Performance Metric Sensitivity Computation for Optimization and Trade-off Analysis in Wireless Networks

John S. Baras, Vahid Tabatabaee, George Papageorgiou, and Nicolas Rentz
Institute for Systems Research, University of Maryland, College Park, MD 20742
Email: {baras, vahidt, gpapag, nrentz}@umd.edu

***Abstract*—We develop and evaluate a new method for estimating and optimizing various performance metrics for multi-hop wireless networks, including MANETs. We introduce an approximate (throughput) loss model that couples the physical, MAC and routing layers effects. The model provides quantitative statistical relations between the loss parameters that are used to characterize multiuser interference and physical path conditions on the one hand and the traffic rates between origin-destination pairs on the other. The model takes into account effects of the hidden nodes, scheduling algorithms, IEEE 802.11 MAC and PHY layer transmission failures and finite packet transmission retries at the MAC layer in arbitrary network topologies where multiple paths share nodes. We apply Automatic Differentiation (AD) to these implicit performance models, and develop a methodology for sensitivity analysis, parameter optimization and trade-off analysis for key wireless protocols. Finally, we provide simulation experiments to evaluate the effectiveness and performance estimation accuracy of the proposed models and methodologies.**

I. Introduction

Multi-hop wireless networks lack widespread commercial deployment due to the lack of systematic methodologies and tools for the efficient design and dimensioning of such networks. The reason for the absence of such tools is the different nature of wired and wireless networks rendering the use of wired network techniques inappropriate for the case of wireless networks. Key quantities, such as the link capacity, that remain constant in a wired network, vary in wireless communication environments with the transmission power, the interference, the node mobility and the channel condition. Thus, model-based performance evaluation tools are needed to assist wireless network engineers and researchers to efficiently design wireless networks.

Packet level simulation of multi-hop wireless networks with the appropriate PHY and MAC layer modelling turns out to be too complex and time consuming for the design and analysis of wireless networks in realistic settings. Our objective is to develop low complexity analytical and computational (numerical) models, which can efficiently *approximate* the performance of wireless networks. Such models can be used for protocol analysis, component based design, parameter tuning and network management and provisioning.

We propose an alternative approach based on the fixed point method and loss network models for performance evaluation and optimization. Loss network models [1] were originally used to compute blocking probabilities in circuit switched networks [2] and later were extended to model and design ATM networks [3]–[6]. In [3] reduced load approximations were used effectively to evaluate quite complex ATM networks, with complex and adaptive routing protocols, and multi-service multi-rate traffic (different service requirements). The main challenge in developing loss network models for wireless networks is the coupling between wireless links. This coupling is due to the transmission interference between different nodes in proximity with each other.

Furthermore, we perform sensitivity analysis to evaluate the resilience and robustness of the solution. For this, we use Automatic Differentiation (AD), which is a powerful method to numerically compute the derivatives of a software-defined function. The generated *implicit* analysis model, based on the fixed point iterations, is the input to the AD. The AD provides the partial derivative of the performance metric (e.g. throughput) with respect to defined input parameters (i.e. design variables or parameters). This method allows for very complex design parameters to be implicitly embedded in the input function to the AD module. We use this methodology to compute the optimal load distribution among multiple paths to maximize the network throughput.

In his seminal work [7], Bianchi considers saturated users with ideal (no channel losses) and homogenous (equal physical data rate) channel conditions for 802.11 MAC layer modeling. The analysis works when there are no hidden terminals, which results in synchronous channel conditions. Kumar et al. [8] showed that the derivation of the access probability can be simplified by viewing the exponential back-off as a renewal process. In [9] and [10] different models are presented for the derivation of the individual node throughput in arbitrary 802.11 network topologies. Our MAC model goes beyond [7] by considering non-saturated users, lossy channels, non-homogeneous physical data rate, and channel with hidden terminals. Our MAC model modifies and generalizes the IEEE 802.11 models presented in [9] and [11] by Hira et al. Their model takes into account blocking and interference, and computes the throughput of the individual nodes in an IEEE 802.11 network with hidden nodes. Here, we modify and generalize the models to consider multiple paths with common nodes. We also provide a general framework to consider effects of non-saturated flows, node scheduling algorithms and MAC layer losses.

978-1-4244-2324-8/08/$25.00 © 2008 IEEE.
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2008 proceedings.

