# Models of Hybrid Wireless Networks with Realistic Workloads

Pieter Kritzinger and Paolo Pileggi

Data Network Architectures Group,
Computer Science Department University of Cape Town,
Private Bag, Rondebosch, 7700, South Africa
Email:{psk,plgpao001@cs.uct.ac.za}

Technical Report CS06-17-01

**Abstract:** More and more the integration of WiFi and WiMaX wireless networks seem to be the favorite metropolitan area networking option for the future. Some authors refer to these as hybrid wireless networks (HWN). Call admission control in HWN, the efficiency of routing protocols and so on, depend on the QoS or performance of the network. In order to predict the performance of these networks one needs to build a prototype or model them. Simulation models are clearly an option, but simulations become complex, are hard to validate and require much processor time when the network becomes large. In this paper we advocate a hierarchy of models build upon an analytic multiclass queueing network model. We show the results of comparing such a network with simulations of the same network and using inter arrival time and packet distributions of measured Internet traffic. While trends are the same between the simulation and analytic model results, the absolute values are not.

**Keywords:** Mesh networks, wireless networks, Weibull distribution, log-normal distribution, performance modeling, multi-class queueing networks, simulation, wireless.

## I. Introduction

It is increasingly evident that an integration of IEEE 802.11 (WiFi), particularly wireless mesh networks with 802.16 (WiMax) will be the access networks of the future. Some authors[1] refer to this as hybrid wireless networks and some such networks are already operational although not all standards have been finalized. Because it is a recent development, there are great number of interesting open research questions in the field such as call admission control (CAC), scheduling, routing, security and so on.

A number of articles have been published on the capacity of multi-hop wireless networks. The most well-known is probably that by Gupta and Kumar[10] which showed that the network capacity scales as $\Theta\sqrt{n}$ with $W$ the channel transmission speed and $n$ the number of nodes. Neither this work nor the follow-on research by Towsley[1] considers the impact of a specific MAC protocol or scheduling schemes. On the other hand, work on call admission control such as that by Niyato and Hossain[12] or routing and scheduling by Shetiya and Sharma [15] use detailed stochastic analyses of a node in isolation and for 802.16 networks only.

Clearly the ideal model of a hybrid wireless network with QoS guarantees will have to take into account the complete network, routing and scheduling and different MAC protocols and so on. We have been unable to find any such models in the literature and decided to begin by developing a multiclass queueing network models of a hybrid wireless network and compare the results with a simulation of the same network. While it is always possible to simulate any system to a chosen degree of detail, we wanted to know whether the analysis of the performance for QoS purposes as required when analysing, for example, (new) routing or scheduling techniques, can be done with Multiclass Queuing Network (MQN) models, possibly using analytical sub-models such as that in [12]at a node or whether we one has to simulate the network to arrive at performance values.

In addition we used measured data and cumulative distribution functions of Internet browsing sessions which were measured by Walters [16] in our model comparisons reported in Section VI below in order to study the impact of realistic distributions on the accuracy of the analalytic models.

## II. Hybrid Wireless Networks

For the purposes of this paper we shall consider the example Hybrid Wireless Network (HMN) illustrated in Figure 1 as representative. It consists of nodes which are,

- either IEEE802.11 standard access points, called *Subscriber Stations* (SSs) (nodes N5 and N6 in the figure), or
- IEEE802.16 standards stations which are called *Base Stations* (BSs) (nodes N1, N2, N3 in the figure).
- Traffic consists of IP packets carrying web browsing traffic.

Subscriber stations communicate with each other or the Base Stations using an IEEE 802.11 standard and BS's communicate using some IEEE 802.16 standard. The importance of the standards in the models is the MAC protocol and the corresponding service discipline and capacity rather than the details of the standard itself. The Internet is modeled (node N4) by a random delay of arbitrary distribution in both the MQN and the simulation models.

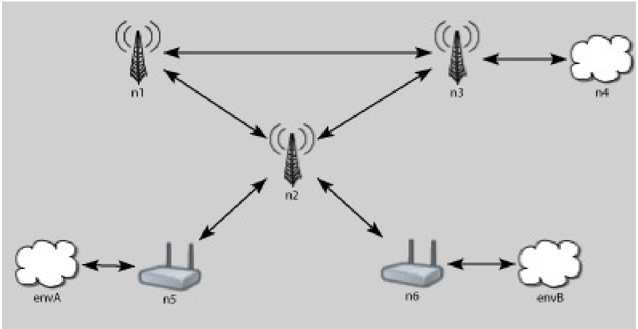


Fig. 1. Example hybrid wireless network

## III. MULTICLASS QUEUEING NETWORKS (MQN)

The analytic models we created of the network in Figure 1 are open multiclass queueing networks (MQN). MQN have their origin in the, now classical, work by Baskett et al[8]. Closed or mixed (open and closed) networks are also allowed if the modeling situation demands it.

