Metamodeling of Cross-layer Cooperative Scheduling in Wireless Local Area Networks

Jie Hui and Michael Devetsikiotis
Department of Electrical and Computer Engineering
North Carolina State University
Raleigh, NC 27695
Email: jhui@ncsu.edu, mdevets@ncsu.edu

Abstract

Multihop, ad-hoc Wireless Local Area Networks (WLAN) based on 802.11 are expected to serve as important segments of next generation wireless networks. In the network layer of intermediate nodes of such networks, a priority scheduler arbitrates the transmission between forwarding traffic and the node's own locally produced traffic. Meanwhile, a new QoS-aware Enhanced Distributed Channel Access (EDCA) of 802.11e provides differentiated access at the MAC layer. Therefore, how to control priority forwarding in a *cross-layer* fashion, namely, by combining scheduling at the network layer with EDCA differentiated access at the MAC layer in order to achieve the best channel utilization is an important design issue. Due to the numerous control variables potentially involved from both layers, analytical modeling is not feasible. Instead of using a simulation model to predict some points of the utilization response surface, we propose to utilize a *metamodel* (used more frequently in other disciplines such as operations research) in order to approximate the complete response surface by fitting regression analysis over a large amount of simulation results. The resulting metamodel allows us to establish and quantify the cross-layer effects through specific interaction items and can serve for the purpose of cross-layer optimization.

I. Introduction

The issue of *cooperation* has received a lot of attention in the context of mobile multihop ad-hoc wireless LAN networks. In an ad-hoc network, participating nodes may be self-interested - they may selfishly turn down the forwarding traffic flow for the other nodes and only transmit its own generated traffic for the reason of saving energy etc. But if every node performs in this manner, no traffic can traverse multiple hops, and network throughput will degrade unacceptably. Therefore, a good utilization of the scarce wireless resources in the whole ad-hoc network depends on the cooperations among participating parties. How to stimulate the cooperation, thus, is a crucial issue in non-cooperative mobile ad-hoc networks.

The existing cooperation stimulation schemes are mostly designed in the *network layer* either by pricing-based or reputation-based approaches. The core concept of pricing-based schemes is to reward the forwarding or relaying behavior and punish the self transmission either in virtual money [1], [2] or other type of credit [3]. In reputation-based schemes [4], [5], [6], a reputation system is maintained by neighborhood monitoring, thus a misbehaving node with bad reputation can be detected and avoided.

Although these cooperation schemes are implemented in wireless LANs, they do not consider the inherent property of wireless channel, i.e., they do not consider medium access control (MAC) effects. Some simulation evaluations have been done but typically in a generic network infrastructure without a real MAC at the bottom.

However, in packet radio networks, especially in mobile ad-hoc networks, the medium access protocol mainly determines the sharing pattern of the radio channel. Hence, different than in wired networks, the MAC cannot be omitted from studies of cooperation in such wireless networks.

Only few reports [7], [8] have appeared studying the misbehavior and cooperation problem solely at *MAC layer* in wireless LAN. These articles explore the binary backoff function at 802.11 Distributed Coordination Function (DCF) MAC and assume a variable contention window (CW) size. However, only to prevent the misbehavior from the MAC point of view by changing the backoff parameters is limited without simultaneous control at the network layer.

Therefore, we argue that cooperation in mobile ad-hoc WLAN is a fundamentally *cross-layer* issue. Neither a purely network layer cooperation nor a solely MAC layer cooperation can achieve the best channel utilization, therefore considering both layers is a must. The promise of cross-layer cooperation enforcement is also forecast in [9] and the same author describes a cross-layer framework in [10].

The objective of cross-layer design in mobile ad-hoc networks is to optimize and exploit the cross-layer interdependencies in order to enhance the performance of the network as a whole. Although many attempts at cross-layer designs, such as [11], [12], [13], [14], [15], [16] and [17] etc., can be found in the literature, there are very few quantitative measurements of cross-layer effects. The reason is that such cross-layer effects are very difficult to capture analytically as the combined cross-layer performance function of the system is intractable.

Therefore we are led to investigate an alternative method - namely, a metamodeling technique [18], [19], [20], in order to find an approximate mathematical function of system performance in terms of the cross-layer design parameters and subsequently quantify the cross-layer effects through the evaluation of interaction terms in the model.