The rest of the paper is organized as follows: Section II describes the scheduler model, provides the set of equations that describe our MAC and PHY layer models, the routing model and the fixed point approach to the problem. Section III discusses how we use Automatic Differentiation in the current framework for performance metric sensitivity computations. Finally, Section IV provides simulation results, compares performance of the fixed point model with OPNET, and demonstrates the effectiveness of the design methodology.

II. The MAC Layer Modelling

A. The Scheduler Modelling

We consider a network that consists of N nodes and a path set P that is used to forward traffic between the source destination (S-D) pairs in the network. Let P_i be the set of paths that goes through a node i. The scheduler behavior is specified by the scheduler coefficient $k_{i,p}$, which is the average serving rate of path p packets at node i. For simplicity, we assume that all packets have the same length. Let $\lambda_{i,p}$ be the arrival rate and $T_{i,p}$ be the service time of path p packets at node i.

The scheduling rate is a function of MAC and PHY layer packet failure probabilities. In the 802.11 RTS/CTS protocol there are two stages for packet transmission: in the first stage the RTS and CTS are sent between two nodes and in the second stage the data packets and the ACKs are sent. While PHY layer failures can happen in both stages, we assume that MAC layer failures (collisions) only occur during the first stage. 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 a PHY layer transmission failure during stage 2 (data packet and ACK transmission), and $l_{i,j}$ is the probability of a PHY layer transmission failure at stage 1 or 2 from node i to node j.

The total average throughput $\bar{\rho}_i$, of node i, is, $\bar{\rho}_i = \sum_{p\in P_i} k_{i,p}E(T_{i,p})$, which is not greater than 1.

In order to model a FCFS queueing policy, we assume that the scheduler coefficients are:

$$k_{i,p} = \begin{cases} \frac{\lambda_{i,p}}{(1-\beta_{i,p}^m)} & \text{if } \sum_{p'\in P_i} \frac{\lambda_{i,p'}}{(1-\beta_{i,p'}^m)} E(T_{i,p'}) \le 1 \\ \frac{\frac{\lambda_{i,p}}{(1-\beta_{i,p}^m)}}{\sum_{p'\in P_i} \frac{\lambda_{i,p'}}{(1-\beta_{i,p'}^m)} E(T_{i,p'})} & \text{otherwise} \end{cases} \tag{1}$$

where m is the maximum number of packet transmission retries in the IEEE 802.11. If utilization of node i is less than one, we can serve all incoming packets as described in the first line of (1). In the 802.11, if m packet transmission attempts fail the packet will be discarded. However, we assume that the scheduler keeps scheduling the same packet until it is successfully transmitted by the MAC layer. Therefore, to compensate for the transmission failures at the MAC layer, the scheduling rate should be higher than the node arrival rate by the $1/(1-\beta_{i,p}^m)$ factor. On the other hand, if utilization is equal to one, all packets can not be served, but the service rate for each path is still proportional to its compensated arrival rate as given in the second line of (1). In this way, we can model a FCFS scheduling policy. For now we assume that all nodes have infinite buffer capacity, and hence there is no packet drop in a node (this assumption is not critical and can be removed later). The fraction of time $\rho_{i,p}$ that node i is serving path p packets is specified by $\rho_{i,p} = k_{i,p}E(T_{i,p})$.

B. The PHY and MAC Layer Modelling

In this section we provide the set of equations that we use to approximate the wireless link loss parameters and packet service times. This set of equations will be used as an *implicit* function to derive loss parameters and packet average service times from the node throughputs. We consider the 802.11 MAC layer with RTS/CTS mechanism. The unit of time is a time slot, which is equal to the back-off slot of the 802.11 protocol. The following notation is used to represent different nodes and node subsets in the network: C_i is the set of nodes within carrier sense range of node i, C_i^+ are the nodes in the set C_i plus node i, C_i^- is the set of nodes not in C_i^+, and $h_{i,p}$ is the next hop of node i in path p.

Suppose that node i is scheduled to serve a packet on path p. Assuming that the node accesses the channel with a fixed probability $\alpha''_{i,p}$, and there are L back-off stages and the minimum window size is W, we can use the following relation from [7]:

$$\alpha''_{i,p} = \frac{2(1-2\beta_{i,p})}{W(1-2\beta_{i,p}) + \beta_{i,p}(W+1)(1-(2\beta_{i,p})^L)}, \tag{2}$$