In these MQN models customers can have different classes $r = 1, \ldots, R$, thus allowing each of them to have a class-dependent service requirement from a server $i = 1 \ldots, N$ where the network consists of $N$ nodes and $R$ customer classes.

Class $r$ customers arrive at node $j$, from the outside according to a Poisson process with mean arrival rate $\lambda_{jr},\ j = 1, \ldots, N$ and $r = 1, \ldots, R$. Customers may change class in going from one server to the next. That is, a customer of class $r$ completing its service at server $j$ may go to server $i$ and change to class $s$ with probability $p_{jr,is}$. Clearly, if $p_{jr,0}$ is the probability that a customer of class $r$ will leave the network at server $j$, the routing probabilities satisfy:

$$\sum_{i=1}^{N} \sum_{s=1}^{R} p_{jr,is} + p_{jr,0} = 1 \quad j = 1, \ldots, N;\ r = 1, \ldots, R.$$

BCMP or product form queueing networks allow for four different types of servers:

1) *First Come First Served (FCFS)* servers which must have exponentially distributed service times. In this case the service times may not be class dependent.
2) Servers with a *Last-Come First-Served Preemptive Resume (LCFS-P)* rule. The service requirement of each class of customer may depend on the class of the customer.
3) The third type of server is of the *Processor-Sharing (PS)* type and the service requirement of each class of customer may depend on the class of the customer. This basically means that if there are $n_{ji}$ class $i$ customers present at server $j$ and the service requirement of a class $i$ customer is exponential with mean rate $\mu_{ji}$ then the mean service completion rate at server $j$ is

$$\frac{n_{ji}}{n_j} \mu_{ji}, \quad i = 1, \ldots, I.$$

4) A server with an *Infinite Service (IS)* capacity. Each customer, upon arrival, starts its service immediately.

Later work by Gelenbe[9] and others described in Chao[4] added *signals* to the network, apart from the regular customers. A signal may or may not have a “customer” associated with it. When a signal arrives at a server, it causes an event to occur. This event may be the addition (positive signal) or the deletion (negative signal) of one or more customers to the server. It may be even more general. The authors could not for the moment see the application of signals in the models discussed here.

The MQN used in this paper is illustrated in Figure 2. Customers of two different down link classes, shown as solid lines, arrive at SS servers N5 and N6, respectively. These servers are assumed to be of type PS since the assumption is that they use 802.11 or CSMA/CA, and typically the more customers arrive the slower the throughput will be at these servers due to the contention resolution. We assume therefore that all customers in the uplink are trying to transmit messages, which may well be the case since there are normally more than one channel available. Customers retain their class and go to

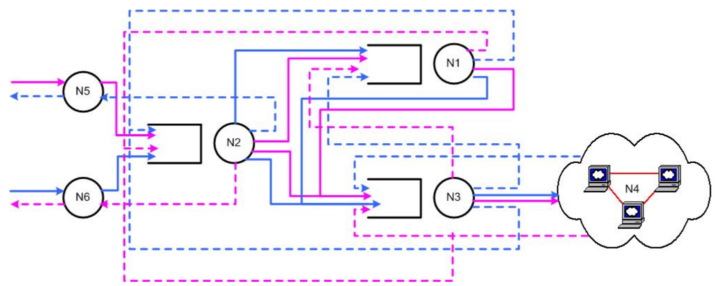


Fig. 2. MQN model of an example mesh network

BS server N2 from which they are routed with probability $p_{2r,js}$ where $j = 1, 3$ and $s =$ uplink$\vee$down-link to servers N1 and N3. All three these servers are assumed to be generalized processor sharing servers as described by Parekh [13] and therefore represented as type PS in the MQN models.

All customers of the uplink classes which complete service at server N1 go to server N3 and all go from there to the Internet, modeled by an IS server N4.

Having received an arbitrarily distributed service time from the IS server, each uplink class changes to its respective downlink class (shown as broken lines) and return via the various servers to their originating SS server as illustrated in Figure 2.

