiate a transmission to node 1 in Figure 24.(a) and Figure 24.(b). In schemes without power control, all messages are sent at the same power level. The nodes 2, 4, and 6, 8 defer on hearing the $RTS_{0,1}$ and $CTS_{1,0}$, respectively as shown in Figure 24.(a). When TPC is used by transmitting the RTS and CTS at maximum power level, nodes 2, 3, 4, and 5 defer on the RTS and the nodes 6, 7, 8, and 9 defer on the CTS as shown in Figure 24.(b). This essentially reduces the chances of collision. However, in highly connected network as in this example, it decreases fairness as well as throughput of the whole network. If the data transmission between nodes 0 and 1 continues for a long time, all other nodes in the network defer for the entire period of transmission when TPC is used. In case of transmission without power control, the simultaneous transmission between nodes 0 and 1, nodes 3 and 7, and nodes 5 and 9 are possible which increases the fairness as well as the throughput of the network.

Significant amount of prior research has identified various shortcomings of the basic TPC protocols. Extensions of the basic TPC schemes have been proposed to resolve these issues. However, to the best of our knowledge, the results and issues regarding the fairness of the TPC protocols that we have identified have not been identified



(a) Omni-directional antenna. Nodes 2,3,5,6 in the direction of omni-antenna defer on RTS while 0→1 transmission is in progress.

(b) Directional antenna. Node 0 can beamform its antenna towards node 1 using angle $\phi 1$ or $\phi 2$. Node 2,3 in the direction of $\phi 1$ ($G_{0,\phi 1}$) and node 2,6 in the direction of $\phi 2$ ($G_{0,\phi 2}$) defer on RTS while 0→1 transmission is in progress.

Figure 25: Model of network topology in schemes with directional antennas.

by any previous work. Regarding the throughput issues, [33] has identified that the throughput of the basic TPC protocols degrades compared to the CSMA/CA scheme without power control. However, the model of wireless channel assumed in [33] is different. The channel model in [33] considers the *carrier sensing zone* outside of the transmission range. The nodes in carrier sensing zone can sense the signal but cannot decode them correctly while the nodes in transmission range sense as well as decode the signal correctly. Therefore, on sensing the channel busy on RTS/CTS packets, the nodes in carrier sensing zone defer for a period of EIFS which is much smaller compared to the time length of entire data and ACK transmission. Thus the nodes in this zone exit the defer state earlier than the nodes in the transmission range[7] and can collide with the ongoing transmission as their RTSs are also sent at P_{max}. Following this observation, [33] concludes the degradation of throughput in the basic TPC scheme which is different from our results. Our conclusion is based on the simplest channel model consisting only of transmission range.

Our proposed research to incorporate the study of TPC based power control schemes is presented in item 2 of Section 9.

8.2.2 Extensions to Study Schemes with Directional Antennas

In our basic framework, we consider the class of protocols that use omnidirectional antennas, i.e., the antennas with 360 degrees pattern resulting in circular transmission ranges. In schemes with directional antennas, the transmitting nodes beamforms its antenna towards the intended receiver and transmit in that direction. In case of reception, the receiving node may beamform its receiving antenna towards the transmitter.

Using the same breakdown of basic models and conceptual components of our framework presented in Section 8.2.1, we can show that we need to change the topology model. However, in this case, the changes in the topology model introduces a new variable that must be tackled by changing our search algorithm.

- **Topology Model:** Figure 25 presents the changes in our network topology model when directional antennas are used. Figure 25.(a) presents the network topology model in our basic framework with omnidirectional antenna. In the figure, node 0 omnidirectionally transmits an RTS to node 1 which is heard by nodes 2, 3, 5, and 6. These nodes defer their transmission on the RTS for the entire duration of transmission from 0 to 1. Figure 25.(b) presents the same network with directional antennas. Node 0 can beamform its antenna towards node 1 at any angle ϕ between a reference line and the antenna. We assume a vertical reference line

[7]The time a node in transmission range defer on correctly receiving the RTS/CTS packet is the desired amount of time it should defer to avoid a collision.

as shown in the figure. Note that the constraints of the angle of the antenna is $0 \leq \phi \leq 2\pi$. In this figure, for simplicity we assume that the node 0 can beamform towards node 1 using angles $\phi 1$ and $\phi 2$ representing two different neighborhoods denoted by $G_{i,\phi 1}$ and $G_{i,\phi 2}$, respectively. Note that depending on the angle of antenna, nodes {2, 3} or nodes {2, 6} defer on overhearing the RTS. Thus the transmission range of node i in the extended framework is modeled by $G_{i,\phi}$ where ϕ denotes the angle at which the node i beamforms its antenna towards the intended receiver j. Note that the change in the topology model introduces a new variable ϕ because there are a certain range of the angles at which the node i can beamform its antenna towards the node j.

