A Model Based Platform for Design and Optimization of Multi-hop 802.11 Wireless Networks

John S. Baras
Institute for Systems Research
University of Maryland
College Park, MD, USA
baras@umd.edu

Vahid Tabatabaee*
Broadcom
San Jose,CA, USA
vahidt@umd.edu

Kaustubh Jain
Institute for Systems Research
University of Maryland
College Park, MD, USA
ksjain@umd.edu

ABSTRACT

We develop a loss model for multi-hop wireless networks based on IEEE 802.11 MAC. Given a multi-hop network topology, connection demands and routes, we model the working of 802.11 MAC in DCF mode to find good approximations to average MAC layer losses, service times and carried load. The model is defined as an implicit function amongst the variables in the model and solved using a fixed point approach. Further, using Automatic Differentiation (AD) on the implicit function, we perform sensitivity analysis and use it in an optimization framework. As an illustration of how this model can help in design and optimization of wireless networks, we optimize the network throughput by appropriate load splitting along multiple paths. We validate our models using network simulations.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design

General Terms

Design,Performance

Keywords

Loss Models, MANETs

1. INTRODUCTION

Multi-hop wireless networks have generated a lot of interest since more than a decade. However, they lack widespread deployment due to their performance issues. There are very few performance models which can aid detailed analysis of performance bottlenecks and design improvements. One of the main reasons in the difficulty in characterizing wireless 'link' capacity. Collisions and interference from neighboring transmissions make the capacity difficult to be determined. The problem is exacerbated by fading channel conditions and node mobility. Thus, the analysis, design and optimization of such systems are daunting tasks. Although use of packet level simulation tools based on PHY and MAC layer models can provide accurate performance measures, they are too complex and time consuming for analyzing the performance bottlenecks. Our objective is to develop hybrid (analytical and numerical) models which can efficiently *approximate* the performance of a wireless network. Towards this end, we focus on the MAC layer modeling and develop a fixed point loss model to evaluate connection and network throughput and packet loss for a wireless network based on 802.11 MAC protocol. These models assist us in performance analysis, design and parameter tuning procedures for wireless networks The methodology is based on development of equations that model interaction and dependency of network parameters and using a fixed point iterative method for finding a consistent solution. Then, we use Automatic Differentiation (AD) [5] for sensitivity analysis and optimize the network performance. We assume we know the exogenous traffic rate for each connection (source-destination pair), and the set of paths (with multiple paths per connections). We then optimally split the traffic of each connection amongst it's multiple paths to improve the overall network throughput.

We apply our methodology on IEEE 802.11 wireless networks and compute the network throughput, packet loss and delay parameters. For the 802.11 MAC layer modeling, modeling link layer losses involves correct modeling of transmission attempt and collision probabilities. These, in turn, need modeling of backoff evolution, packet service times and simultaneous transmission. Even for a relatively simple single-cell network, modeling of the backoff evolution and transmission attempt rates at a node is non-trivial due to the dependence on the neighboring nodes. For multi-hop scenarios, its much more complicated due to hidden nodes and different views of channels at various nodes. Starting with the seminal work Bianchi [4], and extensions by Kumar et al. [10,11], for single-cell networks, [7,8] extended the models for multi-hop scenarios. However, they had many restrictions on the topologies and connections considered and could not be used to analyze realistic scenarios. Baras et al. in [3] extended the model from [8] to arbitrary multi-hop topology. Their model obtained the loss parameters and throughput depending upon the topology and offered load. Subsequently, Jindal et al. [9] extended models from [7] to obtain achievable rate region in arbitrary multi-hop networks. In this work, we

*The author was with ISR during the time of this work.

PE-WASUN'11, November 3–4, 2011, Miami, Florida, USA.

improve the model presented in [3] to make them converge faster and give better performance estimates. We observed that the previous fixed point model [3] did not converge in a few network scenarios and gave unrealistic throughput estimates in certain others. We address those issues in this work and give an improved model. In particular, we use ensure convergence to realistic values using nested iterations. We also modify some of the equations modeling the interdependence of parameters by using better approximations. We also use the model for performance optimization and validate the performance improvements through network simulations. While Jindal's work models the different channel activities in great detail, our model make certain simplifying independence assumptions at the cost of some accuracy. Through simulations, we show, the loss in accuracy due to our simplifying assumptions is not significant and does not affect the overall goal of performance improvement. More importantly, we use sensitivity analysis to optimize the overall network throughput. We validate the performance improvements via network simulations.

Rest of the paper is organized as follows. In Section 2, we describe the network model and give an overview of our fixed point approach. We give the detailed equations relating various MAC parameters in Section 3. We give the sensitivity analysis and optimization framework in Section 4. In Section 5, we first validate the throughput estimates from our model with the empirical throughput from network simulations. Then we use the optimization framework to demonstrate the performance improvements.

2. PRELIMINARIES

2.1 Network Model and Assumptions

We study static multi-hop network topology represented by a graph $G = (V, E)$, where V and E are the set of nodes and edges. There is an edge between two nodes iff they are in the communication range of each other. A collision happens at a node if two nodes in its communication range transmit simultaneously. The network consists of a specified path set P that is used to forward traffic between the source destination pairs. The exogenous traffic arrival rate for each connection and fraction of traffic over each path is also specified. In general, we use indexes i and j for nodes and p for paths. The next and previous hops of node i in path p are shown by $h_{i,p}$ and $h^-_{i,p}$ respectively. P_i denotes the set of paths p passing through node i. C_i denotes the set of nodes that are in the communication range of node i (neighbors of i in graph G) and C_i^+ is C_i plus node i, C_i^- is the set of the nodes not in C_i^+.