### A. Solving the MQN models

Although the theory of MQNs was well-understood and research is still being done in the field, it was originally not at all obvious how to solve such models in finite time. Originally these models were solved using the so-called *Convolution Algorithm* invented by Jeff Buzen[3] but is was soon clear that, with BCMP networks, there were problems with numerical stability and accuracy of the answers. In 1980 Martin Reiser and Steve Lavenberg[14] published the well-known Mean Value Analysis (MVA) solution technique which was subsequently adopted by one of the authors[11] and his colleagues at the time and turned into a software tool[6] called MicroSNAP for solving mixed multiclass queueing networks. MicroSNAP was used to solve the analytical models described in this report.

### B. Parameterization

As anyone who has ever developed an analytical or simulation model knows, the hard part is finding realistic values for the parameters of the model. One way is to measure the parameters by instrumenting an existing system which somewhat contradicts the purposes of modeling which is normally used for early analysis of planned systems.

In the examples, the traffic flowing in a HWN is IP traffic, the largest proportion of which would be web queries on the uplink and web server responses on the down-link. Measurements of web-traffic are many, typically by Crovella[7], and measurements and analyses were done at the authors' own institution by Walters[16]. The variables of interest to this study are the Web client

1) request Inter-arrival Time (IAT),
2) request size or size of uplink requests, and the
3) response size or downlink requests size.

The random variable distribution functions that Walters discovered fitted the data for the IAT variable best is the *Weibull distribution*, which is given by

$$F(x; k, \alpha) = 1 - e^{-\left(\frac{x-x_0}{\alpha}\right)^k} \tag{1}$$

for $x \geq x_0$ and $F(x; k, \alpha) = 0$ for $x < x_0$, where $k > 0$ is the *shape parameter* and $\alpha > 0$ is the *scale parameter* of the distribution. The mean value of the two-parameter Weibull distribution is given by

$$E[X] = x_0 + \alpha\Gamma\left(\frac{k+1}{k}\right) \tag{2}$$

For the Web client request and response distributions the *log-normal distribution* fitted best. This is the probability distribution of any random variable whose *logarithm* is normally distributed. The log-normal distribution has probability *density* function $f(x; \bar{x}_l, \sigma_l)$ given by

$$f(x; \bar{x}_l, \sigma_l) = \frac{1}{x\sigma_l\sqrt{2\pi}} e^{-(\ln(x-x_0)-\bar{x}_l)^2/2\sigma_l^2} \tag{3}$$

for $x > x_0$, where $\bar{x}_l$ and $\sigma_l$ are the mean and standard deviation of the variable's *natural logarithm*. The mean value is given by

$$E[X] = e^{\bar{x}_l + \sigma_l^2} \tag{4}$$

There is no closed form for the log-normal cumulative distribution function.

## IV. OMNET++ and Simulation

MQN assumes exponential distributions for the arrival process and the FCFS servers. Whereas this may be an acceptable assumption, depending on the purposes of the model, the random variables mentioned seldom exhibit this ideal distribution although most analyses, for example that by by Niyato and Hossain[12], implicitly assume that this is the case. Indeed it was the very purpose of our study to determine the error such a simplification would introduce and for which we developed a simulation model of the network shown in Figure 1.

For the simulation we used the simulation development environment called OMNeT++ (Objective Modular Network Testbed in C++). OMNeT++ provides a component architecture for models. Components (modules) are programmed in C++, then assembled into larger components and models using a high-level language (NED). The environment uses a message passing model where the content of a message is defined by the user.

Modules are connected together via gates (or "ports"), and combined to form compound modules. Connections are created within a single level of module hierarchy: a submodule can be connected with another, or with the containing compound module. Every simulation model is an instance of a compound module type. This level (components and topology) is dealt with in NED files.

The simulator writes *output vector* and *output scalar* files. The capability to record simulation results has to be explicitly programmed into the simple modules by the model builder.

An output vector file contains several output vectors, each being a named series of (time stamp, value) pairs. Output vectors can store things like queue length over time, end-to-end delay of received packets, packet drops or channel throughput – whatever the simple modules in the simulation have been programmed to record. It is possible to configure output vectors and to enable or disable recording individual output vectors, or limit recording to a certain simulation time interval.

