Dynamic resource modeling for heterogeneous wireless networks

Dimitrios Tsamis[1]
Electrical Engineering
Stanford University
Stanford CA 94305
dtsamis@stanford.edu

Tansu Alpcan
Deutsche Telekom Laboratories
Ernst-Reuter-Platz 7
10587 Berlin, Germany
tansu.alpcan@telekom.de

Jatinder Pal Singh
Deutsche Telekom Inc. R&D Lab
5050 El Camino Real
Los Altos, CA 94040
jatinder.singh@telekom.com

Nick Bambos
Electrical Engineering
Stanford University
Stanford, CA 94305
bambos@stanford.edu

***Abstract*— High variability of access resources in heterogenous wireless networks and limited computing power and battery life of mobile computing devices such as smartphones call for novel approaches to satisfy the quality-of-service requirements of emerging wireless services and applications. Towards this end, we first investigate a Markov-based stochastic scheme for modeling and estimation of bandwidth and delay on heterogenous wireless networks. Borrowing clustering techniques from machine learning literature for intelligent state quantization, we demonstrate that the performance of the Markov model is enhanced significantly. We implement a measurement tool *Zeus* on smartphones and collect real-world data on 802.11g, 2.5G, and 3G wireless networks. The accuracy of the developed model is evaluated through simulation studies based on the collected data. Furthermore, a distributed rate-control scheme leveraging the predictions of our model is developed and observed to be much more efficient than a baseline additive-increase multiplicative-decrease scheme.**

I. Introduction

The widespread deployment of high-capacity heterogeneous wireless networks and proliferation of ubiquitous mobile computing devices such as affordable smart phones has led to new opportunities as well as challenges for efficient design, modeling, and optimization in distributed computing systems. These challenges stem from factors such as high variability of bandwidth and delay in wireless networks, limited computing power and battery life of mobile computing devices, and stringent quality of service (QoS) requirements of applications such as voice and video.

In this work we investigate novel approaches for modeling and estimation of wireless network access properties such as bandwidth and delay using Markov models supplemented with intelligent state quantization methods from machine learning literature. Using set-oriented numerical methods, we replace quantities such as available bandwidth (ABW) and round trip time (RTT) with their stochastic counterparts (finite-dimensional Markov chains). We thus capture the evolution of the probability distribution over a finite set of states instead of the individual trajectories of these quantities. The choice of quantization scheme describing the underlying state space is of crucial importance as will be illustrated in Section III. Therefore, we make use of machine learning based advanced clustering techniques to develop a smart quantization scheme. The dynamic bandwidth model developed is demonstrated within the framework of a network assisted distributed rate control scheme [1] with QoS guarantees and compared to existing approaches such as additive increase multiplicative decrease (AIMD) rate control and a baseline bandwidth estimator.

Tasks such as bandwidth estimation or distributed rate control can be realized on behalf of mobile computing devices by a network assistance server (NAS) located at the edge of the network in close proximity to the mobile devices with a low latency communication channel. In evolving next-generation all-IP telecommunication infrastructure such a design paradigm based on *network assisted computing* can be a vital architectural component. NAS supports mobile devices in heavy computational tasks and helps their battery preservation. In general NAS can provide assistance in optimization and network control tasks such as the ones described in this work and in general a range of capabilities such as content caching at network edges. Network assisted computing can be seen as a variant of cloud computing that is specifically designed for wireless networks and mobile devices.

A. Related Work

In [2], a Markov Decision Process (MDP) based flow assignment is proposed for muti-homed access to heterogeneous networks for multimedia traffic. In the present work we offer a substantially better modeling over the arbitrary choice of state space in [2]. Furthermore the present work is applicable to any kind of traffic and addresses practical application to mobile devices and networks. A flow control methodology for heterogeneous networks has been proposed in [3]. The work employs H^∞ optimal rate control for the purpose. The modeling and measurement of access network resources is outside the scope of this work. There has been extensive work on available bandwidth estimation tools (see for example [4]–[6]). Although there tools were originally designed for wired networks, there have efforts for bandwidth measurements in wireless environments [7]–[9]. To the best of our knowledge, ours is the first implementation of a measurement tool for mobile devices. [10] remains a classical work in rate control

[1]Research sponsored by Deutsche Telekom AG. D. Tsamis was with Deutsche Telekom Laboratories while this research was conducted.

but does not take into consideration predictions on the behavior of the channel. [11] builds up on the work as a survey on extensions, new results and directions. [1] introduces the notion of delivery contracts and also a stochastic version of the network utility maximization problem. [12] investigates throughput predictions using machine learning methods, but it specifically refers to TCP flows.