Metamodeling, described as time as a 'model of a model' [21], has been used by the simulation community to study the behavior of computer simulations for over thirty years and applied to many fields including manufacturing, queueing models [22], and computer networks [23], [20]. However, to the best of our knowledge, we are unaware of any application to performance analysis in mobile ad-hoc network.

Furthermore, most of the work in the MAC layer only considers the DCF function in which there is only one access entity per node; however, multiple access entities can be supported in the newly QoS-enhanced 802.11e EDCA. Therefore, how to stimulate cooperation in network layer on top of EDCA MAC layer presents an unaddressed question.

Motivated by these needs, in this paper, we study the cross-layer cooperation consisting of a network layer priority scheduler extended from [1] and a MAC layer EDCA priority access scheduler in a mobile ad-hoc network, by applying a metamodeling technique. Our contributions are three fold: first, we advocate the multidisciplinary use of metamodeling in cross-layer design; second, we provide a metamodel of system throughput in functions of cross-layer cooperation parameters both in network layer and MAC layer; third, we quantify the cross-layer effects between MAC and network layer and bring additional insights into the understanding of cross-layer design.

The remainder of this paper is organized as follows: First, we summarize related work about cooperation in mobile ad-hoc networks in Section I. Second, our proposed cross-layer cooperation network model is presented in Section III in details with a network layer priority scheduler and a MAC layer EDCA priority access scheduler. Then the design of simulation experiment and the final fitted metamodel of the system throughput is described in Section IV. Lastly, we present our conclusions in Section V.

II. Related Work

A. Network Layer Cooperation

A virtual currency (or "nuglet") counter is proposed in [1] to pay for each packet locally generated, and also to be earned by forwarding packets on behalf of other nodes. Only if the nuglet counter is positive, can the node send its own packet. Upon forwarding a packet, the nuglet counter increases by one, while it decreases by the number of hops for transmitting a locally generated packet. The major limitation of this scheme is the unfair treatment of the edge nodes who can not pay for their own transmission because of seldom forwarding requests, in addition to the need for a temper-proof hardware module to protect the nuglet counter.

In contract to using a universal utility metric, [2] proposes a layered scheme consisting of a policed best-effort service and a incentive-based priority forwarding: nodes get compensated for forwarding priority packets and nodes are unaffected if they do not forward packets in a priority fashion.

"Sprite", a centralized credit system in [3], determines charge and credit from a game-theoretic perspective and motivates each node to report its actions honestly. There is no need for temper-proof hardware in this scheme, however, some out-of-band mechanism is required for communication of the credits.

The first reputation based scheme is introduced in [4], in which a watchdog identifies misbehaving nodes by performing a neighborhood monitoring and a reputation system keeps track of reputations of each node.

Another reputation-based scheme called "CONFIDENT" is introduced in [5]. In CONFIDENT, a node monitor the routing and forwarding behaviors of its neighbors and take reputation record and trust records, then send alarms to isolate bad nodes upon detecting misbehavior. However, it may degrade the network utilization by introducing significant reputation propagation overhead and by overloading the well behaving nodes.

A reliability index-based approach [6] takes into account not only the presence of possible selfish/malicious nodes but also situations like congestion and wireless lossy links.

B. MAC Layer Cooperation

MAC greediness, reflected in a smaller backoff interval, is detected by receivers and corrected by enforcing a bigger value in [7]. But it requires modification of the standard and also assumes nonrealistic traffic always in uplink.

A game theoretic scheme for CSMA/CA schemes is presented in [8]. It shows how a Pareto-optimal point is achieved in a dynamic game by adaptively changing the contention window size and misbehavior being penalized by jamming.

III. CROSS-LAYER COOPERATION NETWORK MODEL

For simplicity of explanation and to illustrate our approach, we pick a small example ad-hoc WLAN with three stations as our network model as shown in Fig. 1. The insight of using metamodeling to study the cross-layer cooperation can still be applied to any bigger and more complicated networks. In this ad-hoc WLAN, wireless station (WS) 2 is located in between WS1 and WS3, and can talk to both of them. But WS1 and WS3 can not reach each other. Two traffic flows compete for the resources in this network: flow one is from WS1 to WS3 which has to be relayed at WS2, flow two is from WS2 to WS3 directly.