- **Model of protocol message and protocol transition table:** From Section 3.1, semantics of an event e from source node i to destination node j in our basic framework is: $e_{i,j}(\gamma)[\sigma]$ where e is the event ID, γ represents its relative timestamp which is the delay of e from the transition that creates e, and σ represents the set of nodes affected by e. Note that with directional antennas, a message transmitted from node i towards node j depends on the direction of antenna. The semantics of an event e, therefore, in the extended framework is: $e_{i,j,\phi}(\gamma)[\sigma]$ where ϕ represents the angle of the antenna at which the node i beamforms to transmit to node j. In general, the constrains on the angle is represented by inequalities, e.g., $\phi 1 \leq \phi \leq \phi 2$. Note that for TS and TE events (transmit start and end), σ represents i, and RS and RE events, σ represents $G_{i,\phi}$. This change in the semantics of protocol events changes the semantics of the transition table. The main concept behind the changes is as follows. Instead of a message transmission from node i to node j, the extended framework considers the message to be transmitted directionally from node i at an angle of ϕ from a reference line, under the constrains $\phi 1 \leq \phi \leq \phi 2$, with node j as the intended recipient.

- **Model of a scenario:** The components of a scenario in our basic framework are: history entries H_s, H_e and SOI, and the prohibited entries PL_s, PL_e and PL_{SOI}, where SOI and PL_{SOI} represent the system of time inequalities in the history and the prohibited lists, respectively. The introduction of a new variable ϕ adds inequalities representing the constraints on the angle of antenna directions both in the history and the prohibited list. Let us denote the inequalities of the angles in the history and the prohibited list by SOI_ϕ and PL_ϕ. Thus the components of a scenario in the extended framework are: H_s, H_e, SOI, SOI_ϕ, PL_s, PL_e, PL_{SOI}, and PL_ϕ.

- **Changes in timer Rules:** The change in the concept of an event changes the rules presented in Lemma 5, 6, and 7 as all of these rules involve the reception of an event. We only discuss the changes in the successful reception rule presented in Lemma 5 as follows. Given a successful reception of a message m defined by m-RE$_{i,j,\phi}$ at time t_v at node k and corresponding m-RS$_{i,j,\phi}$ at time t_u in a tree node T_y, there must not be any event n-RS$_{p,q,\delta}$ at time t_w or n-RE$_{p,q,\delta}$ at time t_x in T_y such that n is a message of the protocol, k is an element of the sets $G_{i,\phi}$ and $G_{i,\delta}$, and $t_u \leq t_w \leq t_v$ and/or $t_u \leq t_x \leq t_v$.

Case Study: DMAC Protocols:

The basic DMAC protocol is presented in [38]. The antenna model of the protocol is as follows. The transmission antenna always beamforms directionally towards the intended receiver. The reception is omnidirectional when the node is idle and directional when it receives from a transmitter. The main objective of the protocol for directionality is to increase spatial utilization and energy efficiency. We use our framework to perform an initial analysis of worst case performance of the protocol in terms of throughput, fairness and energy efficiency. The issues we identified have been previously known and identified, e.g., in [39]. Two main problems arise due to the use of directional antennas. First, because of directional transmission, there are more hidden terminals compared to the omnidirectional antennas. Second, during directional transmission, a node beamforms its antenna towards the intended receiver and remains **deaf** for the entire period of directional transmission. Similar issues have been observed in [39].

Figure 26 presents a scenario illustrating the problem of hidden terminals and the problem previously known as *deafness*. Let node 1 be transmitting to node 2 by beamforming its antenna towards node 2. $RTS_{1,2}$ is not heard by node 4 as the transmission was directional. Node 4, during the data transmission of node 2, transmits an $RTS_{4,1}$ by beamforming its antenna towards node 1. However, as the transmit antenna of node 1 is beamformed toward node 2, it is deaf to other signals and does not receive the $RTS_{4,1}$. Node 4 comes out of WCTS state, backs off and retransmits the RTS. Such backoff and retransmission lead to degradation of energy efficiency. In

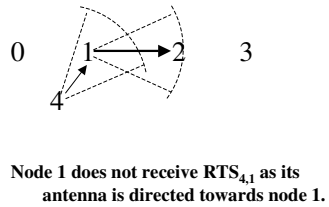


Figure 26: Scenario describing problems with directional antennas.

case of protocols without power control (i.e., with omnidirectional antennas), node 4 would receive the $RTS_{1,2}$ and defer for the entire duration of data and ACK transmission.

Our proposed research to incorporate the study of directional antenna based power control schemes is presented in item 2 of Section 9.

8.3 Application of Framework for Modification and Design of Protocols

Our systematic approach provides us insights about the performance issues and protocol mechanisms. We can use these insights to modify the protocol under study as well as to design new protocols.

We perform a systematic analysis of our results for the basic CSMA/CA as well as a variant with power control mechanisms. Based on our systematic analysis of protocol issues that lead to poor protocol performance, we propose several classes of protocol modifications in the area of power control. Section 8.3.1 presents the systematic approach we take to modify a given class or classes of protocols. Section 8.3.2 presents an overview of several classes of modifications we propose in the area of power control based on our systematic approach. Section 8.3.3 and 8.3.4 presents an overview of our initial analysis and the respective modified protocols based on basic TPC and directional antennas, respectively.

8.3.1 A Systematic Approach for Protocol Modification

We first define the scope of the modifications we plan to perform. In this step we identify the class of protocols and performance metrics that we would like to improve via modifications. In this step, we use our framework to identify the protocol issues that should be addressed during protocol modification. Then we identify the dimensions of the protocol class that we want to use for protocol modification. In this step, we use our insights of the protocol to identify the dimensions that seem more promising for improving performance. In the final step, we provide guidelines for performance improvement.