B. Summary of Contributions

The contributions of the present work can be summarized as follows

- Computationally light bandwidth (and RTT) modeling in heterogeneous wireless networks with intelligent quantization using Machine Learning techniques and Markov models.
- Development of a measurement scheme suitable for smartphones and mobile devices lacking clock accuracy.
- Real data collection via experiments on Wi-Fi, 2.5G, and 3G wireless networks and their analysis.
- Comparative evaluation of bandwidth estimation schemes utilizing the collected data.
- Demonstration of the value of bandwidth estimation through simulation of a finite horizon distributed rate optimization scheme with QoS guarantees.

C. Organization

The rest of this paper is organized as follows: we discuss the collection of real world data on heterogeneous wireless networks in Section II. We discuss dynamic bandwidth modeling approaches and their performance in Section III. In Section IV, we discuss network assisted rate control that utilizes the bandwidth modeling and estimation approaches. We conclude the work in Section V.

II. Data Collection and Experiments

We measure and analyze available bandwidth and delay properties of heterogeneous wireless networks including WiFi (802.11g), 2.5G (GPRS), and 3G (UMTS) networks using the mobile services and security testbed (MoSST) located in Deutsche Telekom Laboratories.

A. Testbed and Experiment Setup

The testbed consists of multiple Nokia N80 smartphones which run Symbian S60 operating system and have connectivity capabilities for 802.11b/g, UMTS, EDGE and GPRS networks as well as a WiFi access point and a network assistance server. Both the smartphones and servers are programmed using Python language (PyS60 version on mobiles) which allows for easy and rapid development. The WiFi access point is directly connected to the server through Ethernet. In the experiments all phones connect to the same 802.11g access point and NAS. A block diagram of the testbed is shown in Figure 1.

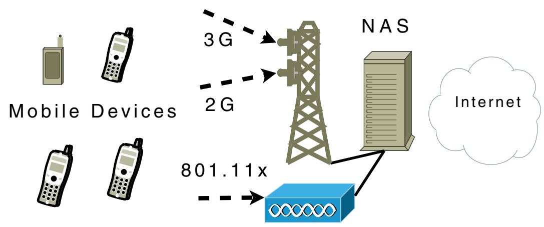

Fig. 1. MoSST testbed with smartphones, access points and NAS.

In recent years, a variety of bandwidth measurement tools have been developed ([4]–[6]), but few of them ([7]–[9]) have focused on wireless networks where capacity is rapidly varying. In order to be efficient in such an environment the measurement tool has to converge quickly, so that the state of the channel is relatively stable during the measurement phase. Moreover, it is desirable that the tool is non-intrusive, i.e. it should not itself use a large portion of the available bandwidth. This is even more imperative in wireless networks, since they usually have lower capacities than wired ones. Mobile phones pose additional challenges, in that they have less accurate timers. In a personal computer one can get microsecond accuracy, whereas in the phones we utilized we have found it to be in the millisecond range.

With these considerations in mind we have developed a new bandwidth and RTT measurement tool *Zeus* that uses the packet pair technique ([13]) and works in two phases. First, a series of packet pairs is sent, as in WBest ([7]). A packet pair consists of two packets that are sent without any delay between them and thus no precision timing is needed. The receiving end observes the delay between the packets and this is translated to a capacity estimate of the channel. At the end of the first phase, the effective capacity is computed as the median of the capacities of the series of packet pairs. Then, once the effective capacity is known, it is used to compute the available bandwidth in the same way as in Spruce [14]: A second series of packet pairs is sent with time gap Δ_{in} and are received by the server with time gap Δ_{out}. Knowing the capacity of the channel (computed in the previous step) the available bandwidth can be derived from the following equation:

$$ABW = C * \left(1 - \frac{\Delta_{out} - \Delta_{in}}{\Delta_{in}}\right) \tag{1}$$

The developed tool is non-intrusive (only 40 packet pairs are sent) and achieves fast convergence times (less than a second). Notice that collaboration of the two ends in a link is needed, as is the case with the majority of available bandwidth measurement tools. Hence, the tool is implemented as a client-server program in Python programming language using PyS60 platform on smartphones and Python scripts on the NAS.

