A Framework for Modeling, Test Generation, and Performance Evaluation of Wireless AdHoc and Sensor MAC Protocols

Shamim Begum, Ahmed Helmy, Sandeep Gupta
Electrical Engineering, University of Southern California
sbegum@usc.edu, sandeep@poisson.usc.edu, helmy@ceng.usc.edu

MAC protocol is the main element for determining efficiency and fairness in sharing the limited communication bandwidth of a wireless channel. These protocols must also exercise power control. While new protocols and architectures have been designed to address these issues, no systematic approach has been proposed for testing these protocols. Traditional performance evaluation approaches [1, 2] typically evaluate average performance but do not capture the extreme cases. Given a MAC protocol for single channel adhoc networks and its performance objective, we systematically derive a set of protocol error descriptions that adversely effect the objective. We propose and develop a novel automatic test scenario generation framework that generates test scenarios leading to these errors. Our test generation algorithms employ efficient intelligent (forward and backward) search techniques to construct the sequence of events leading to these errors. As a case study, we use our framework to analyze performance of IEEE 802.11 for adhoc networks. Using our framework we generate library of scenarios in which some nodes in the network suffer from zero throughput while others achieve average throughput resulting in extreme unfairness in IEEE 802.11 networks. Some of our scenarios achieve channel utilization as low as 3%, a 90% reduction compared to utilization obtained for random scenarios that are commonly used for performance analysis. Empirical analysis of the case studies shows that the complexity of our algorithms is quite practical.

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 searched for test scenarios [3]. We propose a test generation framework that, instead of adopting the validation approach, uses a "falsification approach" and directly targets the error. Our main formalism is that of a finite state machine (FSM) using which we abstract the atomic representation of protocol events, network node states, and integrate time relations among the events and the state transitions into the abstraction. We define such representation of a system as a **scenario**. Figure 1 presents the block diagram of our performance evaluation framework. Given a protocol performance objective (e.g., throughput) we use error generation algorithms to generate a set of conditions that adversely effect the input performance objective. We define the set of conditions as **error**. The error is specified in terms of network node states, protocol events and time relations between the

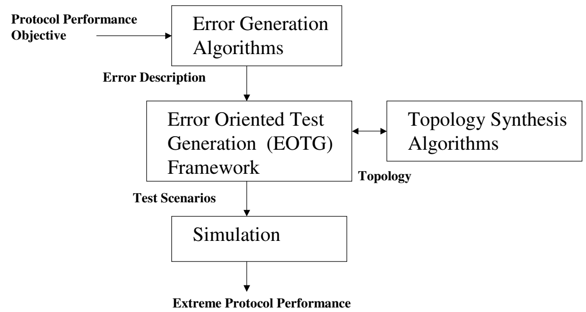

Figure 1: Performance evaluation framework.

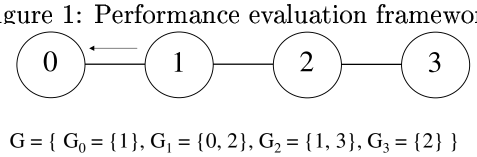

Figure 2: Topology I: A wireless network of 4 nodes.

events and the state transitions caused by these events. We then use an error-oriented test generation framework (EOTG) and topology synthesis algorithms to synthesize test scenarios along with topology information that lead to the error. The test scenarios are then used in a simulation framework to evaluate the protocol performance. The basic idea is to use a mix of forward and backward search and implication techniques to generate test scenarios that can create the target error. The implication is used to specify a partial scenario, to identify the components of the scenario that should be precluded from it, and to prune the invalid scenario whenever such components exist in the scenario.

Our overall model consists of the following:

1. Network topology: A wireless network is modeled as transmission range of each node in the network. Transmission range of node i is a set G_i where members of the set are 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. Protocol Model: A protocol message is modeled to capture meaningful state transitions the message causes. We decompose a message into its transmission and reception because its transmission effects the state of transmitter while its reception effects the states of receivers. A node transmitting the message 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. We further decompose transmission and reception into corresponding *start event* and *end event*. The protocol is

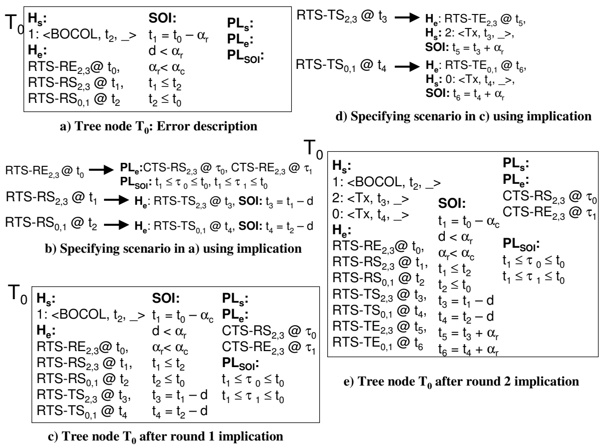

Figure 3: A collision description of a protocol.

specified using a transition table each row of which defines state transitions of network nodes for an event.