1. **Defining the scope of modifications:** We define the scope of the modifications or the classes of modifications within the power controlled CSMA/CA protocols. The performance metrics we plan to consider for the class of protocols are throughput, energy efficiency, fairness and QoS. In this step, we use our framework to generate the scenarios leading to degradation of a given performance metric which we analyze to identify the issues responsible for performance degradation. Following is a list of issues that we have identified using our framework: (1) collision, (2) defer, and (3) deafness. We analyze these issues in step 3 to provide guidelines for performance improvements.

2. **Identifying the dimensions of modifications:** We plan to address the protocol modifications in the area of the following three dimensions.

   (a) **Transmission power:** In the proposed modifications, we plan to vary transmission power in the following three levels. (1) Transmission power in the form of dynamic rate in which the transmission

range of a node i can be modeled as G_i^{rate} for a given *rate*. (2) Transmission power at the protocol message level (as in basic TPC schemes) in which the transmission range of a node i can be modeled as $G_i^{msg-type}$ for a given protocol *msg-type*. (3) Transmission power at the hop level in which a hop distance of 1 and 2 are defined within transmission ranges G_i^{Pmin} and G_i^{Pmax}, respectively, of a node i. Note that the existing class of TPC protocols assume omnidirectional antennas.

(b) **Antenna model:** We assume directionality of antennas in both transmitter and receiver of a wireless node in our proposed modifications. Accordingly, we model four types of antennas as follows: (1) both transmitting and receiving antennas are omnidirectional, (2) transmitting antenna is directional, while receiving antenna is omnidirectional, (3) receiving antenna is omnidirectional while transmitting antenna is directional, and (4) both transmitting and receiving antennas are directional. Note that the existing class of protocols on directional antennas typically assume the transmission of an antenna is at a fixed power level. They also assume that the direction of a directional antenna is fixed at node level and does not change with protocol message.

(c) **Control messages:** We also consider addition of new control messages as a dimension of modification.

3. **Analyzing the issues to provide guidelines for performance improvement:** In this step, we analyze the collisions, defers and deafness to provide a set of guidelines for performance improvement. We use these guidelines to propose the classes of modifications presented in Section 8.3.2.

(a) **Analysis of collisions:** We identify collisions as the most detrimental issue in degradation of throughput and energy efficiency. The main cause of collisions is hidden terminals which increase when directional antennas are used for power control. When an antenna is beamformed towards a direction for transmission, its transmission is hidden from nodes in all other directions. The collisions due to increased hidden terminals lead to more severe throughput degradation in directional antenna protocols. Consider a scenario in Figure 26, in which a transmission is in progress between nodes 1 and 2. Node 3 does not know about this transmission as it does not hear the $CTS_{2,1}$ because the CTS was transmitted directionally from node 2 to node 1. While the $Data_{1,2}$ is in progress, node 3 initiates a transmission by sending $RTS_{3,2}$ to node 2 which collides with the Data. The consequences of a collision are backoff and retransmission. While backoff degrades throughput, retransmission of protocol messages degrades throughput as well as energy efficiency.

Table 11 presents a summary of our analysis of collisions. The first column denotes the message that is under collision. The second column denotes the function of the message with respect to a collision. For example, the function of an RTS message is to alert the neighborhood of the sender to avoid collisions with respective CTS and Data messages. The function of a CTS message is to alert the neighborhood of the intended receiver to avoid collision with the corresponding Data. The third column denotes the amount of retransmission required if the message has a collision. For example, length of RTS, CTS, Data and ACK are given by α_r, α_c, β, and α_a, respectively, which are the amounts of retransmission required if the corresponding message has a collision. The fourth column represents the actual amount of loss incurred due to the collision, considering this message individually. For example, during a collision of Data, the sender needs to retransmit an RTS and the Data while the intended receiver needs to retransmit a CTS, incurring a total loss given by $\alpha_r + \alpha_c + \beta$. The fifth column denotes the corresponding collision probability. Let us denote the probability of collision with RTS, CTS, Data and ACK by p_r, p_c, p_d, and p_a, respectively. Note that length of Data (β) is much higher than the length of the control messages (α_r, α_c, and α_a). Therefore, the probability of a collision with a Data message is much higher than the probability of a collision with a control message ($p_d >> p_r, p_c, p_a$). Our analysis concludes that a collision involving a Data message is the collision which is most likely to occur while the collision involving an ACK message is the most wasteful[8]. The analysis of collisions provide us with the following guidelines to improve throughput:

- Improve average case throughput by reducing the collisions that are most likely to occur, e.g., collisions with Data.
- Improve worst case throughput by reducing the collisions that are most wasteful, e.g., collisions with ACK.

[8]Loss incurred by a collision with ACK is only higher by an amount of α_a than the loss incurred by a collision with Data.

- In general, reducing collisions also reduces energy wastage.

Table 11: Summary: Analysis of collisions.