B. Measurement and Analysis

Measurements were made using multiple mobile phones that connect to a NAS through a common access point. We have made measurements on three different access networks:

TABLE I

AVAILABLE BITRATE AND ROUND-TRIP TIME STATISTICS

	Avg ABW	Std ABW	Avg RTT	Std RTT
WiFi (802.11g)	3.5145	0.6642	9.0988	9.8624
3G (UMTS)	0.3861	0.0130	186.5459	15.6154
2.5G (GPRS)	0.0388	0.0017	297.9426	34.9475

WiFi, UMTS (3G) and GPRS (2.5G). Even though we did not artificially create any background noise, the measurements were done in a busy environment where base stations were shared with other users and all common effects of wireless channels were present such as mobility, shadowing, etc.

Table I shows the first and second order statistics of the measurements. It is obvious that going from 802.11 to UMTS to GPRS the available bandwidth decreases an order of magnitude. The roundtrip times are also much lower for the WiFi network, as was expected, since the access point was in the same local network as the server. This is not the case for UMTS and GPRS as we did not have access to the wireless base stations to locate the NAS directly next to them, which would have decreased the RTTs significantly. The histograms for the available bandwidth and roundtrip times on the 3G network are depicted on Figure 2. In line with the computed statics, both histograms are spread over a wide range of values and indicate high variability of wireless network characteristics.

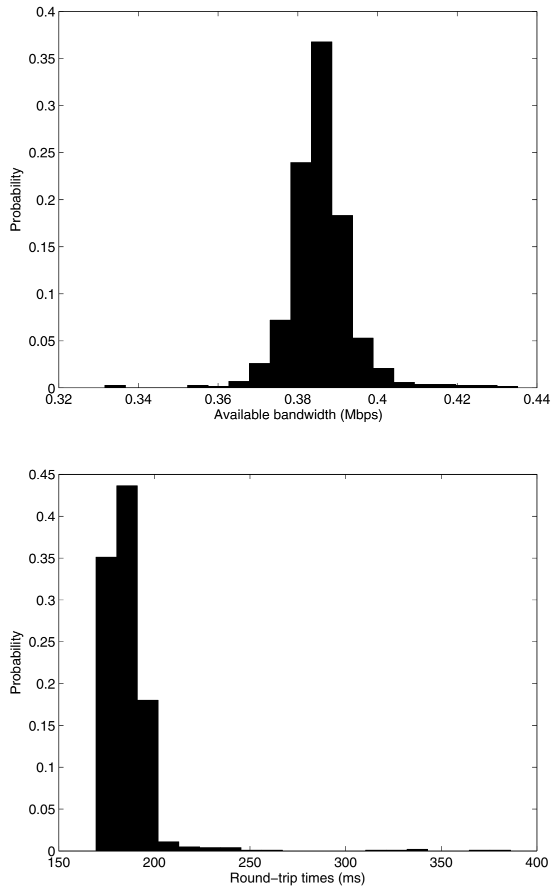

Fig. 2. Histograms of available bandwidth (top) and round trip times (bottom) on the 3G network.

III. DYNAMIC BANDWIDTH MODELING

As observed from the experiments in the previous section, available bandwidth and round trip times of wireless networks vary significantly over time. To model these dynamic process, we resort to a set-oriented stochastic approach and use finite-state Markov chains. It has been shown that Markov models are suitable for describing dynamics of wireless networks ([15], [16]). Let $S = \{1, 2, .., K\}$ be the states of the Markov model (their choice will be explained later in this section) and let $s(n)$ be the state at time n. The elements of the transition matrix P are defined as:

$$P_{ij} = \mathrm{Prob(s(n+1) = j|s(n) = i)},$$

where P_{ij} is the probability of transitioning from state i to state j in the next time step. Subsequently, the stochastic system is described by

$$p(n+1) = p(n)\ P,$$

where $p(n)$ is the probability vector over the state space S at time n.