We denote the average transmission time of node i during $T_{i,p}$ with $v_{i,p}$. There are two different components in $v_{i,p}$: (i) the average time $d_{i,p}$ spent in the successful transmission and (ii) the average time $f_{i,p}$ spent in failed transmissions. We have [12], [13],

$$v_{i,p} = (1-\beta_{i,p}^m)d_{i,p} + \frac{1-\beta_{i,p}^m}{1-\beta_{i,p}}\beta_{i,p}f_{i,p} \tag{3}$$

where

$$d_{i,p} = T_{\text{RTS}}^{(i,p)} + \text{SIFS} + T_{\text{CTS}}^{(h_{i,p},p)} + \text{SIFS} + T_{\text{P}}^{(i,p)} + \text{SIFS} + T_{\text{ACK}}^{(h_{i,p},p)}, \tag{4}$$

and

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

with

$$\tau_H = T_{\text{RTS}}^{(i,p)} + \text{SIFS} \tag{6}$$

$$\tau_P = T_{\text{RTS}}^{(i,p)} + \text{SIFS} + T_{\text{CTS}}^{(h_{i,p},p)} + \text{SIFS} + T_{\text{P}}^{(i,p)} + \text{SIFS} \tag{7}$$

where $T_{\text{RTS}}^{(i,p)}$, T_{CTS}, $T_{\text{P}}^{(i,p)}$ are the transmission times for the RTS, CTS and data packets on the corresponding connection respectively.

978-1-4244-2324-8/08/$25.00 © 2008 IEEE.
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2008 proceedings.

Consider a node j in the neighborhood of node i. Node j expects to receive a path p packet from node i, if i is scheduled to serve path p, and there is no transmission from node i neighbors that are hidden from j and i accesses the channel. Therefore, the probability that j receives a path p packet from i in a time slot is,

$$\alpha_{i,p,j} = \rho_{i,p}(1 - \theta_{i,j})\alpha''_{i,p} \quad \text{for all } j \in C_i \tag{8}$$

where $\theta_{i,j}$ is the probability of transmissions from node i neighbors that are hidden from node j.

$$\theta_{i,j} = 1 - \prod_{n \in C_i \bigcap C_j^-} (1 - \sum_{p' \in P_n} \rho_{n,p'} \frac{v_{n,p'}}{E(T_{n,p'})}) \tag{9}$$

The probability that a path p transmission from node i is successful is [13]:

$$\begin{aligned} 1 - \beta_{i,p} = & \left(1 - l_{i,h_{i,p}}\right)\left(1 - \theta_{h_{i,p},i}\right) \\ & \times \prod_{j \in C^+_{h_{i,p}} \bigcap C_i} \left(1 - \sum_{p\prime \in P_j} \alpha_{j,p\prime,h_{i,p}}\right) \\ & \times \prod_{j \in C^+_{h_{i,p}} \bigcap C_i^-} \left(1 - \sum_{p\prime \in P_j} \alpha_{j,p\prime,h_{i,p}}\right)^{V_{i,p}} \end{aligned} \tag{10}$$

where $V_{i,p} = T_{\text{RTS}}(i,p) + \text{SIFS}$.

C. Computing the Service Time Components

$T_{i,p}$ is the time to finish a *successful or unsuccessful* transmission of a path p packet at node i, *after* it is scheduled for transmission at node i. The average service time $E(T_{i,p})$ has four components: $d_{i,p}$ is the time spent for successful transmission of path p packets at node i, $u_{i,p}$ is the average time consumed for successful transmission of node i neighbors, $b_{i,p}$ is the average back-off time of node i for path p packets, $c_{i,p}$ is the average time spent in failed transmissions.

$$E(T_{i,p}) = (1 - \beta^m_{i,p})d_{i,p} + u_{i,p} + b_{i,p} + c_{i,p} \tag{11}$$

For the RTS/CTS mode of operation $d_{i,p}$ is given by (4), and $u_{i,p}$ is given by (3). The average back-off time is $b_{i,p} = \sum_{n=0}^{m} W_n\beta^n_{i,p}$, where $W_n = CW_n/2$ is the average back-off time at the n^{th} stage, and CW_n is the contention window at the n^{th} stage. The average number of collisions is $c_{i,p} = \frac{y_{i,p}}{x_{i,p}} w_{i,p}$, where [13]:

$$w_{i,p} = \frac{\sum_{j \in C_i^+} \left(\sum_{p' \in P_j} \alpha''_{j,p'}\beta_{j,p'}\rho_{j,p'}\right)(1 - \theta_{j,i}) f_{j,p'}}{\sum_{j \in C_i^+} \left(\sum_{p' \in P_j} \alpha''_{j,p'}\beta_{j,p'}\rho_{j,p'}\right)(1 - \theta_{j,i})}$$