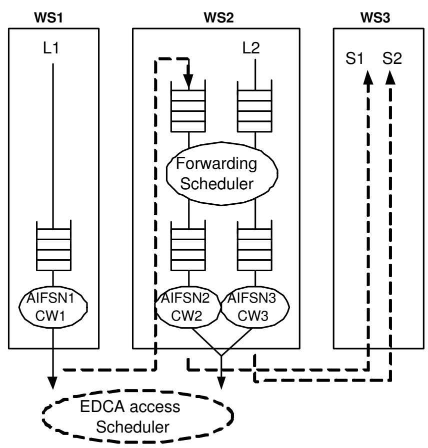

Fig. 1. Example small ad-hoc WLAN network.

A. Application Layer: Traffic Profile

We assume an exponentially distributed traffic inter-arrival profile in this ad-hoc network. For flow one, we define $L1$ as the exponentially distributed arrival rate of traffic in WS1. For flow two, we define $L2$ as the exponential rate of traffic generation in WS2. In our network, we assign $L1 = L2 = load$. Note that the load is normalized in terms of the physical layer transmission rate of 802.11.

B. Network Layer: cooperation forwarding scheduler

Two flows coexist at WS2. At the network layer, WS2 uses a forwarding scheduler to determine the priorities between forwarding traffic and own traffic. Since the station may be selfish and try to maximize its own throughput, cooperation needs to be stimulated and selfishness needs to be punished by setting the appropriate forwarding rule.

There are some scheduler schemes designed in the literature [1], [2], [3]. In this paper, we design our own priority forwarding scheduler by extending the scheme in [1] with a generalized *award* and *punishment* and an upper threshold ($UpThrshd$) and a lower threshold ($LoThrshd$) over the counter.

TABLE I

NETWORK LAYER COOPERATION SCHEDULER SCHEME.

Virtual money counter N2
WS2 maintains a counter N2. Initially, N2=K
Scheduling Rule
IF $N2 > UpThrshd$,
WS2 only sends its own traffic and does not forward traffic;
ELSEIF $LoThrshd < N2 < UpThrshd$,
WS2 sends its own traffic and also forwards traffic;
ELSEIF $N2 < LoThrshd$,
WS2 only forwards traffic for WS1, and not send its own.
ENDIF
Update Rule of N2
IF the transmitted packet belongs to forwarding traffic,
$N2 = N2 + award$.
ELSEIF the transmitted packet belongs to its own traffic,
$N2 = N2 - punish$.
ENDIF

Here, K is a constant, and $UpThrshd$, $LoThrshd$, $award$ and $punish$ are four control variables the priority forwarding scheduler used.

The advantages of our scheduler are the flexible increasing slope (award) and the decreasing slope (punish) of the counter and also the flexible resources arbitration between forwarding and self traffic by adjusting the two thresholds of the counter.

C. MAC Layer: IEEE 802.11e EDCA scheduler

Before introducing the effects of EDCA, we have to explain briefly the basis of EDCA - DCF MAC in legacy 802.11.

A legacy DCF wireless station performs CSMA/CA with the following BEB procedures [24] to access the wireless medium (Fig. 2):

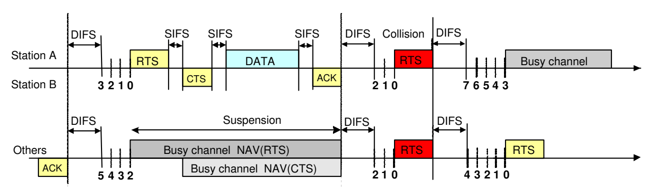

Fig. 2. DCF Access Procedure.

Defer: A station with a pending packet has to wait for the channel to be idle for the duration of a Different Inter Frame Space (DIFS) before the transmission in order to give priority access to Polling or Control Messages. If the channel is sensed busy during this period, the station has to wait for another idle DIFS after the channel is idle again, then performs a random backoff. **Backoff:** Then, the station has to wait for an additional random backoff time, which is uniformly distributed between 0 and CW_{min} slots. If the channel is sensed to be busy during this period, the station suspends backoff until the channel is idle for DIFS again. **Handshaking:** If the packet size is bigger than a threshold, a two-way handshaking procedure is performed to further reduce the DATA collision probability, including a RTS and a CTS packet. **Data Transmission:** If the above procedures are successful, the DATA packet will be transmitted. **Confirmation:** Then, the station awaits an acknowledgement from the destination for confirmation. **Collision and Retransmission:** If more than one station begin their transmissions at the same time, collision happens. The collided station will defer, backoff and retransmit with a new contention window size ($CW_{new=}CW_{old} * 2 + 1$) until CW_{max} is reached, then stays unchanged at CW_{max}. If the retransmission attempts reach a retry limit, the packet will be discarded.