To learn the transition matrix from a set of observed data, the assumption is made that P is stationary for at least for some time period. Under this assumption, we use a standard set-oriented method [17] to compute P. The observed data are split in pairs and the sets C_k^{in} and C_l^{out}, which contain input and output data points k and l, are computed. Subsequently,

$$P_{ij} = \frac{\sum_{[k:C_k^{out} \in j]}}{\sum_{[l:C_l^{in} \in i]}}$$

where $\sum_{[k:C_k^{out} \in j]}$ is the number of data points k such that $C_k^{out} \in j$. Given enough samples this procedure eventually approximates the Markov transition matrix.

The previous procedure works theoretically for any chosen quantization scheme. However, its performance crucially depends on the choice of states. There exists naturally multiple methods for defining the state space. One naive option is to adopt a uniform quantization scheme on the range of values. This approach has the following two issues:

- The data does not necessarily follow an uniform distribution. If some values are more probable than others, then it would be desirable to have more states around them.
- In the case of multi-dimensional data the number of states grows exponentially. If there are three dimensions and 10 uniformly quantized states are used for every dimension, then the transition matrix would grow to be $1000 * 1000$ in size. This brings a computational burden and most importantly it takes a lot of data to train it.

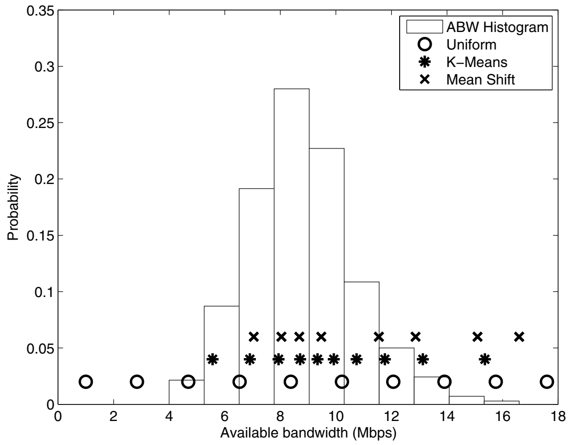

Fig. 3. ABW quantization using different methods on a single network.

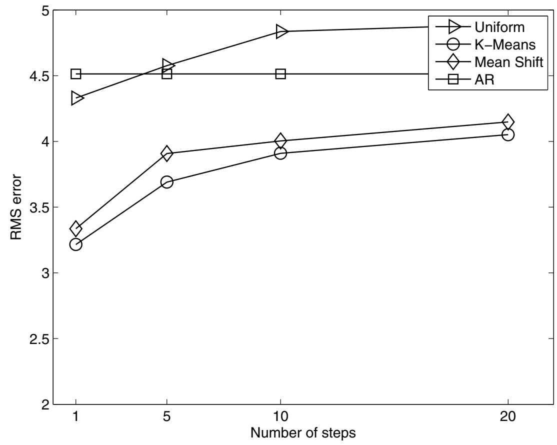

Fig. 4. Comparison of RMS errors for different prediction schemes.

A more intelligent approach to describing the state space is to borrow a technique from the machine learning community: clustering is a procedure that takes a set of data points and divides them into clusters, so that points in the same cluster are more related than points in different clusters. The centers of the clusters provide a good choice for states of the Markov model.

One of the most common and established clustering techniques is K-Means, where the only input is the number of clusters K. K centers are initially chosen at random. Then, all the data points are assigned to their closest center and the centers are recalculated iteratively. A more modern technique is Mean-Shift [18], which does not require the number of clusters to be defined. The user defines a window size around the centers and then Mean-Shift the significant cluster centers that do not fall in the same window. Both techniques are essentially gradient methods and as such are computationally light.