Message coll	Fnc to avoid coll	Retx.	Actual loss	Coll. prob.	Comments
RTS	CTS, ACK	α_r	α_r	p_r	
CTS	Data	α_c	$\alpha_r + \alpha_c$	p_c	
Data		β	$\alpha_r + \alpha_c + \beta$	p_d	Most probable
ACK		α_a	$\alpha_r + \alpha_c + \beta + \alpha_a$	p_a	Most wasteful

(b) **Analysis of defer:** We identify defer as the most detrimental issue in degradation of short-term fairness by itself, i.e., degradation of short-term fairness without considering any other performance metric in conjunction. Defers can be classified as necessary and unnecessary. A defer of a node i is said to be *necessary* if it is necessary to avoid collision in the neighborhood, i.e., if the node i was not on defer, it would certainly collide in the neighborhood. A defer is said to be *unnecessary* if it does not avoid a certain collision in the neighborhood. In other words, if the node was not on defer, it would not collide in the neighborhood. We identify that there are two consequences of a defer irrespective of its necessity. First, the unnecessary defer (denoted by UD) is a consequence of defer[9]. Second, silent drop (denoted by SD) in defer state is an even more severe consequence of a defer. UD causes degradation of fairness and throughput while SD degrades fairness, throughput as well as energy efficiency.
There are two messages on which a node can defer by overhearing. The length of the defer period on overhearing RTS and CTS messages are given by $\theta_r(=\alpha_c + \beta + \alpha_a)$ and $\theta_c(=\beta + \alpha_a)$, respectively. Therefore, a UD on RTS is more wasteful compared to a UD on CTS. Comparing UD and SD, an SD on a defer which is UD on CTS is the most wasteful because it incurs the loss due to the drop as well as the loss due to the fact that the defer was unnecessary.

(c) **Analysis of deafness:** During directional transmission, a node beamforms its antenna towards the intended receiver and remains **deaf** for the entire period of directional transmission. We consider the deafness problem at the message level. For example, given a sequence of RTS-CTS-Data-ACK transmissions between two nodes i and j, the antenna of node i is beamformed or directed towards node j for RTS and Data transmissions while the antenna of node j is directed towards the node i for CTS and ACK transmissions. Note that $\beta >> (\alpha_r, \alpha_c,$ and $\alpha_a)$. Therefore, the deafness problem is the most likely to occur and most severe while node i is directed towards node j for the transmission of Data. Deafness causes degradation of both fairness and throughput. The guideline for performance improvement is as follows:

- Improve fairness and throughput by reducing the deafness problem while Data is transmitted.

8.3.2 Classes of Protocol Modifications

1. **Changing power level on message type to improve throughput:** Our analysis of protocol issues presented in Section 8.3.1 suggests that the power control at the level of protocol message (type) is a good candidate for improving throughput as well as energy efficiency. We propose following types of modifications to improve throughput and energy efficiency, each of which we plan to study extensively using our framework. We expect these modifications to also improve fairness. Let P_{max} denote the maximum power at which a node can transmit and $P_{desired}$ denote the desired power a node transmit to reach an intended receiver.

   - Transmit RTS, Data and ACK at $P_{desired}$, and CTS at P_{max}: Note from Table 11 that CTS is the control message that alerts the nodes in the neighborhood of the intended receiver to avoid collision with Data which is the most likely to occur. Sending CTS at P_{max} alerts the maximum number of nodes in the neighborhood of the intended receiver. Thus sending CTS at P_{max} maximally reduces the collisions of Data. This modification is expected to improve the average case throughput in addition to the CSMA/CA scheme. It is also expected to improve energy efficiency by reducing retransmissions due to collision. Compared to the basic TPC protocol, this modification is expected to improve the

[9]Note that necessary defer is not identified as a *consequence*.

fairness by allowing fewer nodes to allow transmission as the RTS is sent at $P_{desired}$. In the basic TPC protocol, RTS is sent at P_{max} to increase throughput, however, sending RTS at P_{max} causes maximum number of nodes to defer on RTS reducing the fairness (see Section 8.2.1).

- Transmit CTS, Data and ACK at $P_{desired}$, and RTS at P_{max}: Note from Table 11 that RTS is the control message that alerts the nodes in the neighborhood of the transmitter to avoid collisions with ACK which cause the maximum loss. Sending RTS at P_{max} alerts the maximum number of nodes in the neighborhood of the transmitter. Thus sending RTS at P_{max} is expected to maximally reduce the loss. This modification is expected to improve the worst case throughput in addition to the CSMA/CA scheme. It is also expected to improve energy efficiency by reducing retransmission due to collisions. Compared to the basic TPC protocol, this modification is expected to improve the fairness by allowing fewer nodes to allow transmission as CTS is sent at $P_{desired}$. In the basic TPC protocol, CTS is sent at P_{max} to increase throughput, however, sending CTS at P_{max} causes maximum number of nodes to defer on CTS reducing the fairness (see Section 8.2.1).
- Transmit Data, ACK at $P_{desired}$, and RTS, CTS at P_{max}: It reduces reduces the collision of Data and ACK as much as possible. This modification is expected to improve the average case and worst case throughput in addition to CSMA/CA scheme. It is also expected to improve energy efficiency by reducing retransmission due to collisions. This is in fact the basic TPC scheme as proposed in [33]. As we discovered, the fairness of this protocol degrades compared to basic CSMA/CA which in turn degrades throughput.