## V. Parameter Values

Throughout we used the measured parameter values for IP traffic from the work by Walters[16] as discussed in Section III-B. The values he found were of the same magnitude as those measured by other authors such as Choi[5] and Barford [2]. The following table shows the parameter values Walters and Choi, respectively, found for the Web client parameters used in our models. With no rational argument for or against one or the other set of values, and since they were very much of the same order, we chose the values measured by Walters for our models and list the values by Choi merely in support for those we have chosen.

| **Parameter** | **MEAN ($\bar{x}$)** | | **STD ($\sigma$)** | |
|---|---|---|---|---|
| | Choi | Walters | Choi | Walters |
| Request Size (bytes) | 360 | 418 | 107 | 156 |
| Response Size (bytes) | 7 758 | 5 222 | 126 168 | 15 994 |
| IAT (milliseconds) | 900 | 1 500 | 2 200 | 5 700 |

TABLE I

TABLE OF MEASURED PARAMETER VALUES

Both Choi and Walters found that the web client response and request sizes, respectively, were best approximated by a log-normal distribution, Eq. 3, which is what we therefore used in our final OMNET++ model. The mean and standard deviation used to compute the parameters (cf Eq. 3) are taken from Table I.

Choi reports that the Gamma distribution fits the Web client request IAT best, while Walters determined the best (or better said, the best of the worst) to be the Weibull distribution and the Gamma distribution only third best (Walters [16] page 104). The difference, as to how well the two distributions fit the measured data, are slight and we decided to use the Weibull distribution for the Web client request IAT. We also assumed the number of request arrivals at each of the SSs to be the same in both the MQN and OMNET++ model. When varying the mean IAT in each experiment, we kept the shape parameter (i.e., $k = 0.371$ in Eq. 2 and measured by Walters) the same and computed the corresponding $\alpha$. In all cases we used mean IAT values which would stress the capacity of the network so as the emphasize the error between the analytic and simulation values.

An important parameter value we did not know, and which Walters did not measure, is the Internet response time modeled by node N4 in either Figure 2 or 1. Numerous measurements exist however and, depending on the access network and workload, one may as well choose an arbitrary value.

We arbitrarily chose values of 1Mbps for the mean service times at a 802.11 node and 1.4Mbps at an 802.16 node. Mean request sizes on the up-link were chosen to be 488 bytes and 5222 bytes respectively (see Table I)

Wherever IP requests are routed to two servers, for instance at the AP node N1, they are assumed to go to either node with equal probability. This is an arbitrary assumption and is a parameter of either type of model and can easily be modified if measured values are available.

## VI. Experimental Results

With the types of distributions and the values of the parameters chosen, we were able to experiment with both a MQN and an OMNET++ simulation models. From Figure 1 it should be clear that the node most utilized will be the AP node N2 which is therefore the node where we recorded the queue length results reported below. For the MQN model we computed the response time by ....

### A. Validation

The first scenario we tested was to use exponential distributions or all variables in the models. This was done to validate the analytic versus the simulation models and the results are illustrated as plots `OM: all exponential` in Figures 3 through 4 for the response time of a SS (either node N5 or N6) and Figures 5 through 6 for the queue length at the BS node N2. The analytic results are annotated with the letters MS in the figures, the simulated results with OM. 90% confidence levels are shown only for the plots with the Weibull IAT and log-normal distributions described later. As can be seen from the figures the error is relatively small, about 5% for the response time and almost 0% for the queue length measurements. This hardly a surprise since the simulation is