Figure 3 illustrates how the states are chosen given the probability distribution of available bandwidth of a single network. The uniformly spaced points are not a proper choice in this case, because most of the data lie around the center. K-Means using $K = 10$ clusters produces much more satisfactory results. The points obtained by Mean-Shift also seem to be around the center, even though there is a couple of outliers that are probably caused by local maxima encountered.

The Markov model using various quantization schemes to determine the states was evaluated on the dataset collected in Section II. As a baseline, the mean of the previous values in the training window can be used as a prediction for the future. All the methods were trained using a window of $W = 80$ data points and the results were used for the next 20 steps. After these 20 steps the methods are re-trained on the latest W data points. The metric used is the RMS error of the predictions versus the real measured values.

As seen in Figure 4, K-Means and Mean-Shift based quantization have comparable performance, with K-Means being consistently better. They both outperform the uniformly quantized states by a large margin. In fact, after a point even predicting the mean value is better than using the Markov model with uniform states. This highlights a significant result: Markov models can indeed be very powerful, but only if the right state space is used. Furthermore, by determining states using the intelligent clustering techniques not only the performance of the model is greatly enhanced, but also the number of states is decreased by orders of magnitude. This is especially important since it leads to shorter training periods, which is crucial for the underlying stationarity assumption to hold and less computational and storage requirements.

IV. Network Assisted Rate Control

We implement a distributed rate control scheme with QoS guarantees in order to showcase the value of the dynamic bandwidth model and estimator. We run simulations based on the real data obtained via measurements and compare its performance to a standard AIMD rate control scheme. The distributed rate control scheme is based on [1] and can be seen as an example network assisted computing application. It solves the following dynamic network utility maximization problem

$$\begin{array}{ll} \text{maximize} & \sum_{t=\tau}^{T} \sum_{j} U_{jt}(f_{jt}) \\ \text{subject to} & R_\tau f_\tau \leq C_\tau \\ & R_t f_t \leq \hat{C}(t|\tau), \quad\quad t = \tau+1, \ldots, T \\ & Q_j f_j \geq q_j, \quad\quad j = 1, \ldots, n \\ & 0 \leq f_{jt} \leq f_{jt}^{max}, \quad t = \tau, \ldots, T, j = 1, \ldots, n \end{array} \tag{2}$$

There are n flows in the system, indexed by j. U_{jt} is the utility of flow j at time t as a function of its rate f_{ij}. R_t is the routing matrix, f_t is the vector of rates and C_t is the link capacity, all of them at time t. Notice that $C(\tau)$ refers to the current (measured) capacity and $\hat{C}(t|\tau)$ is the expected capacity of the link given its history. It is the quality of this prediction that affects the quality of the solution.

Q_j and q_j refer to the notion of delivery contract indicator matrix (defined over time), a requirement that the total of flow j in some time interval should meet or exceed some specified minimum quantity q:

$$\sum_{t=t^{init}}^{t^{fin}} f_{jt} \geq q$$

This QoS formulation is well-suited for multimedia communications. Because of the buffering that happens on the receiver side it is more important to ensure that the buffers do not go empty than to focus on the instantaneous rate. Thus, in practice the delivery contracts correspond to QoS guarantees that the system makes to the user.

The optimization problem is solved in a MPC (model predictive control) manner. The utility is maximized over a time horizon and from the solution that covers this horizon only the first step is used. At the next time step the optimization problem is re-solved and again only the first step of the solution is used. Of course this increases the computational load especially in the setting of a mobile device with limited battery and computing power. Therefore, we have also implemented a version that utilizes the first five steps of the solution instead of recomputing at each step, in our simulations. The optimization problem 2 can be solved by the following distributed algorithm:

$$\begin{aligned}
p_{jt} &= (R_t^T \lambda_t)_j - (Q_j^T \mu_j)_t & t &= 1, \ldots, T \\
f_{jt} &= \operatorname*{argmax}_{0 \leq z \leq f_{jt}^{max}} (U_{jt}(z) - z p_{jt}), & t &= 1, \ldots, T \\
\lambda_t &= (\lambda_t - \alpha(C_t - R_t f_t))_+, & t &= 1, \ldots, T \\
\mu_j &= (\mu_j - \alpha(Q_j f_j - q_j))_+, & j &= 1, \ldots, n
\end{aligned} \tag{3}$$

where λ_t is the vector of link prices (as in [10]), μ_j is the vector of contract subsidies for flow j and $\alpha > 0$ is the step size, an algorithm parameter.