2. **Changing directionality on message type to improve throughput:** Existing classes of protocols based on directional antennas assume that the directionality of antenna is fixed and does not change at the level of individual messages. Based on the discussion of collisions and hidden terminal problems in directional antenna system presented in Section 8.3.1, we propose the following types of modifications that improve throughput and energy efficiency.

   - Transmit RTS, Data, ACK directionally to the intended receiver and CTS omnidirectionally: As mentioned earlier that transmitting CTS by alerting as many nodes as possible in the neighborhood of intended receiver maximally reduces collision of Data. This argument still holds in directional antenna systems. However, the increased number of hidden terminals in directional antenna system increases the collision and further reduces throughput. Therefore, transmitting CTS omnidirectionally is expected to improve average throughput and energy efficiency compared to CSMA/CA as well as existing directional antenna protocols.
   - Transmit CTS, Data, ACK directionally to the intended receiver and RTS omnidirectionally: This modification is expected to maximally improve the worst case throughput by reducing collisions with ACK. The improvement of throughput and energy efficiency are expected to be higher compared to CSMA/CA as well as existing directional antenna protocols.
   - Transmit Data, ACK directionally to the intended receiver and RTS, CTS omnidirectionally: This modification is expected to maximally improve the average and worst case throughput by reducing collisions with Data and ACK.
   - Transmit RTS, CTS directionally to the intended receiver and Data, ACK omnidirectionally: Transmitting Data and ACK omnidirectionally significantly reduces the severity of deafness problem as during Data transmission a node can hear messages of others to react to them later. Therefore, this modification is expected to significantly improve fairness. However, the omnidirectional transmission of Data increases the number of nodes to which the Data is likely to collide. Though such a collision does not effect the receiver of the omnidirectional Data, it may collide with others in the neighborhood and reduce throughput.

3. **Changing power level and directionality on hop distance from sender and receiver:** Let us denote the nodes in the neighborhood of node i as $G_i^{Pdesired}$ and G_i^{Pmax} when the transmission from node i is sent at $P_{desired}$ and P_{max}, respectively. Let us assume that the node is capable of transmitting only at these two power levels. The hop distance of a node j in $G_i^{Pdesired}$ is 1. Also consider another node k whose hop distance in G_i^{Pmax} is 2. Based on these assumptions, consider Figure 24.(b) in which all nodes within

hop distance of 1 and 2 defer equally on overhearing the $RTS_{0,1}$ and $CTS_{1,0}$ in case of basic TPC schemes. We propose following modifications. A node j at a hop distance of 1 from node i defers for the entire period on overhearing an $RTS_{i,m}$ or a $CTS_{i,m}$, while a node k at a hop distance of 2 from node i is free to transmit to a node l[10] such that $l \neq m$, with a low power, e.g., at $P_{desired}$. This modification is expected to improve fairness without degrading throughput. In fact, we expect the throughput to improve in this modification. We propose a similar modification in the context of directional antenna that a node k at a hop distance of 2 from i is free to transmit directionally to a node l such that $l \neq m$. However, the directional transmission, in this case, is not as effective for improvement of throughput as the TPC described above. The direction at which node k transmits to node l may coincide with the direction from node i to m and potentially collide with the ongoing transmission.

4. **Introducing new control messages:** Note that the UD problem occurs as the nodes in the neighborhood of senders and receivers do not know if the RTS/CTS exchange was successful. In other words, the UD problem (and consequently UD followed by SD) would not occur if the nodes in the neighborhood of the sender and the receiver would have a knowledge of the collision. We propose to add a new control message that we expect to improve fairness as well as throughput by solving the UD and SD problem partly.

   - The transmitter sends a Data Send (DS) packet after receiving the CTS successfully, right before sending Data. Nodes deferring on overheard RTS can get out of the defer state if they do not receive the DS packet within the expected interval. The DS packet in the neighborhood of the sender has been suggested in MACAW [14].
   - The receiver relays the DS packet received from the transmitter. Nodes deferring on overheard CTS can get out of the defer state if they do not receive the DS packet. Relaying of DS in the neighborhood of the receiver has not been previously suggested.

8.3.3 Case study: Modification and Design based on basic TPC Schemes

The basic TPC protocols are extensions of CSMA/CA protocols to incorporate transmission power control. Consider the transmission from node 0 to node 1 in Figure 24. All nodes in the network defer while 0→1 transmission is in progress in the basic TPC scheme. We apply the first of the class 1 modifications proposed in Section 8.3.2 in conjunction with CSMA/CA to improve performance. Let us call the new TPC scheme as **C-TPC** that operates as follows.

Unlike basic TPC scheme, RTS is sent at $P_{desired}$. As in the classical TPC scheme, CTS is sent at P_{max} and Data and ACK packets are sent at $P_{desired}$. Figure 27 presents the worst case throughput and fairness scenario as generated by our framework using C-TPC protocol in the given topology. As in C-TPC, RTS and CTS are sent at $P_{desired}$ and P_{max}, respectively, the nodes 2 and 4 defer on the RTS (similar to CSMA/CA) and the nodes 6, 7, 8, and 9 defer on the CTS (similar to TPC). As RTS is sent to avoid a collision with ACK, as such a collision is less likely to occur compared to a collision with Data (which is avoided by sending a CTS), by sending the RTS at a lower power we expect the C-TPC protocol to achieve more fairness and throughput compared to classical TPC schemes. As RTS, Data and ACK packets are sent at the desired power level, we expect the C-TPC to achieve more energy efficiency compared to CSMA/CA.