4. Error model: We define **error** as conditions to adversely effect a given protocol performance objective. It is specified in terms of states of network nodes, protocol events and time relations between state transitions and events. Given throughput, utilization, fairness, etc. as protocol performance objectives, we derive the following list of error descriptions: collision, unnecessary defer, backoff-on-failed-transmission, and silent drop. The list is not complete, however, we are currently working to design a systematic approach namely the error generation algorithms to derive a complete list.

5. Model of a scenario: A **scenario** is defined by the following: history of network node states (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), protocol events (PL_e) and time relations (PL_{SOI}) between prohibited and history entries that are prohibited or precluded from the scenario. The scenario in Figure 3.(a) presents a description of collision of $\mathrm{RTS}_{2,3}$ (Request-To-Transmit) and $\mathrm{RTS}_{0,1}$ at node 1 in topology I. While node 1 is receiving $\mathrm{RTS}_{2,3}$ during interval [t_1, t_0], it starts receiving $\mathrm{RTS}_{0,1}$ at time t_2 leading to a collision at t_2.

We present a high level description of our EOTG algorithm as follows. The detailed of our algorithms and the case studies are presented in our technical report [9]. The input error scenario is first copied to T_0, the root of the search tree. Given an event or state in a scenario, the EOTG algorithm derives unique information using backward and forward implications of the event/state. Since the derived information might already exist in the scenario (by some other causal relations), the algorithm checks if the newly generated information exist in the scenario. The information is added in the scenario if it is confirmed that the information does not exist. While adding the information to the scenario, any conflict or inconsistency between existing and newly added information is checked. In such a case the scenario is pruned. Otherwise, given the event/state, the algorithm checks 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, its predecessors are created. Child nodes of the scenario are created by creating all possible choices of the predecessors given all events/states which are not justified in the scenario. Thus the process of implication and enumeration continues until all entities in the scenario are justified and we reach at a leaf node leading to the input error condition.

Our framework is applicable for performance evaluation of all MAC protocols that uses handshaking as the basic access mechanisms, however, we only perform case studies of IEEE 802.11 [4], MACA [5], and MACAW [6], and present our results of IEEE 802.11. The test scenarios generated using our framework are simulated using network simulator ns-2 [10] to evaluate the performance of the protocol. We run simulations of the test (EOTG) scenarios on topologies shown in Figure 4 in ns-2 simulator. The arrows in the figure represent the directions of flows in the scenarios. We use CBR sources at a rate of 6 MBPS for topologies III, IV, VI, and 0.6 MBPS for the rest. We regenerate the sequence of CBR sources according to test scenarios generated automatically by our EOTG algorithm. For example, in Figure 4.(a) the start times of $1 \rightarrow 0$ flow and $3 \rightarrow 2$ flow are 5 seconds and 5.01, seconds respectively. Total simulation time is 50 seconds for all simulations. Figure 5 presents the throughput of destination nodes of individual flows and total network throughput for each of the topologies presented in Figure 4. Note that in Figure 5.(a), the average throughput of node 2 is about 50% of the average throughput of node 0. Based on the basic topology of EOTG scenario (Figure 4.(a)), we systematically construct topologies with two objectives: (1) to allow a target node to starve more (fairness), and/or (2) to allow more nodes to starve (throughput). For example in topology II (Figure 4.(b)), defer state of node 2 is extended because of the transmission from $1 \rightarrow 0$ and $3 \rightarrow 4$ resulting in almost zero throughput. Topologies III and IV are extensions where more nodes starve as shown in Figure 5. Topologies V and VI are extensions from topologies III and IV, respectively in which throughput of one of the nodes reaches almost zero. Note that in all these scenarios, all other nodes achieve average throughput except the target nodes. These results demonstrate that IEEE 802.11 is unfair in a sense that some nodes in the network starve completely while other nodes achieve average throughput. It also demonstrates that the throughput can reach to zero with the increase of 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 affect performance of TCP and real-time applications [7]. Note that channel utilization of topologies III and IV is **3.7%** and that of topology VI is **2.9%**, which are very low compared to 45-65% typically obained in previous performance studies [8].