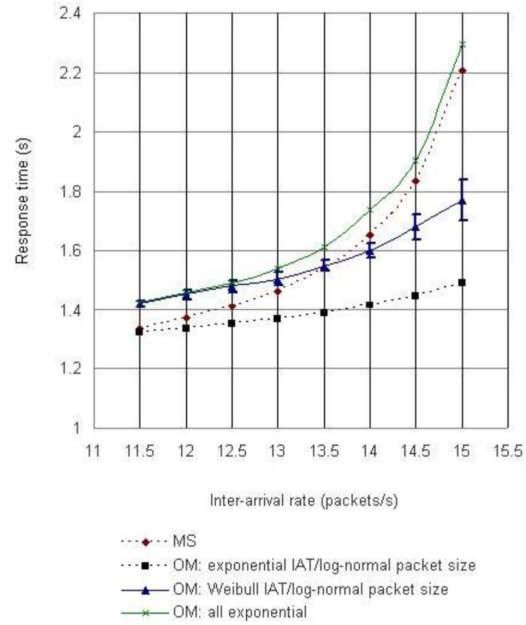


Fig. 3. Network response time

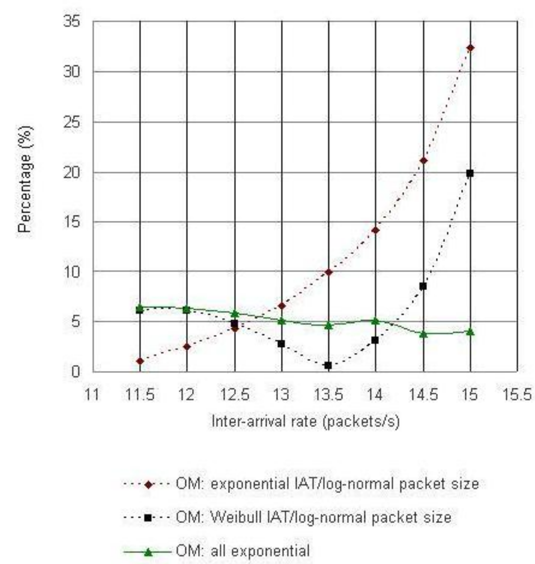


Fig. 4. Error between mean network response time simulated and analytic results

of the analytic model behavior with identical assumptions and vice-versa. The greater error in the response time derives from the difference between the way the analytic tool calculates the so-called *residence time* of a customer in the network and the simulation.

### B. Weibull Inter-arrival Times

In the second scenario we next changed the IAT distribution in the simulation from exponential to Weibull on our way to the last scenario which uses the measured distributions everywhere. We used parameter values suggested by [16]. The results are illustrated as the plots `OM: Weibull IAT/log-normal message size` in the figures already mentioned.

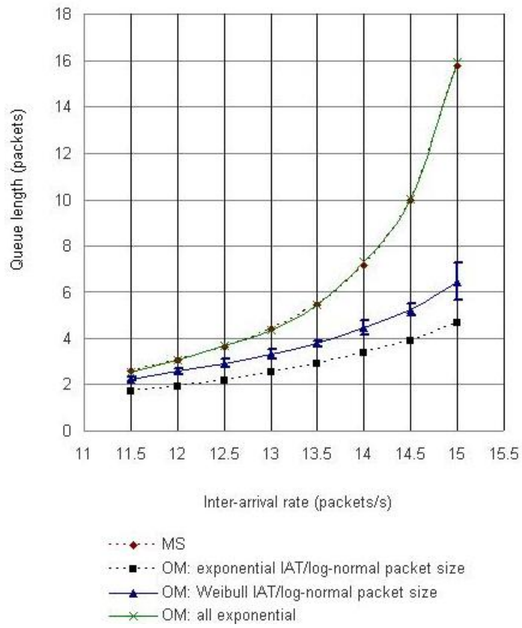


Fig. 5. Mean queue length at BS N2

The simulation model now reports a longer response time than in the first scenario. The results are fairly accurate except at high arrival rates with an average error between the analytic and simulated response time values of 12% increasing from 0% percent to a high of 33% at saturation. The trend in the mean queue length results is the same except that the error is unacceptably high with a mean of 48%. The Weibull distribution is a heavy tail distribution, with a higher probability than the exponential distribution of long IAT. It is to be expected that the queue length will be overstated in the case of the exponential distribution.

### C. Log-normal Service Times

In the last scenario, we changed the simulation model so that *both* the IAT and the message size distributions correspond to the measured behavior. The results are illustrated as the plots `OM: Weibull IAT/log-normal message size` in Figures 3 through 4 for the response time of a SS (either node N5 or N6) and Figures 5 through 6 for the queue length at the BS node N2.

The analytic values lie outside the 90% confidence interval except for very low arrival rates. The error (and that is what is of most interest), in the response times is an acceptable 6% to 20% at network saturation. The error between the analytic model values and those from the simulation, in the queue length at node N2, varies from about 12% to a high 60% at saturation which is of course unacceptable.

In conclusion to this section, the authors do not claim that the parameter values or their distributions used in the models accurately represent any typical Web browsing scenario, since