During the first communication between two nodes, RTS cannot be sent at $P_{desired}$ as the transmitter does not have any prior knowledge about the power level enough to reach the intended receiver. Thus, at the beginning (when no information about $P_{desired}$ is available), the transmitter must send the initial RTS at P_{max}. Subsequent RTS messages can be sent at $P_{desired}$.

8.3.4 Case Study: Modification and Design based on Directional Antennas

We perform an initial case study of the fourth of the class 2 modification proposed in Section 8.3.2. We name the new protocol as **C-DMAC**. We propose a new antenna model for C-DMAC as follows. The receive antenna is the same as DMAC, i.e., the node receives omnidirectionally while in idle. In all other times, the reception is directional to maximally reduce collisions. Unlike DMAC, the directionality of transmission antenna depends

[10]Node k gets the information of the receiver from the respective control message.

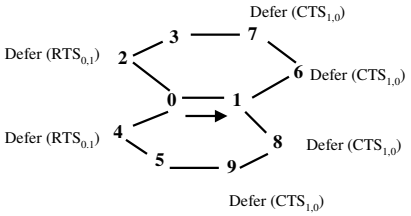


Figure 27: Worst throughput and fairness scenario in schemes with C-TPC.

on the message type and node density[11] of the network. RTS and CTS are transmitted directionally towards the intended receiver when node density is high and omnidirectionally when node density is low. The directional transmission of RTS/CTS reduces the collisions in highly densed network, while the omnidirectional transmission increases fairness in low density networks where collision is not a problem. Data and ACK are always transmitted omnidirectionally to avoid the problem of deafness. During the omnidirectional transmission of Data and ACK, a node receive RTS sent to itself and may respond to it later depending on the protocol mechanism[12].

Note that the RTS and CTS are sent to alert the neighborhood that are potential candidates for collisions with ACK and Data, respectively. As CTS transmission as well as Data reception are both directional in C-DMAC (like DMAC), the probability of a collision with Data is same in C-DMAC and DMAC. Similarly, as the RTS transmission and the ACK reception are both directional in C-DMAC and DMAC, the probability of a collision with ACK is the same in C-DMAC and DMAC. Therefore, the throughput in C-DMAC does not degrade compared to DMAC, but the energy efficiency in C-DMAC is expected to improve compared to DMAC as we eliminate the deafness problem. The omnidirectional transmission of Data and ACK, however, increases collisions at other nodes in the neighborhood. To reduce collisions, the transmission of Data and ACK can be sent at $P_{desired}$.

Our proposed research for protocol modification and design is presented in item 2 of Section 9.

9 Summary and Proposed Work

We propose a framework for worst case performance evaluation of wireless adhoc MAC protocols. Given a protocol performance objective, our framework first generates the protocol conditions that meet our study objective of generating worst case scenarios. It then applies our test generation algorithm based on search methods which is a mix of forward and backward search. Given the protocol condition or the partial scenario described in terms of network node states, events, and time relations, the test generation algorithm uses heuristics to generate the worst scenario, often in practical run time. We have used our framework to analyze throughput, energy efficiency and fairness of IEEE 802.11, MACAW and other protocols. Using it, we have generated scenarios that lead to extreme unfairness and reduction of throughput and energy efficiency by a factor of 12 and 8, respectively. The empirical results show that although the basic search algorithm is exponential, the use of heuristics reduces the complexity drastically to generate the worst scenarios in practical run times (for all cases we have tried).

We propose further extensions of our basic framework to incorporate the performance analysis of wireless MAC protocols for quality of service and power control. Using our framework, we have identified several problems and issues with existing protocols that have not been previously identified. Based on an initial results, we propose modifications and new protocols that are expected to improve performance. The fact that we could easily extend

[11] The number of nodes per unit area.

[12] For example, in MACAW [14], the node transmits the receiver-initiated RRTS packet to the sender of RTS.

our framework to new protocols and objectives illustrates the robustness as well as wider applicability of our framework.

A brief outline of our future work follows.

1. **Enhancing the performance of framework:** Currently our search framework reduces the complexity using heuristics that guide the search to generate the scenario in practical time. We plan to further reduce the complexity using bounding conditions.

2. **Complete the proposed extensions:** We propose extensions to incorporate the study of QoS and power control in CSMA/CA MAC protocols. We have already carried out modeling for these extensions and have performed initial manual analysis for specific case studies. We plan to complete these extensions and to thoroughly evaluate the classes of modifications by incorporating them into our framework. More features must be added to the framework to completely automate the extensions and case studies. We need to prove the correctness of the proposed extensions and perform experiments and simulations to demonstrate that the modifications provide expected benefits.
   Following is an outline for the proposed extensions presented in Section 8.

   - To incorporate the study of QoS, we only need to incorporate new heuristics in our framework. We plan to perform a case study of IEEE 802.11e and analyze the worst case QoS scenarios for various topologies.
   - Incorporating TPC schemes in our framework requires changes of search and implication procedures that use topology information. We plan to carry out case studies of basic TPC schemes as well as modifications proposed in Section 8.3 for various topologies.
   - The most extensive modifications of our framework is needed to incorporate the schemes based on directional antennas. Complexity of our algorithms is expected to increase due to the addition of the new variable ϕ. We plan to study the basic DMAC protocol as well as modifications proposed in Section 8.3.