We are currently working to automate the **error generation algorithm** (Figure 1). Input to the algorithm is a performance objective, for example, throughput, fairness, energy efficiency, etc. The algorithm uses the protocol model to generate a complete list of error descriptions, for example, collision, back-off-on-failed-transmission etc. We briefly outline the basic idea of error generation algorithm using throughput as an example performance objective. If α denotes the amount of data/payload successfully transmitted in time β, the

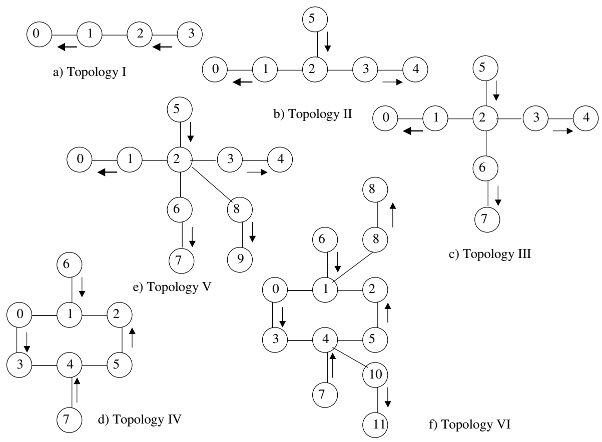

Figure 4: Example EOTG topologies simulated.

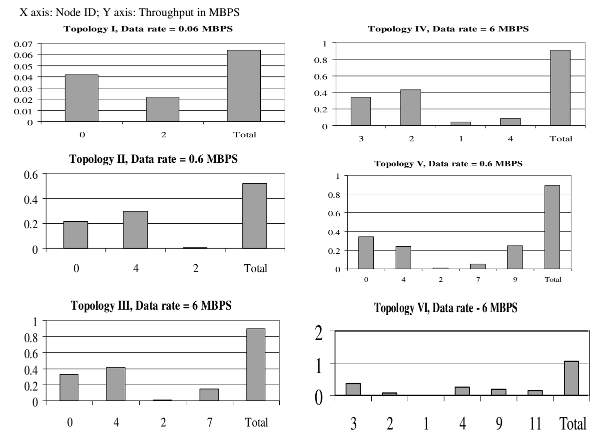

Figure 5: Average throughput of networks.

throughput is expressed as α/β. Using the parameters α and β, we derive a list of protocol events which are pre-conditions to satisfy these parameters, for example, successful data reception, etc. These conditions are used by the error generation algorithm to derive a list of *wanted conditions* depending on our study objective. For example, if our objective is to generate test scenarios that reduces throughput, then *collision*, *back-off-on-failed-transmission*, *silent-drop* would be generated as *wanted conditions*. While on the other hand, *successful acknowledgment reception* would be generated as wanted condition if our study objective is to increase throughput. These conditions are input as error condition directly to our EOTG framework for scenario generation on a given topology. Thus the automation of the error generation algorithm would allow us to analyze a broad class of protocol performance objective (that includes energy or power efficiency, for example), and in turn, to evaluate performance of a wider class of wireless MAC protocols, for example, adhoc and sensor networks.

We propose a framework for performance evaluation of wireless adhoc MAC protocols. Given a protocol performance objective, our framework systematically generates test scenarios to evaluate the given protocol performance. The future work includes an extension of our search algorithms to incorporate selection of search criteria, and therefore, to generate test scenarios that minimize (or maximize) a given protocol performance objective, for example, throughput or energy efficiency. Such a framework would be very useful to evaluate the best case or worst case performance of a broad class of wireless adhoc and sensor MAC protocols.

References

[1] 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", MASCOTS 99.

[2] S. Sharma, "Analysis of 802.11b MAC: A QoS, Fairness, and Performance Perspective", TR-126, CS, SUNY, SB, Jan '03.

[3] F. Lin, P. Chu, M. Liu, "Protocol Verification Using Reachability Analysis", CCR, vol. 17, no. 5, p. 126-135, Oct '87.

[4] 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 , pp. i -445.

[5] P. Karn, "MACA - A New Channel Access Method for Packet Radio", ARRL/CRRL, pp. 134-140, '90.

[6] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: A Media Access Protocol for Wireless LANs", SIGCOMM '94, pp. 212-225, Sep '94.

[7] C. Koksal, H. Kassab, H. Balakrishnan, "An Analysis of Short-Term Fairness if Wireless Media Access Protocols", ACM SIGMETRICS, pp 118-119, 2000.

[8] 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.

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

[10] L. Breslau et al, "Advances in Network Simulation", IEEE Computer, 33 (5), pp. 59-67, May '00.