For the MAC layer, we consider a slotted time system, where a time slot equals one backoff time slot of the IEEE 802.11 protocol. For simplicity, we assume that the data packets have equal length and all nodes use the same data rate. In the 802.11 protocol with RTS/CTS exchange there are two stages for packet transmission: 1) the RTS and CTS are sent between two nodes: and 2) the data packet and the ACK are sent. Different transmission failures from node i to node j or from node i over path p are represented as follows: $\beta_{i,p}$ is the probability of PHY or MAC layer transmission failure during stage 1 or 2. $\epsilon_{i,p}$ is the probability of PHY layer transmission failure during stage 2, and $l_{i,j}$ is the probability of PHY layer transmission failure at stage 1 or 2 from node i to node j. PHY layer losses, ϵ and l, are part of the input, while β is part of the model's output parameters. The other important output parameters are the probability of transmission from node i neighbors that are hidden from node j, $\theta_{i,j}$, the arrival rate of the path p packets at node i, $\lambda_{i,p}$, and the service times, $T_{i,p}$. From various $\lambda_{i,p}$, we can obtain the connection and network throughput. In obtaining the above parameters, we assume the topology does not change for till the system reaches steady-state and time averages converge. We also assume that the packet errors are independent and not bursty. In modeling the interdependence between parameters, we use some independence assumptions on the collision events so that mean of one parameter can be determined using mean of other. For example, mean transmission duration and service times can be used to determine the mean attempt rate and collision probability. We explain these independence assumption in more detail when we define the model equations.

Thus, for a given set of inputs, which include the topology graph $G = (V, E)$, the set of paths P, the exogenous traffic arrival rate for connections, fraction of traffic that is transmitted over each path and the PHY layer error probabilities, we find the average system performance. In particular, we find MAC layer packet transmission attempt rate, service times, throughput and loss rate.

2.2 Model Structure

Here, we give a brief overview of the model structure that approximate the average behavior of a multi-hop wireless network based on the IEEE 802.11 MAC protocol. However, we will present the detailed set of equations in the next section. The model structure enables us to find a consistent solution for the given set of non-linear equations. The derived equations are used iteratively in this structure to find a consistent solution for all parameters. Let $\underline{Y}$ be the set of parameters and $\underline{F}(\underline{Y})$ be the set of equations that express the average system behavior by approximating each parameter in terms of the other parameters. One common way to find a solution to the given set of equations is to use fixed point iterations,

$$\underline{Y}^{new} = \underline{F}(\underline{Y}) \tag{1}$$

This approach was used in [2, 3]. However, on further investigation, we observed that for large networks when the arrival rate is close or above the network capacity, the parameters may not converge. In some other scenarios, the fixed point algorithm converges to give unrealistic values. A single-loop fixed point algorithm oscillate or converges to unrealistic values because of the implicit dependence of losses at a node and service times of its neighboring nodes.Hence, we use a two-step algorithm to find link losses and service times. First, we fix β and θ, and run a fixed point algorithm to find the service times T, and other parameters. Then, using these parameters, we update β and θ. Hence, we update the parameters iteratively in two nested loops as follows:

Let $\underline{X}$ be the set of parameters excluding $\underline{\theta}$ and $\underline{\beta}$ and $F_{\underline{X}}(\underline{X}, \underline{\beta}, \underline{\theta})$, $F_{\underline{\theta}}(\underline{X})$ and $F_{\underline{\beta}}(\underline{X}, \underline{\theta})$ be the set of equations that are derived to approximate $\underline{X}, \underline{\theta}$, and $\underline{\beta}$ respectively. The iterative algorithm is described in the Algorithm 1 pseudo code. The inner while loop updates $\underline{X}$ parameters using a fixed point iteration. The outer loop updates $\underline{\theta}$ using a convex combination (weighted average) of previous and new value, where ϵ specifies the weight. The updated values of $\underline{\theta}$ and $\underline{X}$ are then used to update $\underline{\beta}$. The convergence criteria in both cases is based on the maximum difference in old and

Algorithm 1 FPA Model

Initialize: $(\underline{\beta}, \underline{X}, \underline{\theta})$
while $\underline{\beta}, \underline{\theta}$ are not converged **do**
  while $\underline{T}$ is not converged **do**
    $\underline{X}^{new} = F_{\underline{X}}(\underline{X}, \underline{\beta}, \underline{\theta})$
    $\underline{X} = \underline{X}^{new}$
  end while
  $\underline{\theta}^{temp} = F_{\underline{\theta}}(\underline{X})$
  $\underline{\theta}^{new} = \epsilon \underline{\theta}^{temp} + (1-\epsilon)\underline{\theta}$
  $\underline{\beta}^{temp} = F_{\underline{\beta}}(\underline{X}, \underline{\theta}^{new})$
  $\underline{\beta}^{new} = \epsilon \underline{\beta}^{temp} + (1-\epsilon)\underline{\beta}$
  $\underline{\theta} = \underline{\theta}^{new}$
  $\underline{\beta} = \underline{\beta}^{new}$
end while

new computed values of the specified parameters, which are $\underline{\theta}$ and $\underline{\beta}$ for the outer loop and $\underline{T}$ for the inner loop.

We need to update θ and β with memory, otherwise the *FPA* algorithm does not converge when the network load is high. As an example, suppose we start with low initial values for θ. This results in low estimates of the hidden nodes effect and low losses. Hence, the estimated traffic rates over links will be high. In the next iteration, the high traffic rates, results in high θ values, which in turn results in high losses and low traffic rates. The low traffic rate estimates, results in low θ values in the next iteration. Therefore, fixed point equations, oscillate between high θ, low data rate, and low θ high rate regimes. Hence, we have to update θ parameters in small step sizes. We tested *FPA* model with many scenarios. In all the scenarios, the algorithm converged to give realistic solution.