3. **Extension to study MAC protocols on multiple channels:** The MAC protocols we have studied use a single channel. Earlier MAC protocols typically assume a single channel for all wireless nodes, while more recent approaches assume multiple channels for more efficient use of the wireless medium [10]. We plan to extend our framework to tackle the class of protocols based on multiple channels. Extension to incorporate multiple channel will require major change in the models as well as in the algorithms.

4. **Topology synthesis:** The topology is an input in our basic framework as well as in all extensions. We plan to incorporate topology synthesis algorithms in our framework. The objective of our topology synthesis framework is to generate the topology or a set of topologies that lead to the worst case performance of a given performance metric. The class of worst case topologies represents the class of connection patterns of wireless nodes that should be avoided during the deployment of the network. Prior knowledge of connections patterns that should be avoided is very useful, specially in the context of sensor networks.

5. **Incorporate the study of sensor MAC protocols:** We performed initial analysis of sensor MAC protocols (case study on SMAC[40]) using our basic framework. Sensor MAC protocols that are based on the basic CSMA/CA schemes can be handled by our framework without any significant modifications. The study only requires more periodic timers to control the sleep/wakeup cycles of wireless nodes as well as the sensors on the nodes. We plan to extend our topology synthesis approach to sensor MAC protocols. Using topology synthesis framework, we expect to generate or synthesize topologies or connection patterns that would lead to the worst case of a given performance metric, e.g., energy efficiency or latency. This will identify connection patterns to be avoided during deployment of the sensor network.

References

[1] G. Holzmann, "Design and Validation of Computer Protocols", Prentice Hall, ISBN 0L35399254.

[2] F. Lin, P. Chu, and M. Liu, "Protocol Verification Using Reachability Analysis", Computer Communication Review, vol. 17, no. 5, pp. 126-135, October 1987.

[3] R. Alur and D.L. Dill, "A theory of timed automata", Theoretical Computer Science 126:183-235, 1994.

[4] M. Kwiatkowska, G. Norman and D. Parker, "PRISM 2.0: A Tool for Probabilistic Model Checking", 1st International Conference on Quantitative Evaluation of Systems (QEST'04), pp. 322-323, IEEE Computer Society Press. September 2004.

[5] M. Kwiatkowska, G. Norman and D. Parker, "Probabilistic model checking in practice: Case studies with PRISM", Special issue of ACM Performance Evaluation Review on Performance and Verification, 32(4), pp. 16-21, March 2005.

[6] M. Kwiatkowska, G. Norman and J. Sproston. "Probabilistic Model Checking of the IEEE 802.11 Wireless Local Area Network Protocol", PAPM/PROBMIV '02, volume 2399 of LNCS, pp. 169-187, July 2002.

[7] M. Duflot, M. Kwiatkowska, G. Norman and D. Parker, "A Formal Analysis of Bluetooth Device Discovery", 1st International Symposium on Leveraging Applications of Formal Methods (ISOLA'04), November 2004.

[8] A. Helmy and D. Estrin, "Simulation based 'STRESS' Testing Case Study: A Multicast Routing Protocol", 6th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS), July 1998, Montreal, Canada.

[9] A. Helmy, D. Estrin, and S. Gupta. "Fault-oriented Test Generation for Multicast Routing Protocol Design", Joint International Conference on Formal Description Technique/Protocol Specification, Testing, and Verification (FORTE/PSTV), pp. 93-109, November 1998.

[10] R. Jurdak, C. Lopes, and P. Baldi, "A Survey, Classification and Comparative Analysis of Medium Access Control Protocols for Ad Hoc Networks", IEEE Communication Surveys and Tutorials, 1st quarter 2004, vol. 6, no. 1, pp. 2-16.

[11] IEEE Std 802.11-1997 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 IEEE Std 802.11-1997,Page(s): i -445

[12] ANSI/IEEE, "IEEE Standard for Wireless Lan Medium Access Control (MAC) And Physical Layer (PHY) Specifications", November 1999 Edition.

[13] P. Karn, "MACA - A New Channel Access Method for Packet Radio", 9th ARRL/CRRL Computer Networking Conference, pp. 134-140, 1990.

[14] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: A Media Access Protocol for Wireless LANs", ACM Special Interest Groups on Data Communication (SIGCOMM), pp. 212-225, September 1994.

[15] C. Fullmer and J. Garcia-Luna-Aceves, "Floor Acquisition Multiple Access (FAMA) for Packet-Radio Networks", Conference on Applications, Technologies, Architectures and Protocols for Computer Communication, pp. 262-273, 1995.

[16] F. Iulucci, M. Gerla, and L. Fratta, "MACA-BI (MACA BY Invitation): A Receiver-Oriented Access Protocol for Wireless Multihop Networks", 8th IEEE International Symposium on Personal, Indoor and Mobile Radio Communication, vol 2. 1997, pp 435-439.