To provide differentiated channel access, EDCA [25], supports up to four access categories (AC) in one QoS station for packets belonging to eight user priorities (UPs) or frame types. AC values of 0, 1, 2 and 3 represent best effort, background, video and voice AC respectively. The mapping between UP or frame type and AC can be found in the draft [25].

Comparing to the equal access of DCF contentions by using the same $DIFS$, CW_{min} and CW_{max}, EDCA offers *differentiated* access through EDCA parameter set $AIFS[AC]$, $CW_{min}[AC]$, $CW_{max}[AC]$ and $TXOPlimit[AC]$ for a corresponding AC (AC=$0,1,2,3$). $AIFS[AC]$ is determined by $AIFS[AC] = SIFS + AIFSN[AC] * T_{slot}$, where $AIFSN[AC]$ is an integer indicating the number of slots after a SIFS duration a station should defer before either invoking a backoff or starting a transmission. Transmission opportunity (TXOP) is a new scheme to improve the efficiency of the protocol. A backoff entity can transmit multiple packets within one TXOP, of which the maximum length is $TXOPlimit[AC]$.

Therefore, we can implement an EDCA priority access scheduler at MAC layer to arbitrate the radio channel resource to multiple access entities by manipulating their EDCA parameters, include AIFSN, CWmin, CWmax and TXOP. In our experimental network, there are three access entities sharing the wireless medium. Access entity one carries traffic $L1$ in WS1; access entity the forwarding traffic for WS1 in WS2; and access entity three transmits WS2 own generated traffic $L2$ in WS2 itself. Our EDCA scheduler is represented in the bottom part of Fig. 1.

In our scheduler, we assign the same TXOP for all access entities and one TXOP only accommodates one packet frame. Here, because WS1 and WS2 can pick different values for their EDCA parameters even for the same AC. Hence, disregarding the AC of the three access entities, we denote the EDCA parameters for them to be $AIFSN_1$ and CW_1, $AIFSN_2$ and CW_2, $AIFSN_3$ and CW_3[1], for access entity one, two and three respectively.

IV. METAMODEL

Viewing this system as a "black box", the input variables to the system come from three layers, and the output responses that we are interested in are the throughput performance values:

TABLE II
METAMODEL INPUT AND OUTPUT

Input
Application layer: traffic load L_1 and L_2;
Network layer: forwarding scheduler parameters $UpThrshd$, $LwThrshd$, $award$ and $punish$;
MAC layer:
EDCA parameters $AIFSN_1$, $AIFSN_2$, $AIFSN_3$, CW_1, CW_2, and CW_3.
Output
throughput of WS1 (S1) and throughput of WS2 (S2)

We denote the input by a vector X and the output by Y. The objective of our study is to find out the model function of $\mathbf{Y} = \mathbf{f}(\mathbf{X})$. An analytical model is not feasible due to the large dimensionality of the design space.

This leads us to construct a *metamodel*, or model of the simulation model. A *simulation model* is used to generate the response surface over the entire design space by emulating the behavior of the real system, because of the great difficulty getting data directly from the real system. Then, the *metamodel* is a fitted mathematical model $\hat{\mathbf{Y}} = \mathbf{g}(\mathbf{X})$ of the simulation model by performing regression analysis over simulation results from multiple runs $(X_1, X_2, ..., X_n)$. A good metamodel should have small approximation and random error $\epsilon = Y - \hat{Y}$.

Metamodeling techniques involve: (1) Experiment design for generating data; (2) Choosing a model to represent the data and then fitting the model to the observed data; (3) and Evaluating the fitted model. We perform these steps in the following subsections.

[1]We let CW=CWmin and $CWmax = (CWmin - 1)^5 - 1$

A. ***Experiment Design***

The experiment design is to decide the simulation configurations before the runs in order to obtain the desired information. Our experiment design uses a 2^k factorial design approach, and the configuration of each input factor is shown in table III.