3. THE MAC LAYER EQUATIONS

In this section we provide the set of equations that we use to approximate the wireless links average loss parameters and service times. We first present the set of equations that are used in the fixed point iteration (inner loop) and then the outer loop equations for updating the θ and β parameters. These equations are similar to the one introduced in [3], however, we change some of the equations and the underlying independence assumptions to make them more accurate.

3.1 Inner loop (fixed point) equations:

The set of equations that are numbered are used in the fixed point iterations and the estimated parameter in the left hand side of the equations are specified with superscript *'new'*. Hence, the following equation specifies that it is used in the fixed point iteration to update $y_{i,p}$ variables.

$$y_{i,p}^{new} = f(\underline{X}) \tag{2}$$

The unnumbered equations are auxiliary and used in computation of fixed point equations. These computations are carried out for every path p and node i in the network.

3.1.1 *Link Loss and Hidden Nodes Modeling:*

To obtain the transmission failure probability, we first need to characterize the probability of a node accessing the channel, assuming that this node is scheduled to serve a packet on the path p. Let W and M be respectively the minimum and maximum sizes of 802.11 back-off windows respectively. Then L, the number of back-off stages which the window size reaches its maximum value, is $L = \log_2 \frac{M}{W}$. Let m be the maximum retry limit. Suppose that node i is scheduled to serve a packet on path p. Assuming that node accesses the channel with a fixed probability $\alpha''_{i,p}$, and setting $\beta = \beta_{i,p}$, we can use the following relation derived from Eq. (1) of [11]:

$$\alpha''^{new}_{i,p} = \begin{cases} \frac{2(1-2\beta)(1-\beta^{m+1})}{(1-2\beta)(1-\beta^{m+1})+W(1-\beta-\beta(2\beta)^L(1+\beta^{m-L}(1-2\beta)))}, \\ \qquad \text{if } L < m \\ \frac{2(1-2\beta)(1-\beta^{m+1})}{(1-2\beta)(1-\beta^{m+1})+W(1-\beta)(1-(2\beta)^{m+1})}, \\ \qquad \text{if } L \geq m \end{cases} \tag{3}$$

The scheduler behavior is, then, specified by the scheduler coefficient $k_{i,p}$, which is the average serving rate of the path p packets at node i and is given by the following equations:

$$k_{i,p}^{new} = \begin{cases} \lambda_{i,p} & \text{if } \sum_{p' \in P_i} \lambda_{i,p'} T_{i,p'} \leq 1 \\ \frac{\lambda_{i,p}}{\sum_{p' \in P_i} \lambda_{i,p'} T_{i,p'}} & \text{otherwise} \end{cases} \tag{4}$$

The total average throughput $\bar{\rho}_i$, of node i, is $\bar{\rho}_i = \sum_{p \in P_i} k_{i,p} T_{i,p}$. Note that if $\sum_{p' \in P_i} \lambda_{i,p'} T_{i,p'}$, which is the required utilization at node i to serve total incoming traffic, is less than one then we can serve all packets and the scheduling rate equals the arrival rate. On the other hand if this condition is not satisfied the scheduling rate should be adjusted to make sure that the utilization does not exceed 1. Here, we have assumed that the scheduling rate is proportional to the arrival rate for each path. This assumption is reasonable and valid for scheduling policies such as FCFS scheduling policy.

Let, $v_{i,p}$ be the channel holding time by node i when it attempts to transmit path p packets to node $h_{i,p}$. There are two different components in $v_{i,p}$: (i) the average holding time τ_P when the attempted transmission is successful and (ii) the average holding time $f_{i,p}$ when attempted transmission fails. We have,

$$f_{i,p}^{new} = \frac{\epsilon_{i,p}}{\beta_{i,p}} \tau_P + \left(1 - \frac{\epsilon_{i,p}}{\beta_{i,p}}\right) \tau_H \tag{5}$$

The first and second terms are the average channel holding times when there are transmission failures in the data packet/ACK and RTS/CTS failure respectively:

$$\tau_P = T_{\text{RTS}} + \text{SIFS} + T_{\text{CTS}} + \text{SIFS} + T_{\text{P}} + \text{SIFS} + T_{\text{ACK}} + \text{DIFS}$$
$$\tau_H = T_{\text{RTS}} + \text{SIFS} + T_{\text{CTS}} + \text{SIFS}$$

where SIFS and DIFS are IEEE 802.11 parameters and T_{RTS}, T_{CTS}, T_{P} and T_{ACK} are the transmission times for RTS, CTS, data and ACK on the corresponding connection respectively. We have neglected the propagation delay between the two nodes for simplicity. Now, we can compute the average holding time $v_{i,p}$,

$$v_{i,p}^{new} = (1 - \beta_{i,p}^m)\tau_P + \frac{1-\beta_{i,p}^m}{1-\beta_{i,p}} \beta_{i,p} f_{i,p} \tag{6}$$

The average loss factor for path p transmission at node i is $\beta_{i,p}^m$ since after m transmission failures a packet is discarded.

Hence, the arrival rate of path p packets to node $h_{i,p}$ is.

$$\lambda^{new}_{h_{i,p},p} = k_{i,p}(1-\beta^{m}_{i,p}) \text{ for all } i, p \in \Pi_i \quad (7)$$

The arrival rate for the first-hop of the path is determined by the exogenous arrival rate and the routing policy and from the fixed point perspective is fixed and given by:

$$\lambda_{i,p} = \text{Exogenous arrival rate for path } p \text{ if } i \text{ is the first node of } p \quad (8)$$

The average throughput of path p at node i is the fraction of time that node i is busy serving transmission of path p packets.

$$\rho^{new}_{i,p} = k_{i,p}T_{i,p} \quad (9)$$

The attempt rate of node i for path p, $\alpha''_{i,p,j}$, as given in (3) is conditional to node i being scheduled to serve path p packets. The unconditional attempt rate is,

$$\alpha^{new}_{i,p,i} = \rho_{i,p}\alpha''_{i,p} \quad (10)$$

Consider a node j that is neighbor of a node i. The access (attempt) rate of node i for path p packets as observed by node j is different from the access rate of node i given in (10), since node j does not know about transmission from i neighbors that are hidden from it. Hence, $\alpha_{i,p,j}$, the attempt rate of path p packets at node i as observed by node j is,

$$\alpha^{new}_{i,p,j} = \rho_{i,p}(1-\theta_{i,j})\alpha''_{i,p} \text{ for all } j \in C_i \quad i \neq j \quad (11)$$

where θ_{ij} is the probability of hidden node transmissions.