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

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

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

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

D. The Routing Model and the Fixed Point Implementation

The routing model specifies a fixed set of paths and the fraction of incoming traffic that is sent over each path at the source node. The incoming traffic rates of the nodes are derived from the scheduling and loss rates of their upstream links as follows:

$$\lambda_{h_{i,p},p} = k_{i,p}(1 - \beta^m_{i,p}) \text{ for all } i, p. \tag{12}$$

The fixed point algorithm attempts to find a consistent solution for the sets of equations given by the PHY and MAC layer, the routing and the scheduling models. The fixed point algorithm starts from the source node of each path at each iteration where the arrival rate $\lambda_{i,p}$ are fixed and given. Given the input arrival rates of a node i and its neighbors it uses the PHY, MAC, and the scheduling model equations provided in the previous sections to compute the scheduling rates $k_{i,p}$. Then, we use (12) to compute the next hop incoming traffic rate. Then we repeat the same procedure for the next hop. We continue iterating and updating over all paths in the network until a fixed point is reached.

In order to make the convergence of our fixed point algorithm faster, we initialize the values of our parameters assuming communication is perfect for every connection, the time $T_{i,p}$ consists only of the time taken by successful transmission, plus the back-off time needed for the first trial. Moreover, since we assume perfect channel conditions, every probability of failure is initialized to zero. To ensure convergence of the fixed point iterations and that there is no oscillation between multiple points we update the values of parameters using a weighted sum of current values and previous values [13].

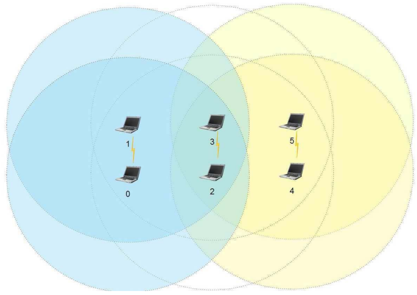
III. Automatic Differentiation for Design

Although the fixed point algorithm can provide the basis for performance analysis of a given network configuration, we need a methodology for network configuration and optimization. We use optimal routing design as an example to illustrate our proposed design methodology. We implement the Dreyfus K-shortest path algorithm [14] for path selection. We use the gradient projection method to find the optimal values for the routing parameters (routing probabilities) to maximize the network throughput.

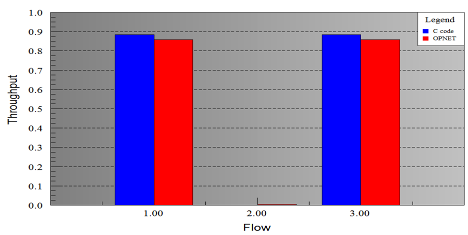
The gradient projection method requires iterative computation of the throughput gradient. The fixed point method provides a computational scheme that, after convergence (i.e. the fixed point), describes the performance metric (i.e. throughput) as an implicit function of the design parameters (i.e. routing parameters). Since we do not have analytic expressions of the performance metric, we use Automatic Differentiation (AD) [15] for the optimization.

Let P_c be the connection c path set, and C the set of active connections in the network. The network throughput T is:

$$T = \left(\sum_{c \in C}\left(\sum_{p \in P_c} \lambda_{last,p}\right)\right) / \left(\sum_{c \in C}\left(\sum_{p \in P_c} \lambda_{first,p}\right)\right) \tag{13}$$

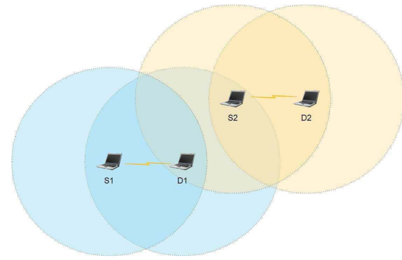


(a) Topology

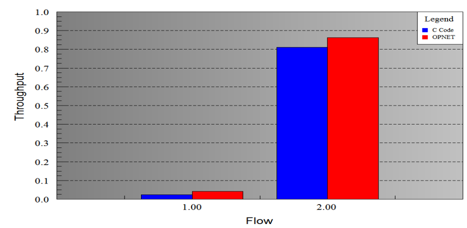


(b) Throughput of flows

Fig. 1. Flow-in-the-Middle Scenario



(a) Topology



(b) Throughput of flows

Fig. 2. Information Asymmetry Scenario

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, we can write our optimization problem in the following way:

$$\begin{aligned} \max\ & T = T(\pi_{1,c_1}, \cdots, \pi_{n_{c_1},c_1}, \cdots, \pi_{n_{c_m},c_m}) \\ \text{s.t.}\ & \sum_{i \in P_c} \pi_{i,c} = 1, \quad \pi_{i,c} \geqslant 0, \quad \forall (i,c) \in P_c \times C \end{aligned} \tag{14}$$