In our simulations, we assume one flow per mobile user and use $n = 10$ flows. We consider a single link case where all flows share the same link to the server. Thus, the routing matrix is constant. The horizon is taken to be $T = 50$. The delivery contracts span the horizon and the QoS targets q are randomly chosen. The utility function for the optimization problem 3 is a standard logarithmic one. To evaluate the algorithms the packet losses were also taken into consideration, by discounting them from the logarithmic utility. The simulations are run in Matlab, based on the dataset collected on the 802.11g network in Section II.

The following methods are compared: an exact implementation of 3 where our K-means based Markov model is used to predict the available bandwidth, a modification of the previous method where the five first steps of the optimal solution are used to decide on flow rates, another implementation of 3 using the mean-based bandwidth estimation and finally an AIMD scheme, where on a packet loss the flow rate is reduced to 75% of the previous one and on a success packet transmission the rate increases by 10 Kbps.

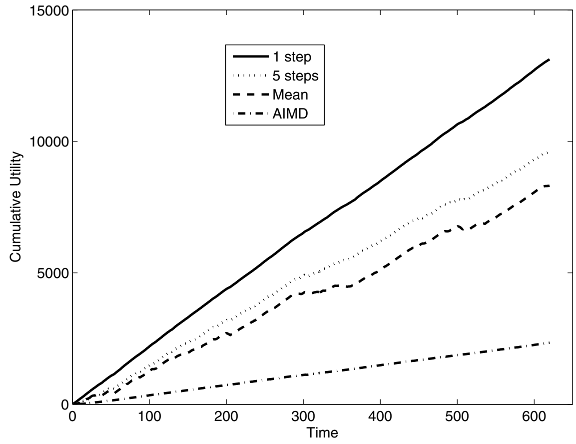

Fig. 5. Cumulative utility

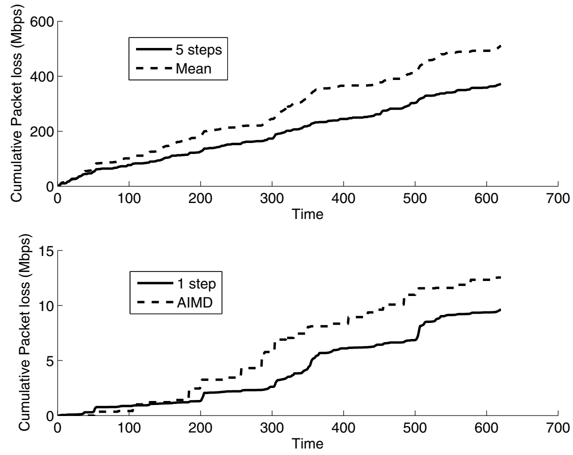

Fig. 6. Cumulative packet losses

Figure 5 shows the cumulative utilities achieved using the described algorithms. It is obvious that the AIMD method performs significantly worse than the other methods. While the other methods use the available bandwidth measurements and predictions to make more informed decisions, AIMD makes its decision based solely on whether a packet was transmitted successfully or not. Figure 6 shows the cumulative packet losses for the algorithms. It is split in two because there are two groups of algorithms with performances that differ by an order of magnitude. AIMD has very low packet losses, because whenever a packet is lost it substantially backs off. The 1-step version of the algorithm also has insignificant losses. This is due to the fact that not exceeding the capacity at the first step is one of the constraints of the optimization problem 2. In practice the constraint is slightly violated, either because the algorithm stops before it has fully converged or because of infeasibility problems. The other two algorithms have higher losses since they are making future decisions without exactly knowing the future capacity. As expected, the K-Means based method works better than the mean-based method since its predictions are more accurate.