3.1.2 The Serving Time Components

The average serving time of a path p packet at node i, $T_{i,p}$, consists of 4 components as follows:

- $s_{i,p}$: average time for successful transmission of path p packets at node i.
- $u_{i,p}$: average time for successful transmission to and from node i neighbors.
- $b_{i,p}$: average back-off time of node i.
- $c_{i,p}$: average time of unsuccessful transmission due to PHY layer errors and collisions at the MAC layer in the neighborhood of node i.

The probability of successful transmission of a packet of path p at node i is $(1-\beta^{m}_{i,p})$. And successful transmission time for a packet is τ_P. Hence, the first component is,

$$s^{new}_{i,p} = (1-\beta^{m}_{i,p})\tau_P \quad (12)$$

Let CW_n be the back-off window size at stage n, then $W_n = CW_n/2$ is the average back-off time at stage n. The average back-off time is,

$$b^{new}_{i,p} = \sum_{n=0}^{m} W_n\beta^{n}_{i,p} \quad (13)$$

Now, we compute $u_{i,p}$, the average time of successful transmission of node i neighbors. The probability of successful transmission of node i, when it is scheduled to transmit path p packets is,

$$q^{new}_{i,p} = (1-\beta_{i,p})\alpha''_{i,p}$$

Assuming that successful transmission of neighbors are independent events, the probability of successful transmission to and from neighborhood of i, which is scheduled to transmit path p packets is,

$$r_{i,p} = 1-(1-q_{i,p})\prod_{j\in C_i}\left(1-(S^1_j+S^2_j)(1-\theta_{ji})\right)$$

where

$$S^1_j = \sum_{p'\in P_j} q_{j,p'}\rho_{j,p'} \quad \text{and}$$

$$S^2_j = \sum_{p'\in P_j, h^{-1}_{j,p'}\in C^-_i}\left(q_{h^{-1}_{j,p'},p'}\rho_{h^{-1}_{j,p'}}(1-\theta_{h^{-1}_{j,p'},i})\right)$$

S^1_j is for transmission from neighbor j and S^2_j is for transmission to neighbor j from nodes that are not i neighbors. Note that node i is aware of transmissions to its neighbors from the CTS and ACK messages.

The probability that the next successful transmission is from node i given that there has been a successful transmission in its neighborhood is,

$$\gamma_{i,p} = \frac{q_{i,p}}{r_{i,p}}$$

Let $Q_{i,p}$ be the average number of successful transmissions in the neighborhood of i per successful transmission from node i,

$$Q_{i,p} = \frac{1-\gamma_{i,p}}{\gamma_{i,p}}$$

Then, $u_{i,p}$ is.

$$u^{new}_{i,p} = Q_{i,p}\tau_P \quad (14)$$

For $c_{i,p}$ we need to compute: (1) $x_{i,p}$ the probability of successful transmission of node i given that there is at least one transmission in its neighborhood and (ii) $y_{i,p}$ the probability that a transmission fails in the neighborhood of i given there has been a transmission. Let $z_{i,p}$ be the probability of at least one transmission in the neighborhood of node i, when it is scheduled to transmit path p packets:

$$z_{i,p} = 1-(1-\alpha''_{i,p})\prod_{j\in C_i}\left(1-(\sum_{p'\in P_j}\alpha''_{j,p'}\rho_{j,p'})(1-\theta_{ji})\right)$$

The probabilities $x_{i,p}$ and $y_{i,p}$ can be computed using conditional probability rules,

$$x_{i,p} = \frac{q_{i,p}}{z_{i,p}} \quad \text{and} \quad y_{i,p} = 1-\frac{r_{i,p}}{z_{i,p}}$$

Then, the average number of collisions during transmission time is $y_{i,p}/x_{i,p}$ and the average collision time is

$$c^{new}_{i,p} = \frac{y_{i,p}}{x_{i,p}}w_{i,p} \quad (15)$$

where $w_{i,p}$ is the average time consumed for failure transmissions in the neighborhood of i:

$$w_{i,p} = \frac{\alpha''_{i,p}\beta_{i,p}f_{i,p} + \sum_{j\in C_i} S^3_j(1-\theta_{j,i})f_{j,p'}}{\alpha''_{i,p}\beta_{i,p} + \sum_{j\in C_i} S^3_j(1-\theta_{j,i})}$$

where $S^3_j = \sum_{p'\in P_j}\alpha''_{j,p'}\beta_{j,p'}\rho_{j,p'}$

Now, we have computed all 4 components of the average transmission time and can update it:

$$T^{new}_{i,p} = s^{new}_{i,p} + u^{new}_{i,p} + b^{new}_{i,p} - c^{new}_{i,p} \quad (16)$$

In the fixed point iteration, after computing the new variables, we calculate the maximum difference between the new and old computed average transmission times:

$$\Delta_{\text{inner loop}} = \max_{i,p} \left|T_{i,p}^{new} - T_{i,p}\right|$$

If $\Delta_{\text{inner loop}} < 1$, we consider that the fixed point iteration is converged, otherwise the inner loop is repeated.