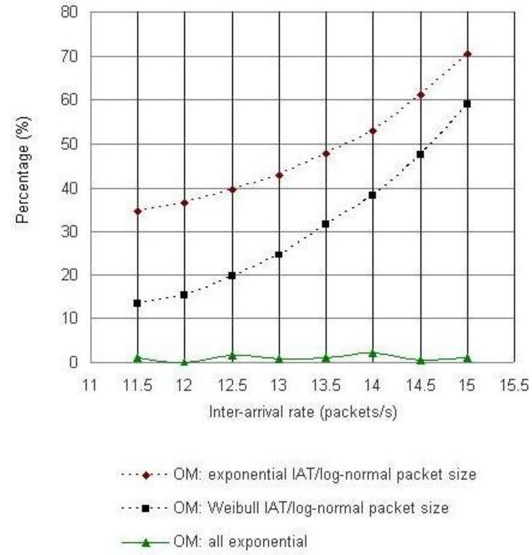


Fig. 6. Error between mean queue lengths at BS N2 for simulated and analytic results

there is no definition of the latter. It is also likely, that due to the different behavior of wireless versus fixed line networks, the characteristics of the variables or their mean values and standard deviations will be different. However, if anything, we wanted to base the parameter values using measured data. We do not believe our assumptions invalidate the basic result of our experiments regarding the accuracy of the MQN models versus that of a simulation of the same network.

## VII. Conclusion

Our original objective was to determine whether analytical MQN, with sub-models of some kind to more accurately model scheduling or routing behavior, would be adequate base models for analyzing hybrid wireless network performance. This we wanted to know since analytic models scale easily and are less prone to errors as is the case with a simulation. We are moreover motivated by the fact that studying the effects of scheduling or admission control based upon QoS parameters need to take the entire network into consideration and cannot be studied in isolation. With a network of many nodes a simulation will become intractable.

In the end, we believe that a MQN model will suffice for the purpose originally given, namely as a base model a hybrid wireless network which takes into account the complete network, provided one is aware that,

- while trends will be the same, which is an important result if comparisons between two hardware or software configurations are all that are needed,
- absolute values will not be the same.

## References

[1] B. Lui, Z. L., and Towsley, D. On the capacity of hybrid wireless networks. In *Trans. InfoCOM* (2003).

[2] BARFORD, P., AND CROVELLA, M. Generating representative web workloads for network and server performance evaluation. In *Performance 1998/ACM SIGMETRICS* (1998), pp. 151–160.

[3] BUZEN, J. Computational algorithms for closed queueing networks with exponential servers. *Communications of the ACM 16*, 9 (September 1973), 527–531.

[4] CHAO X., M. M., AND PINEDO, M. *Queueing Networks: Customers, signals and product from solutions*. John Wiley and Son, 1999.

[5] CHOI, H., AND LIMB, J. A behavioural model of web traffic. In *7th International Conference on Network Protocols, Toronto, Canada* (October 1999).

[6] CROSBY, S., ET AL. Microsnap user manual. Tech. rep., University of Cape Town, Department of Computer Science, August 1990. Version 4.

[7] CROVELLA, M., AND BESTAVROS, A. Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes. In *Proceedings of SIGMETRICS'96: The ACM International Conference on Measurement and Modeling of Computer Systems.* Also, in Performance Evaluation Review, May 1996, 24(1):160-169.

[8] F. BASKETT, K.M. CHANDY, R. M., AND PALACIOS, F. , closed and mixed networks of queues with different classes of customers. *J of the ACM 22* (1975), 248–260.

[9] GELENBE, E. Product-form queueing networks with negative and positive customers. *J of Applied Probability 28*, 565–663 (1991).

[10] GUPTA, P., AND KUMAR, P. Capacity of wireless networks. *IEEE Trans. on Information Theory 46* (2000), 388–404.

[11] KRITZINGER, P., AND VAN WYK, S. Mean Value Analysis: a collection of the results. Tech. Rep. ITR82-14-00, University of Stellenbosch, April 1982.

[12] NIYATO, D., AND HOSSAIN, E. A queueing theoretic and optimization-based model for radio resource management in IEEE 802.16 broadband wireless networks. *IEEE Transactions on Computers 55*, 11 (November 2006), 1473–1488.

[13] PAREKH, A. K., AND GALLAGER, R. G. A generalized processor sharing approach to flow control in integrated services networks: The single-node case. *IEEE/ACM Transactions on Networking 1*, 3 (June 1993), 334–357.

[14] REISER, M., AND LAVENBERG, S. Mean Value Analysis of closed, multichain queueing networks. *J of the ACM 27*, 2 (April 1980), 313–322.

[15] SHETIYA, H., AND SHARMA, V. Algorithms for routing and centralized scheduling to provide qos in ieee 802.16 mesh networks. In *WMuNeP '05: Proceedings of the 1st ACM workshop on Wireless multimedia networking and performance modeling* (New York, NY, USA, 2005), ACM Press, pp. 140–149.

[16] WALTERS, L. A web browsing workload model for simulation. Masters, University of Cape Town, Department of Computer Science, May 2004.