TABLE III
EXPERIMENT DESIGN

Factors	**Levels of Variation**	**Level Values**
CW_1, CW_2, CW_3	2	8,32
$AIFS_1, AIFS_2, AIFS_3$	2	0,2
punish,award	2	0,1
LoThrshd	2	0,20
UpThrshd	2	80,100
$L_1 = L_2$=Load	2	0.2,0.5

Out simulation model is built in Arena [26], and can be divided into the following main parts: traffic generator, network layer forwarding scheduler and EDCA access scheduler. Although we are not able to validate the simulation model with respect to the real system, we can achieve partial verification since the EDCA access scheduler is already verified in [27], [28] with respect to an analytical model.

Besides the controllable input factors as shown in Table III, the rest of the 802.11b 2Mbps DSSS MAC/PHY parameters in the simulation are shown in Table IV.

TABLE IV
802.11 MAC/PHY SIMULATION PARAMETERS

Parameter	**Value**	**Parameter**	**Value**
RTS	0.352 ms	SIFS	0.01 ms
CTS	0.304 ms	PHY/MAC header	0.328 ms
ACK	0.304 ms	Tslot	0.02 ms

Our simulation model corresponds to a terminating simulation. We run each simulation replication for 2 hours and run 10 replications for each of the input combination.

B. ***Fitting the Metamodel***

The type of metamodel can be response surface, neural networks, induction learning and Kringing, etc. Here we choose to use response surface model due to its reasonable number of factors and the well-established theory and techniques of response surface methodology [29].

The most widely used response surface approximating functions are low-order polynomials. We pick a first-order polynomial function with interactions because we not only want to study the main effect of each factor but also their interactions.

$$S_1 = \beta_0 + \sum_{i=1}^{k} \beta_i X_i + \sum_{i=1}^{k} \sum_{j=1, i<j}^{k} \beta_{ij} X_i X_j. \tag{1}$$

Here, k is the number of the factors, which is 11 in this experiment and $X_1 = Load$, $X_2 = award$, $X_3 = punish$, $X_4 = LoThrshd$, $X_5 = UpThrshd$, $X_6 = CW_1$, $X_7 = CW_2$, $X_8 = CW_3$, $X_9 = AIFSN_1$, $X_{10} = AIFSN_2$, and $X_{11} = AIFSN_3$.

We use least square regression analysis[2] over the simulation data in order to determine the coefficients of the polynomials. We run SAS GLM [30] program over the data we collect from simulation, and obtain the ANOVA statistics for the model. The high value (0.926761) of R^2, a goodness of fit index, indicates that the model exhibits a very high degree of explanatory power in characterizing the throughput performance.

[2]supported by GLM in SAS

We call this model as full-model, since it includes all input factors. But not all interactions in this model are significant, in other words, can be omitted from the model. We judge that the factors with t-test values larger than 0.05 are statistically insignificant and then delete them from the model. The new model without nonsignificant interactions is called reduced-model.

We re-fit the regression model for this reduced-model by SAS GLM. The R^2 of the reduced-model is still high enough (0.925798), which means it can still explain the data well. Also, the t-test values for each polynomial terms in the new model is statistically significant.

Therefore, after inserting these fitted coefficients of the reduced model βs into the equation 1 [3], our final metamodel of throughput S_1 is:

$$
\begin{aligned}
&S_1 = -.4670 - .0024 * load + 0.1946 * award \\
&+0.5214 * punish - .0014 * LoThrshd \\
&+0.0062 * UpThrshd - .0017 * CW_1 - .0006 * CW_2 \\
&+0.0010 * CW_3 + -.0043 * AIFS_1 \\
&-.0027 * AIFS_2 + 0.0053 * AIFS_3 \\
&-.1470 * load * award + 0.2421 * load * punish \\
&+0.0045 * load * CW_1 + 0.0022 * load * CW_2 \\
&-.0055 * load * CW_3 + 0.0181 * load * AIFS_1 \\
&+0.0126 * load * AIFS_2 - .0278 * load * AIFS_3 \\
&+0.0103 * award * punish + 0.0017 * award * LoThrshd \\
&-.0025 * award * UpThrshd + 0.0003 * award * CW_2 \\
&-.0002 * award * CW_3 - .0047 * punish * UpThrshd \\
&+0.0007 * punish * CW_1 + 0.0003 * punish * CW_2 \\
&-.0008 * punish * CW_3 - .0047 * punish * AIFS_3 \\
&-.000008 * CW_1 * CW_2 + 0.00002 * CW_1 * CW_3 \\
&+0.0001 * CW_1 * AIFS_3
\end{aligned}
$$