3.2 Outer loop Computations:

After convergence of the fixed point parameters in the inner-loop, we use them to update the θ and β parameters in the outer loop. Recall that $h_{n,p}$ and $h_{n,p}^{-}$ are the next and previous hops of node n in path p respectively. Then, $\theta_{i,j}$, probability of transmission from neighbors of i that are hidden from j (when there is no detected transmission in the neighborhood of j), is:

$$\theta_{i,j}^{temp} = 1 - \prod_{n \in C_i \cap C_j^{-}} \left(1 - \frac{S_n^4 + S_n^5}{1 - S_n^6}\right) \qquad (17)$$

where

$$S_n^4 = \sum_{p' \in P_n, h_{n,p'} \in C_j^{-}} \left(\frac{v_{n,p'}}{T_{n,p'}} \rho_{n,p'}\right)$$

$$S_n^5 = \sum_{p' \in P_n, h_{n,p'}^{-} \in C_i^{-} \cap C_j^{-}} \left(\frac{v_{n,p'}}{T_{n,p'}} \rho_{n,p'}\right)$$

$$S_n^6 = \sum_{p' \in P_n, h_{n,p'} \in C_j^{+}} \left(\frac{v_{n,p'}}{T_{n,p'}} \rho_{n,p'}\right)$$

S_n^4 is the probability of transmission from hidden node n to another node not in the vicinity of node j. S_n^5 is the transmission probability to hidden node n which also keeps the channel busy (due to CTS). S_n^6 is the probability of transmission from hidden node n to neighbor of node j, of which the node j is aware-of.

As mentioned in Section 2.2, we update θ with some memory to avoid oscillations. To that end, at each iteration, the new value of θ is convex combination of the old value and the *temp* value derived in (17),

$$\theta_{i,j}^{new} = \epsilon\theta_{i,j}^{temp} + (1-\epsilon)\theta_{i,j} \qquad (18)$$

In our model, we set $\epsilon = 0.1$, so that the θ parameter changes are gradual and the parameters converge. We use the computed θ^{new} parameters to update the link loss parameters β in a similar way,

$$\beta_{i,j}^{temp} = (1 - l_{i,h_{i,p}})(1 - \theta_{h_{i,p},i}^{new}) \times \prod_{j \in C_{h_{i,p}}^{+} \cap C_i} \left(1 - \sum_{p' \in P_j} \alpha_{j,p',h_{i,p}}\right) \times \prod_{j \in C_{h_{i,p}}^{+} \cap C_i^{-}} \left(1 - \sum_{p' \in P_j} \alpha_{j,p',h_{i,p}}\right)^{V_{i,p}} \qquad (19)$$

The first product term is the probability of no new transmission from those neighbors of $h_{i,p}$ that are not hidden from i, In the second product term, $V_{i,p} = T_{\text{RTS}}(i,p) + \text{SIFS}$ is the vulnerable period during which those neighbors of $h_{i,p}$ that are hidden from i are not aware of the ongoing transmission and may cause collision. Therefore, the second product term is the probability of no new transmission from those neighbors of $h_{i,p}$ that are hidden from i during the vulnerable period. Then we update β with memory,

$$\beta_{i,p}^{new} = \epsilon\beta_{i,p}^{temp} + (1-\epsilon)\beta_{i,p} \qquad (20)$$

After computation of new θ and β variables in the outer loop, we compute the maximum difference between the new and old values,

$$\Delta_{\text{outer loop}} = \max_{i,j,p} \left((\theta_{i,j}^{new} - \theta_{i,j}), (\beta_{i,p}^{new} - \beta_{i,p})\right) \qquad (21)$$

The iterative algorithm is continued until $\Delta_{\text{outer loop}} < 0.01$. At this stage the FPA model is considered converged and the parameters are used to estimate system performance. In particular, offered and carried load for each path can be obtained from $\lambda_{src,p}$ and $\lambda_{dest,p}$, respectively, where *src* and *dest* are the source and destination for path p. Denoting by P_c the set of paths used in connection c, the throughput of connection c between *src* and *dest* is obtained as

$$T_c = \frac{\sum_{p \in P_c} \lambda_{dest,p}}{\sum_{p \in P_c} \lambda_{src,p}} \qquad (22)$$

4. SENSITIVITY ANALYSIS AND NETWORK OPTIMIZATION

Estimating the system performance using the solution to the system of equations is not sufficient for systematic design and trade-off analysis of systems. For the latter we also need to quantify reliability and robustness of the solution using sensitivity and perturbation analysis. However, due to the complexity of relations of the component models, it is not possible to compute the derivatives analytically. Numerical methods such as Automatic Differentiation (AD) can be used to obtain the derivatives and sensitivity parameters. AD [5] provides the partial derivative of the performance metric (e.g. throughput) with respect to the input parameters (i.e., design variables or parameters). We can use the computed derivatives in gradient-based optimization methods to improve the performance. We use optimal routing design as an example to illustrate our proposed design methodology. But trade-off analysis and optimization can be carried out with many different objectives. For example, throughput sensitivity to backoff window size, retry limit, etc. can be obtained using AD. Using these sensitivity and gradient values, we can find optimal backoff window size. This setup is similar to the one introduced in [3], but we summarize the method.

Given the network topology and traffic demands, we find multiple paths and optimally split traffic along these multiple paths so as to achieve optimal network throughput. We implement the Dreyfus K-shortest path algorithm [6] for path selection. We use the gradient projection method to find the optimal values for the routing probabilities to maximize the network throughput. The gradient projection method requires iterative computation of the throughput gradient. The fixed point method provides a computational scheme that, after convergence, describes the performance metric (i.e. throughput) as an implicit function of the design parameters (i.e. routing parameters). Since we do not have analytic expressions of the performance metric, we use ADOL-C [1] for obtaining the parameter sensitivities using AD.