Denoting by $\overline{\nabla}_c$ the connection c average gradient, and by $\beta > 0$ the step size, the route probabilities are iteratively updated as follows:

$$\pi_{i,c_k} = max\ (0, \pi_{i,c_k} + \beta(\frac{\delta T}{\delta \pi_{i,c_k}} - \overline{\nabla}_{c_k})), \forall k \in \{0, \cdots, m\} \tag{15}$$

IV. Results

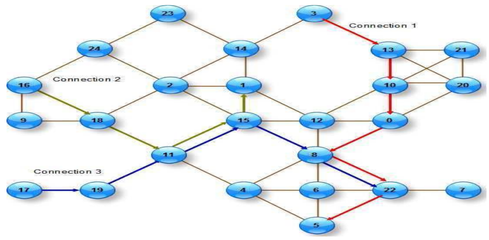
Starvation Models: Multi-hop ad-hoc wireless networks are known to display unfairness in the throughput achieved by the different source-destination pairs in the network. In some cases, the discrepancies are such that they will induce the starvation of connections. We consider two starvation models that are presented in [10] to display unfairness in throughput distribution and test the validity of our model by comparing its throughput results to a discrete event simulator, OPNET 12.0 results.

The first experiment considers the Flow-in-the-Middle (FIM) scenario. The network is composed of 6 nodes and 3 links, as depicted in Fig 1(a). In this scenario, node 0 receives node 2 transmissions, but not node 4's. Node 4 receives node 2 transmissions, but not node 0's. However, node 2 receives both node 0 and node 4 transmissions. This scenario results in starvation of the middle [13]. Results of OPNET simulations and our model analysis is given in Fig 1(b). As can be seen, our fixed point algorithm models this scenario very well.

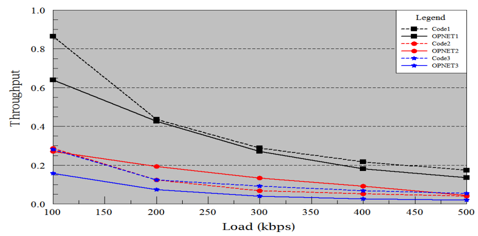
The second starvation case is the Information Asymmetry (IA) scenario, represented in Fig 2(a). In this case, the sources of the two flows are not within hearing range of each other. The main problem in this scenario is that while S2 is aware of the presence of another flow in its neighborhood (it can sense the activity of D1), S1 has no knowledge of the fact that a communication affecting its transmission is happening simultaneously in the vicinity. This means that flow 1 will not be able to fairly compete with flow 2 [10]. This will result in flow 1 having a much lower throughput than flow 2. Fig 2(b) shows the accurate modelling of this unfairness by our fixed point analysis as results match simulation results obtained with OPNET 12.0.

Multihop Connections, Throughput Approximation and Optimization We set up a simple network, presented in Fig 3(a). The blue nodes represent the wireless stations, the brown links the possible wireless connections between the nodes, and the pointed colored links the paths used in the three connections: (i) from node 3 to node 5, (ii) from node 17 to node 22 and (iii) from node 16 to node 1. Our routing algorithm finds the shortest paths between the source and destination nodes having nodes involved in different connections. Using these paths we then employ our set of fixed point equations to compute the throughput of these connections according to the desired load. As can be seen in Fig 3(b) the fixed point model

978-1-4244-2324-8/08/$25.00 © 2008 IEEE.
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2008 proceedings.



(a) Topology



(b) Opnet and fixed point model throughput curves

Fig. 3. Model Evaluation

results are close to the OPNET results.

Next we use the fixed point model with AD to enhance the routing performance. A fixed set of paths are given and we tune the probabilities (portions) of sending traffic over the paths to maximize the throughput. We consider again the network topology shown in Fig 3(a) with the same 3 source-destination pairs. We consider three alternative routings: (1) shortest path only, (2) all given paths with equal probability, and (3) using AD and gradient projection method to find the optimal probabilities. Fig. VI shows the network throughput v.s. the connection path number. Optimization-based algorithm performance improves as the path count increases and it clearly outperforms other policies.