[17] L. Bononi, M. Conti, and L. Donatiello, "Distributed Contention Control Mechanism for Power Saving in Random Access Ad-Hoc Wireless Local Area Networks", International Workshop on Mobile Multimedia Communication, IEEE, August 1999, pp. 114-123.

[18] J. Garcia-Luna-Aceves and A. Izamaloukas, "Reversing the Collision Avoidance Handshake in Wireless Networks", ACM/IEEE International Conference on Mobile Computing and Networking, 1999, Seattle, Washington.

[19] S. Khurana, A. Kahol, A. Jayasumana, "Effect of Hidden Terminals on the Performance of IEEE 802.11 MAC Protocol", IEEE 23rd Annual Conference on Local Computer Networks (LCN), pp 12-20, October 1998, Boston, Massachusetts.

[20] S. Khurana, A. Kahol, S. Gupta, P. Srimani, "Performance evaluation of distributed co-ordination function for IEEE 802.11 wireless LAN protocol in presence of mobile and hidden terminals", 7th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS), March 1999, College Park, Maryland.

[21] C. Koksal, H. Kassab, H. Balakrishnan, "An Analysis of Short-Term Fairness if Wireless Media Access Protocols", ACM SIGMETRICS international conference on Measurement and modeling of computer systems, pp 118-119, 2000.

[22] S. Sharma, "Analysis of 802.11b MAC: A QoS, Fairness, and Performance Perspective", Technical Report TR-126, Department of Computer Science, State University of New York, Stony Brook, January 2003.

[23] L. Bononi, M. Conti, E. Gregori, "Runtime Optimization of IEEE 802.11 Wireless LANs Performance", IEEE Trans. Parallel Distrib. Syst, pp. 66-80, January 2004.

[24] S. Begum, S. Gupta, A. Helmy, "Test Generation Framework for Performance Evaluation of Wireless AdHoc MAC Protocols", Technical report, http://www-scf.usc.edu/ sbegum/tech-report-eotg-1.pdf.

[25] L. Breslau, D. Estrin, K. Fall, S. Floyd, J. Heidemann, A. Helmy, P. Huang, S. McCanne, K. Varadhan, Y. Xu, and H. Yu, "Advances in Network Simulation", IEEE Computer, 33 (5), pp. 59-67, May 2000.

[26] LINDO systems, http://www.lindo.com.

[27] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function", IEEE Journal on Selected Areas of Communication (JSAC), vol. 18, no. 3, pp. 535-547, March 2000.

[28] "Analyzing the Short-term Fairness of the IEEE 802.11 in Wireless Multi-hop Radio Networks", C. Barrett and D. Engelhart, MASCOTS 2002.

[29] L. Li and P. Sinha, "Throughput and Energy Efficiency in Topology-Controlled Multi-hop Wireless Sensor Networks", WSNA 03.

[30] IEEE 802.11 WG, IEEE 802.11eD5.0, "IEEE Standard for Information Technology - Telecommunications And Information exchange Between Systems - Local And Metropolitan Area Networks-specific Requirements - Part 11: Wireless Medium Access Control (MAC) And Physical Layer (PHY) Specifications Amendment 8: Medium Access Control (MAC) Enhancements for Quality of Service Enhancement", IEEE Std 802.11e-2005 (Amendment to IEEE Std 802.11, 1999 Edition (Reaff 2003)), 2005 Page(s) 0_1 - 189.

[31] S. Mangold, C. Sunghyun, G.R. Hiertz, O. Klein, B. Walke, "Analysis of IEEE 802.11e for QoS Support in Wireless LANs", IEEE Wireless Communications, Volume 10, Issue 6, Dec 2003, pp 40-50.

[32] http://compnetworking.about.com/od/networkdesign/l/bldef-qos.htm.

[33] E. Jung and N. Vaidya, "A Power Control MAC Protocol for Ad Hoc Networks", MOBICOM 2002, September 2002, Atlanta, Georgia.

[34] S. Agarwal, S. Krishnamurthy, R. H. Katz, and S. K. Dao, "Distributed Power Control in Ad-Hoc Wireless Networks", PIMRC 2001.

[35] J. Gomej, A. T. Campbell, M. Naghshineh, and C. Bisdikian, "Conserving Transmission Power in Wireless AdHoc Networks", ICNP 2001.

[36] M. B. Pursley, H. B. Russel, and J. S. Wysocarski, "Energy-Efficient Transmission and Routing Protocols for Wireless Multi-hop Networks and Spread-Spectrum Radios", EUROCOMM 2000, pp 1-5, 2000.

[37] A. Nasipuri, S. Ye, J.You and R. Hiromoto, "A MAC Protocol for Mobile AdHoc Networks using Directional Antennas", IEEE WCNC, Chicago, IL, September 2000.

[38] R. Choudhury, X. Yang, R. Ramanathan and N. Vaidya, "Using Directional Antennas in Ad Hoc Networks", Final Report submitted from Texas A&M University to BBN Technologies, July 2001.

[39] R. Choudhury, X. Yang, R. Ramanathan and N. Vaidya, "Using Directional Antennas for Medium Access Control in Ad Hoc Networks", MOBICOM 02, Atlanta, Georgia, September 2002.

[40] W. Ye, J. Heidemann, D. Estrin, "An Energy-Efficient MAC Protocol for Wireless Sensor Networks", INFOCOM 2002, New York, NY.