Now we present the methodology to optimize the overall throughput in the network by changing the path probability distribution of each connection on the network. The total network throughput T is:

$$T = \frac{\sum_{c \in C} (\sum_{p \in P_c} \lambda_{last,p})}{\sum_{c \in C} (\sum_{p \in P_c} \lambda_{first,p})} \quad (23)$$

Then, assuming there are $m = |C|$ active connections in the network, n_c paths used in the connection c and denoting by $\pi_{i,c}$ the probability associated with using path i in connection c, we know that the total throughput is a function of these input probabilities, namely: $T = T(\pi_{1,c_1}, \cdots, \pi_{n_{c_1},c_1}, \cdots, \pi_{n_{c_m},c_m})$. Thus, the optimization problem is:

$$\begin{aligned} & max\ T = T(\pi_{1,c_1}, \cdots, \pi_{n_{c_1},c_1}, \cdots, \pi_{n_{c_m},c_m}) \\ & \text{subject to} \sum_{i \in P_c} \pi_{i,c} = 1, \quad \forall c \in C \quad (24) \\ & \pi_{i,c} \geqslant 0, \quad \forall (i,c) \in P_c \times C \end{aligned}$$

We solve the optimization problem using gradient projection method. Denoting by $\overline{\nabla}_c$ the connection c average gradient, and by $\eta > 0$ the step size, the route probabilities are iteratively updated, to find the optimal solution, as follows:

$$\pi_{i,c_k} = max\ (0, \pi_{i,c_k} + \eta(\frac{\delta T}{\delta \pi_{i,c_k}} - \overline{\nabla}_{c_k})), \forall k \in \{0, \cdots, m\} \quad (25)$$

5. RESULTS

We run two sets of simulation studies. We first validate the model by comparing the FPA throughput estimates for different network scenarios with those obtained through network simulations. We use OPNET Modeler 14.5 for network simulation. For each scenario under study, we increase the offered load on each connection proportionally, and see how much of the load is carried to the destination. In the second set, we use AD to optimize the overall network throughput. In particular, we find the optimal routing probability / traffic splits between different routes so that the overall network throughput is maximized. We also validate the performance improvements from optimal route splitting through network simulations.

In each of the scenarios, the network topology is specified along with the traffic demands (offered load) along with the multihop paths for each of the connections. The nodes positions are fixed for each scenario. In order to focus on the MAC, we assume unit disc model for communication range with zero PHY losses within the disc. Hence the links can be specified using an adjacency matrix. The traffic demands are specified by setting up UDP traffic at the source nodes. The traffic source generates packets of fixed length (8384 bits) and with constant inter-arrival time, the interval depending upon the desired load. We modify the MANET routing protocol to use pre-specified static routes instead of standard routing protocols like OLSR or AODV. We use the 802.11 b MAC with 11 Mbps data rate. However, the control messages (RTS, CTS, ACK) are sent at 1Mbps. The minimum and maximum contention window size is 32 and 1024, and the maximum retry limit is 4. The RTS threshold is set to a low value so that the RTS-CTS mode is always used. We use the free-space propagation model for wireless channel. Zero PHY layers losses (unit disc model) are achieved by modifying the BER curves. If transmission of two packets overlaps, they are declared as lost.

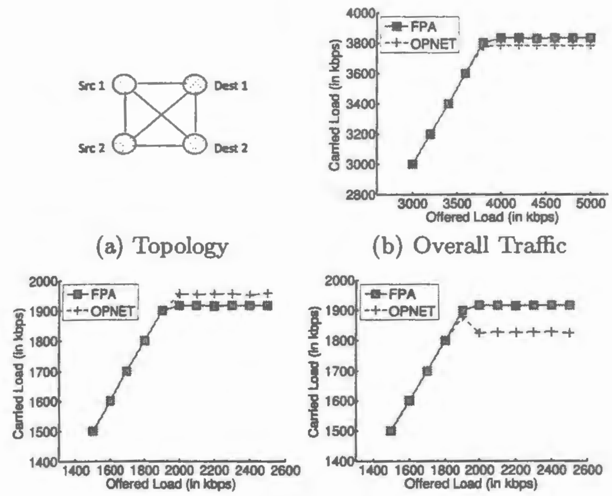
5.1 Validation of Throughput Approximation Models

At first, we use simple scenarios to validate the 802.11 MAC model. We look at different two-edge topologies as shown in Figs. 1(a),2(a),3(a),4(a). The edges in the topology graphs show the communication and interference links. These four scenarios cover the different possibilities of interference between two simultaneous transmissions and can be seen as building blocks for more complicated scenarios [9]. In each scenario, we assign equal traffic on both the sources and increase the traffic proportionally till the load is high. We do not study very high load scenarios where the traffic demand exceeds the network capacity and system goes into unstable region. For the coordinated nodes scenario, Fig. 1 shows how the FPA model throughput matches closely with the simulated one. Connection 1 and 2 should have same average throughput due to the symmetry. But due to the limitations of the simulator, connection 2 achieved slightly lower throughput, which explains the larger difference in Fig 1(d). However, the overall (and the average) throughput is approximated very well by the model (Fig. 1(b)). Figs. 2, 3 and 4, show the corresponding results for the other three topologies. In all the scenarios, the FPA throughput gives good approximations to the throughput obtained from network simulations. The lower throughput for connection 1 in Fig. 2(c) can be attributed to higher estimates for hidden node transmission probability (θ). At the worst, the FPA throughput shows an error of 20 %. However, in all the scenarios, the FPA model follows the trends of increasing (or decreasing) traffic demands. They give good approximation to capacity (or saturation) throughput in those scenarios. This is an improvement from the previous model of [3], which in some scenarios gave very high throughput. For example, in Far Hidden Node scenario, the FPA would converge to give negligible link losses. However, the newer model does not suffer from this problem.