V. Conclusions

We introduced a numerical models for design and analysis of multi-hop wireless networks. Routing, scheduling, PHY and MAC layers are represented with a set of interdependent equations in the model. The MAC layer model considers arbitrary multi-hop wireless network based on the 802.11 model. A fixed point iteration is proposed to find a consistent solution for all equations and to form an implicit model for the network performance. For design and optimization, we use Automatic Differentiation on top of the fixed point model to numerically compute the gradient of the performance metric (throughput) with respect to the design parameters (routing probabilities). Then, gradient projection method is used to compute routing probabilities that maximize the throughput.

VI. Acknowledgements

This work is prepared through collaborative participation in the Communications and Networks Consortium sponsored by the U.S. Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD19-01-2-0011. Research was also supported by NASA Marshall Space Flight Center under award no. NCC8-235.

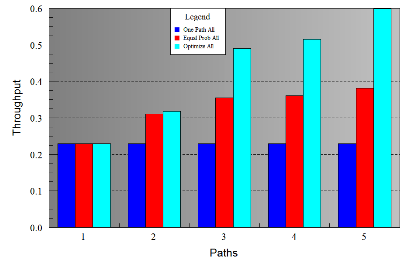


Fig. 4. Throughput v.s. path count for different routings

References

[1] F. P. Kelly, "Loss networks," *The Annals of Applied Probability*, vol. 1, no. 3, pp. 319–378, Aug. 1991.

[2] ——, "Blocking probabilities in large circuit switched networks," *Advances in Applied Probability*, vol. 18, no. 2, pp. 473–505, Jun. 1986.

[3] M. Liu and J. S. Baras, "Fixed point approximation for multirate multihop loss networks with adaptive routing," *IEEE/ACM Trans. Networking*, vol. 12, no. 2, pp. 361–374, Apr. 2004.

[4] D. Mitra, J. A. Morrison, and K. G. Ramakrishnan, "ATM network design and optimization: A multirate loss network framework," *IEEE/ACM Trans. Netw.*, vol. 4, no. 4, pp. 531–543, Aug. 1996.

[5] S. Chung, A. Kashper, and K. W. Ross, "Computing approximate blocking probabilities for large loss networks with state-dependent routing," *IEEE/ACM Trans. Netw.*, vol. 1, no. 1, pp. 105–115, Feb. 1993.

[6] A. G. Greenberg and R. Srikant, "Computational techniques for accurate performance evaluation of multirate, multihop communication networks," *IEEE J. Sel. Areas Commun.*, vol. 5, no. 2, pp. 266–277, Feb. 1997.

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

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

[9] K. Medepalli and F. A. Tobagi, "Towards performance modeling of IEEE 802.11 based wireless networks: A unified framework and its applications," in *Proceedings of the IEEE Conference on Computer Communications, (INFOCOM 2006)*, Barcelona, Spain, Apr. 2006.

[10] M. Garetto, T. Salonidis, E.W. Knightly, "Modeling Per-flow Throughput and Capturing Starvation in CSMA Multi-hop Wireless Networks," *Proceedings of the IEEE Conference on Computer Communications, (INFOCOM 2006)*, Barcelona, Spain, Apr. 2006.

[11] M. M. Hira, F. A. Tobagi, and K. Medepalli, "Throughput analysis of a path in an IEEE 802.11 multihop wireless network," in *Proceedings of the IEEE Wireless Communications and Networking Conference, (WCNC 2007)*, Hong Kong, China, Mar. 2007.

[12] J. S. Baras, V. Tabatabaee, G. Papageorgiou, and N. Rentz, "Modelling and optimaztion for multi-hop wireless networks using fixed point and automatic differentiation," in *Proceedings of the* 6^{th} *Intl. Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks, (WiOpt 2008)*, Berlin, Germany, Mar. 2008.

[13] ——, "Performance metric sensitivity computation for optimization and trade-off analysis in wireless networks," in *http://hdl.handle.net/1903/7547*, ISR, Tech.Rep.ISR 2008-8, Feb. 2008.

[14] S. E. Dreyfus, "An appraisal of some shortest path algorithms," *Operations Research*, vol. 17, no. 3, pp. 395–412, May 1969.

[15] M. Bucker, G. Corliss, P. Hovland, and U. N. amd Boyana Norris., Eds., *Automatic Differentiation: Applications, Theory and Implementations*, ser. Lecture Notes in Computational Science and Engineering. Berlin, Germany: Springer, 2006, vol. 50.