C. *Evaluation of the metamodel: Cross-layer effects*

From the existence of interactions consisting of two factors from different layers (Table V), we come to the conclusion that these two factors are cross-layer correlated. For example, $award$ from the network layer forwarding scheduler has different effects on S_1 for different value of $CW2$ from MAC layer. When keeping all other factors unchanged, a unit increase of $award$ will increase S_1 by $(0.1946 + 0.0103punish + 0.0017LoThrshd - 0.0025UpThrshd +$ **0.0003*8** $- 0.0002CW3)$ for $CW2 = 8$ and by $(0.1946 + 0.0103punish + 0.0017LoThrshd - 0.0025UpThrshd +$ **0.0003*32** $- 0.0002CW3)$ for $CW2 = 32$.

TABLE V

CROSS-LAYER INTERACTIONS

Application & Network Layer Interactions
$load * award$, $load * punish$
Application & MAC Layer Interactions
$load * CW_1$, $load * CW_2$, $load * CW_3$,
$load * AIFS_1$, $load * AIFS_2$, $load * AIFS_3$
Network & MAC Layer Interactions
$award * CW_2$, $award * CW_3$, $punish * CW_1$
$punish * CW_2$, $punish * CW_3$, $punish * AIFS_3$

[3]The coefficients for the nonsignificant terms are zero.

Therefore, we are able to quantify the cross-layer effects using a metamodeling technique. Taking into account all interactions, a cross-layer optimization is necessary in order to achieve the optimal throughput of $S1$, and our metamodel can serve well towards this goal. Although we only discuss the metamodeling of S_1 in this paper, the same procedure can be applied to S_2 and the total throughput, and the same conclusion about cross-layer effects will apply.

V. Conclusion

Cooperation needs to be enforced in ad-hoc networks in order to combat the selfish or greedy behavior of certain users. The cooperative scheduling scheme we proposed in this paper considers not only the network layer forwarding treatments but also the newly QoS-enhanced wireless medium sharing patterns, therefore, leads to a cross-layer design. Due to the large dimensionality of the control variables involved with the cross-layer scheduler, an analytical performance model is not feasible. Hence, we propose to apply a novel metamodeling methodology to fit a response surface by doing regression analysis over the simulation data.

The fitted metamodel not only provides an approximate close form expression of throughput performance which helps optimization in our next step; but also quantifies the cross-layer effects through statistically significant interactions, which to our best knowledge is the first published in this setting. On the other hand, our study advocates a multidisciplinary application of metamodeling in the study of cross-layer control and optimization, which also brings additional insight in understanding the behavior of ad-hoc networks.

References

[1] L. Buttyan and J. P. Hubaux, "Stimulating cooperation in self-organizing mobile ad noc networks," *ACM/Kluwver Mobile Networks and Applications (MONET)*, October 2003.

[2] B. Raghavan and A. C. Sneoeren, "Priority Forwarding in Ad Hoc Networks with Self-Interested Parties," in *Proc. of Workshop on Economics of Peer-to-Peer Systems*, 1991.

[3] S. Zheng, J. Chen, and Y. R. Yang, "Sprite: A Simple, Cheat-Proof, Credit-Based System for Mobile Ad-hoc Networks," in *Proc. of Infocom*, March 2003.

[4] S. Marti, T. Giuli, K. Lai, and M. Baker, "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks," in *Proc. of Mobihoc*, June 2000.

[5] S. Buchegger and J. Y. L. Bouder, "Performance Analysis of CONFIDENT Protocol Cooperation of Nodes - Fairness in Dynamic Ad-hoc Netowrks," in *Proc. of Mobihoc*, June 2002.

[6] M. Conti, E. Gregori, and G. Maselli, "Towards Reliable Forwarding in Ad Hoc Networks," in *Proc. of PWC 2003*, September 2003.