V. Conclusion

We address the problem of modeling bandwidth and delay for wireless networks with heterogeneous characteristics. Towards this end we develop and implement a tool *Zeus* on smartphones to measure available bandwidth and round-trip times on WiFi and cellular networks. Employing the measurement results we create a Markov model to predict future variations in network characteristics. The model is enhanced via machine learning based clustering techniques for intelligent state quantization. The accuracy of these predictions is then tested against the real data. The developed model is then utilized evaluated within a distributed rate control framework.

In general, a network assisted computing is a promising paradigm as illustrated by our the distributed rate control with QoS guarantees assisted by a NAS. Our ongoing efforts include evaluation of flow assignment policies in multi-homed environments. We are also investigating a game-theoretic approach to rate control.

References

[1] N. Trichakis, A. Zymnis, and S. Boyd, "Dynamic network utility maximization with delivery contracts," in *Proc. of the 17th IFAC World Congress*, Seoul, South Korea, July 2008.

[2] J. P. Singh, T. Alpcan, P. Agarwal, and V. Sharma, "A markov decision process based flow assignment framework for heterogeneous network access," *ACM Wireless Networks*, (accepted, to appear).

[3] T. Alpcan, J. Singh, and T. Basar, "Robust rate control for heterogeneous network access in multihomed environments," *Mobile Computing, IEEE Transactions on*, vol. 8, no. 1, pp. 41–51, Jan. 2009.

[4] V. J. Ribeiro, R. H. Riedi, R. G. Baraniuk, J. Navratil, and L. Cottrell, "Pathchirp: Efficient available bandwidth estimation for network paths," in *Passive and Active Measurement Workshop*, 2003.

[5] M. Jain and C. Dovrolis, "End-to-end available bandwidth measurment: Measurement methodology, dynamics, and relation with tcp throughput," in *Proc. IEEE SIGCOMM*, Pittsburgh, Pennsylvania, USA, Aug. 2002.

[6] J. Navratil and R. L. Cottrell, "Abwe :a practical approach to available bandwidth estimation," presented at the PAM'03, 2003.

[7] M. Li, M. Claypool, and R. Kinicki, "Wbest: a bandwidth estimation tool for ieee 802.11 wireless networks," in *Proceedings of 33rd IEEE Conference on Local Computer Networks (LCN)*, 2008.

[8] S. H. Shah, K. Chen, and K. Nahrstedt, "Available bandwidth estimation in ieee 802.11-based wireless networks," in *roceedings of 1st ISMA/CAIDA Workshop on Bandwidth Estimation*, 2003.

[9] H. Fu, L. Lin, and W. Jia, "Efficient wireless link bandwidth detection for ieee 802.11 networks," in *ICC '05: Proceedings of the IEEE International Conference on Communications*, 2005.

[10] F. Kelly, A. Maulloo, and D. Tan, "Rate control for communication networks: shadow prices, proportional fairness and stability," *Journal of the Operational Research Society*, vol. 49, no. 3, pp. 237–252(16), March 1998.

[11] Y. Yi and M. Chiang, "Stochastic network utility maximization," *European Transactions on Telecommunications*, 2008.

[12] M. Mirza, J. Sommers, P. Barford, and X. Zhu, "A machine learning approach to tcp throughput prediction," in *SIGMETRICS '07: Proceedings of the 2007 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*. New York, NY, USA: ACM, 2007, pp. 97–108.

[13] S. Keshav, "Packet-pair flow control," *IEEE/ACM Transactions on Networking*, 1995.

[14] J. Strauss, D. Katabi, and F. Kaashoek, "A measurement study of available bandwidth estimation tools," in *IMC '03: Proceedings of the 3rd ACM SIGCOMM conference on Internet measurement*. New York, NY, USA: ACM, 2003, pp. 39–44.

[15] A. Konrad, B. Zhao, A. Joseph, and R. Ludwig, "A markov-based channel model algorithm for wireless networks," *Wireless Networks*, pp. 189–199, September 2003.

[16] K. Salmatian and S. Vaton, "Hidden markov modeling for network communication channels," in *Proc. ACM SIGMETRICS*, 2001.

[17] G. Froyland, *Nonlinear Dynamics and Statistics: Proceedings, Newton Institute, Cambridge, 1998*. Birkhauser, 2001, ch. Extracting dynamical behaviour via Markov models, pp. 283–324.

[18] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002.