Next, we study a 30-node multi-hop scenario shown in Fig. 5. The overall carried load in FPA model matches well with OPNET. Table 1 shows the various connections, and Figs. 6, 7 and 8 show carried load for some of the connections. Single-hop connections achieve 100 % throughput for both FPA and OPNET (plots not shown). In this scenario, offered load on individual connections are not necessarily equal. The plots show that the carried load for individual connections is approximated reasonably well by the FPA. Some connections show higher approximation error due to the independence assumptions used in the model. However, the capacity for the overall network and longer connections is approximated well. We validate the FPA model in many other scenarios. In all the scenarios, the fixed point algorithm converged quickly to give good approximation to connection and network throughput.

5.2 Throughput Optimization Using AD

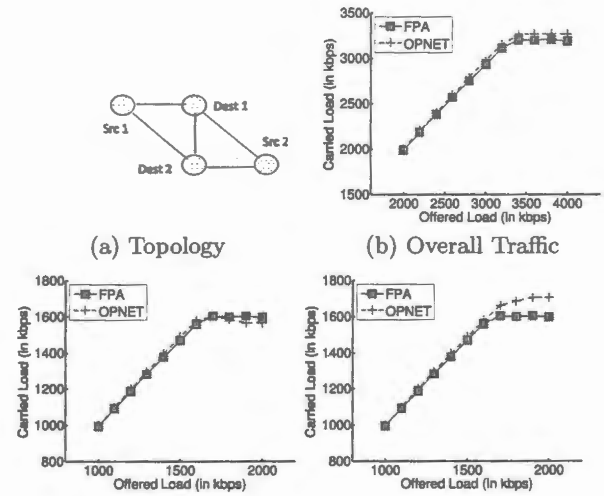
Now we set multiple paths per connections in the 30-node scenario, and use the optimization framework (Sec. 4) to demonstrate the utility of the FPA model. At first we assign equal traffic splits between multiple paths and compare



(a) Topology (b) Overall Traffic

(c) Connection 1 (d) Connection 2

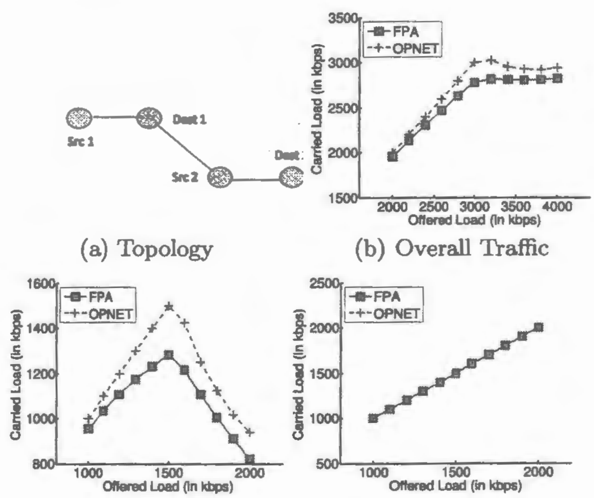
Figure 1: Coordinated Nodes Scenario



(a) Topology (b) Overall Traffic

(c) Connection 1 (d) Connection 2

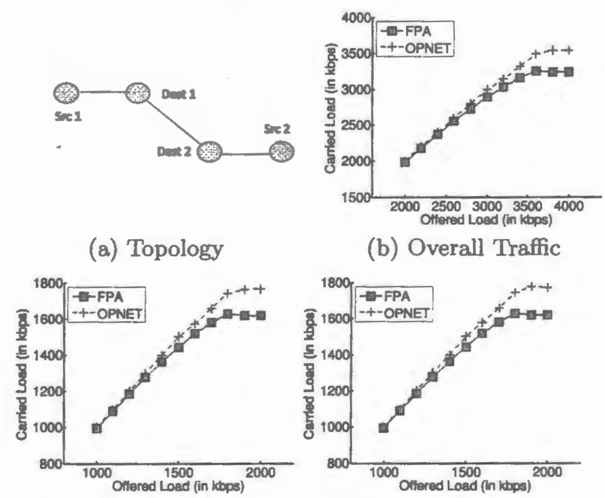
Figure 3: Near Hidden Nodes Scenario



(a) Topology (b) Overall Traffic

(c) Connection 1 (d) Connection 2

Figure 2: Asymmetric Nodes Scenario



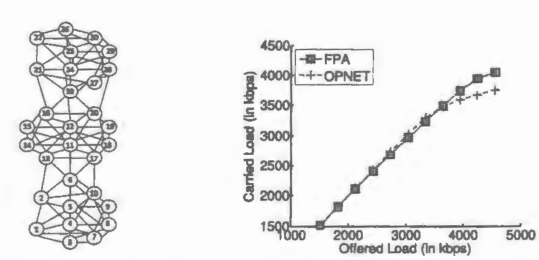
(a) Topology (b) Overall Traffic

(c) Connection 1 (d) Connection 2

Figure 4: Far Hidden Nodes Scenario

Connections	1-hop	2-hop	3-hop	5-hop
(Src, Dest) pairs	(2,4), (5,7), (17,12), (18,19), (24,29), (24,26)	(3,10), (8,6), (11,2), (15,18), (20,13) (21,12), (21,30), (22,23)	(2,19), (22,11)	(21,1)

Table 1: Connections for the 30 node scenario



(a) Topology (b) Overall Carried Load

Figure 5: 30-node scenario

the FPA and OPNET traffic. Then we run optimization algorithm using AD in the FPA model to find optimal traffic splits to maximize the network throughput. Using these optimized traffic splits, we run the OPNET simulations to obtain the newer throughput. As can be seen from Fig. 5, the carried load in FPA approximates well the actual carried load. Moreover, despite the small error in the throughput approximation, route-split optimization leads to improved performance in network simulations. Again, similar experiments with different topologies yielded the same results: predicted increase in throughput due to route-split optimization in FPA model are validated through network simulations.