[7] P. Kyasanur and N. Vaidya, "Detection and Handling of MAC Layer Misbehavior in Wireless Networks," in *Proc. of Dependable Computing and Communications Sysmposium (DCC) at the International Conferecne on Dependable Systems and Networks (DSN)*, June 2003.

[8] M. Cagalj, S. Ganeriwal, and I. Aad, "On Cheating in CSMA/CA Ad Hoc Networks," Tech. Rep., 2004.

[9] M. Conti, E. Gregori, and G. Maselli, "Cooperation Issues in Mobile Ad Hoc Networks," in *Proc. of 24th International Conference on Distributed Computing Systems Workshops - W6: WWAN (ICDCSW'04)*, March 2004.

[10] M. Conti, G. Maselli, G. Turi, and S. Giordano, "Cross-layering in mobile ad hoc network design," *IEEE Computer special issue on ad hoc networks*, Feburary 2004.

[11] Y. Kwon and K. Lat, "Improving transport layer performance by using a novel medium access control protocol with fast collision resolution in wireless lans," in *Proc. of International Workshop on Moddling Analysis and Simulation of Wireless and Mobile Systems*, 2002.

[12] S. Toumpis and A. J. Goldsmith, "Capacity Regions for Wireless Ad Hoc Networks," *IEEE Transactions on Wireless Communications*, no. 4, pp. 746–748, April 2003.

[13] R. L. Cruz and A. V. Santhanam, "Optimal Routing, Link Scheduling and Power Control in Multi-hop Wireless Networks," in *Proc. of Infocom*, March 2003.

[14] S. Pollin, B. Bougard, and G. Lenoir, "Cross-Layer Exploration of Link Adaptation in Wireless LANs with TCP Traffic," in *Proc. of Symposium IEEE Benelux Chapter on Communications and Vehicular Technology*, 2003.

[15] M. Chiang, "To Layer or Not to Layer: Balancing Transport and Physical Layers in Wireless Multihop Networks," in *Proc. of Infocom*, March 2004.

[16] L. Chen, S. H. Low, and J. C. Doyle, "Joint Congestion Control and Media Access Control Design for Ad Hoc Wireless Networks," in *Proc. of Infocom*, March 2005.

[17] X. Lin and N. B. Shroff, "The Impact of Imperfect Scheduling on Cross-Layer Rate Control in Wireless Networks," in *Proc. of Infocom*, March 2005.

[18] W. D. Kelton, "Statistical Analysis of Simulation Output," in *Proc. Winter Simulation Conference*, 1997.

[19] R. R. Barton, "Metamodels for Simulation Input-Output Relations," in *Proc. Winter Simulation Conference*, 1991.

[20] T. G. Bailey, K. Bauer, and A. B. Marsh, "Response Surface Analysis Stochastic Network Performance," in *Proc. Winter Simulation Conference*, 1989.

[21] T. W. Simpson, J. D. Peplinski, P. N. Koch, and J. K. Allen, “Metamodels for Computer-based Engineering Design: Survery and recommnendations,” *Engineering with Computers*, pp. 129–130, 2001.

[22] S. C. Agrawal, *Metamodelig A Study of Approximations in Queueing Models*. The MIT Press, 1985.

[23] W. H. Shaw, N. J. Davis, and R. A. Raines, “The application of metamodeling to interconnection network analysis,” *ORSA Journal on Computing*, 1994.

[24] “IEEE 802.11 part 11: Wireless lan medium access control (MAC) and physical layer (PHY) specifications,” Aug. 1999, IEEE WG.

[25] IEEE 802.11 WG, “Draft Supplement to IEEE Standard 802.11-1999: Medium Access Control (MAC) Enhancements for Quality of Service (QoS),” Nov. 2003, IEEE 802.11e D6.0.

[26] Arena Website, http://www.arenasimulation.com.

[27] J. Hui and M. Devetsikiotis, “Performance Analysis of IEEE 802.11e EDCA by a Unified Model,” in *Proc. Globecom 2004*, Dec 2004.

[28] ——, “A unified model for the performance analysis of ieee 802.11e edca,” *to appear in IEEE Transaction on Communications.*

[29] R. H. Myers and D. C. Montgomery, *Response Surface Methodology*, 1995.

[30] SAS Website, http://www.sas.com.