6. CONCLUSION AND FUTURE WORK

We presented a loss model for design and optimization of multi-hop wireless networks. Given any static multi-hop topology and offered load on various connections, the model finds average throughput, link losses and other MAC pa-

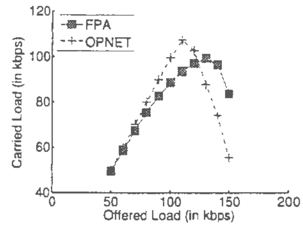
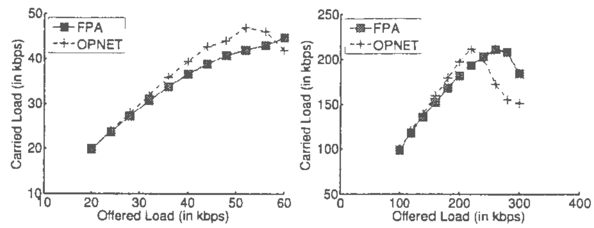
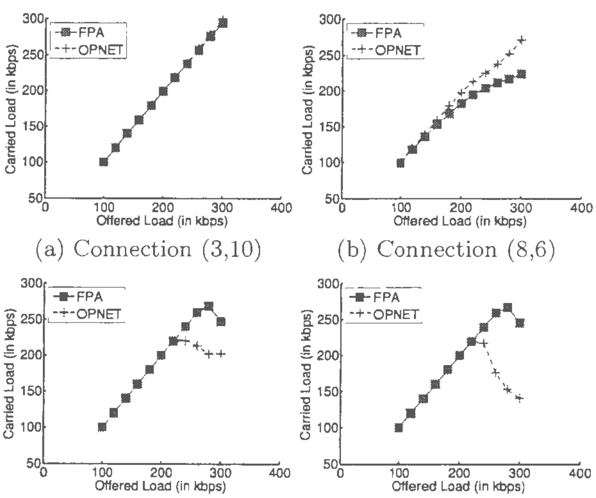


Figure 6: 5-hop connection in 30-node scenario



(a) Connection (2,19) (b) Connection (22,11)

Figure 7: 3-hop connections in 30-node scenario



(a) Connection (3,10) (b) Connection (8,6)

(c) Connection (21,30) (d) Connection (22.23)

Figure 8: 2-hop connections in 30-node scenario

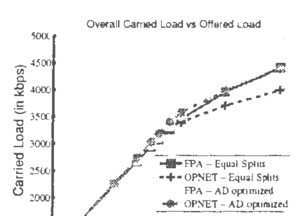


Figure 9: Optimized Carried Load in 30-node scenario

rameters through a fixed point approach. The FPA algorithm converges to give good performance estimates. We validated our model by comparing the throughput estimates with the empirical throughput from network simulations. We also demonstrated the utility of the model in network performance optimization, by performing sensitivity analysis based on model parameters. Using AD, we saw how the network throughput varies by rerouting some traffic along multiple paths. We demonstrated that by optimizing the traffic splits using the FPA model, the network throughput can be increased. We also validated the performance improvements using network simulation results.

Acknowledgement

Research supported by the Communications and Networks Consortium sponsored by the U.S. Army Research Laboratory under the Collaborative Technology Alliance Program Cooperative Agreement DAAD19-01-2-001.

7. REFERENCES

[1] Automatic differentiation by overloading in c++. URL.

[2] J. S. Baras, V. Tabatabaee, G. Papageorgiou, and N. Rentz. Modelling and Optimization for Multi-hop Wireless Networks Using Fixed Point and Automatic Differentiation. In *Int. Symp. on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks*, 2008.

[3] J. S. Baras, V. Tabatabaee, G. Papageorgiou, and N. Rentz. Performance Metric Sensitivity Computation for Optimization and Trade-off Analysis in Wireless Networks. In *Proceedings of the IEEE Global Communications Conference*, 2008.

[4] G. Bianchi. Performance Analysis of the IEEE802.11 Distributed Coodination Function. *IEEE Journal on Selected Areas in Communications*, 18(3):535–547, March 2000.

[5] H. M. Bücker, G. F. Corliss, P. D. Hovland, U. Naumann, and B. Norris, editors. *Automatic Differentiation: Applications, Theory, and Implementations*, volume 50 of *Lecture Notes in Computational Science and Engineering*. Springer, New York, NY, 2005.

[6] S. E. Dreyfus. An appraisal of some shortest path algorithms. 17(3):395–412, May 1969.

[7] M. Garetto, T. Salonidis, and E. W. Knightly. Modeling per-flow throughput and capturing starvation in CSMA multi-hop wireless networks. *IEEE/ACM Trans. Netw.*, 16(4):864–877, 2008.

[8] M. Hira, F. Tobagi, and K. Medepalli. Throughput Analysis of a Path in an IEEE 802.11 Multihop Wireless Netwoks. In *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, Hong Kong, China, March 2007.

[9] A. Jindal and K. Psounis. The achievable rate region of 802.11-scheduled multihop networks. *IEEE/ACM Trans. Netw.*, 17(4):1118–1131, 2009.

[10] A. Kumar, E. Altman, D. Miorandi, and M. Goyal. New Insights from a Fixed Point Analysis of Single Cell IEEE 802.11 WLANs. In *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, 2005.

[11] A. Kumar, E. Altman, D. Miorandi, and M. Goyal. New insights from a fixed-point analysis of single cell ieee 802.11 wlans. *IEEE/ACM Trans. Netw.*, 15(3):588–601, 2007.