ENERGY-EFFICIENT MULTIMEDIA COMMUNICATIONS IN LOSSY MULTI-HOP WIRELESS NETWORKS

Al Harris, Cigdem Sengul, Robin Kravets, and Prashant Ratanchandani
Department of Computer Science
University of Illinois at Urbana-Champaign
{aharris,sengul,rhk}@cs.uiuc.edu

Abstract A key concern in multi-hop wireless networks is energy-efficiency due to battery-power constrained mobile nodes. The network interface is a significant consumer of energy [7, 8, 15] causing a substantial amount of energy to be wasted by sending packets that cannot be used by the receiver. Given the small MAC layer packet sizes of wireless channels as compared to multimedia application data frames, inter-packet dependencies are formed (*i.e.*, the loss of a single packet renders a group of packets useless). In this paper, we present an application-aware link layer protocol to reduce the energy wasted by sending such useless data in lossy networks.

Keywords: Energy-efficient design, application-aware MAC, multi-hop wireless networks

1. Introduction

The increase in the availability of mobile computing devices has led to the design of communication protocols for multi-hop wireless networks. However, wireless networking poses implementation challenges due to characteristics such as resource constraints (*e.g.*, battery power) and lossy links. Our research addresses the issue of energy-efficient multimedia communication in such wireless networks by revisiting per-hop mechanisms.

The key observation for wireless communication is that energy consumption is affected by decisions at all layers. Essentially, although high quality communication is an end-to-end issue, lossy communication channels are a MAC layer issue. Therefore, a cross-layer approach is necessary in multimedia communication where there are dependencies between packets. Blindly processing each packet at the MAC layer may waste energy from the transmission of unusable data. The majority of current research in multi-hop wireless networks assumes that each packet is an independent application layer frame, and so the loss of one packet does not impact the correctness of others. However, the small size of MAC layer packets (1.5KB for IEEE 802.11 [11]) does not support the framing of larger application layer data frames (*e.g.*, on the order of 10KB for MPEG I-frames). This mismatch forces applications to fragment their data across multiple MAC layer packets, creating dependencies between such packets (*i.e.*, the loss of one packet implies the loss of the entire frame). Therefore, while information about link quality is essential to achieve effective communication, the type and characteristics of application data also impact the success of energy conservation techniques.

The contribution of our research is an investigation of the impact of per-hop mechanisms on end-to-end communication in the presence of lossy communication channels. Energy savings are achieved by exposing application layer framing [5] information to the link layer. Our approach is based on the observation that data transmitted to the receiving application, but not usable by the application, represents wasted energy. To prevent the transmission of partial frames to the receiver, we propose the use of a link layer mechanism that tracks the transmission of individual packets of an application layer frame at each hop, dropping all the packets of a frame if a single packet is lost. This mechanism is based on the idea that a partial frame should be dropped as soon as possible and only packets belonging to complete frames should be forwarded. Additionally, combined with a hop-by-hop reliability mechanism (*e.g.*, as in IEEE 802.11 [11]), the proposed approach also compensates for transmission-based losses and increases energy efficiency.

This paper is organized as follows. Section 2 presents the motivation for developing energy-efficient protocols. Section 3 defines two performance metrics: *effectiveness* and *energy-efficiency*. Section 4 presents the application-aware link layer protocol in detail and Section 5 evaluates the effectiveness of our protocol via a simulation study. Finally, Section 6 presents conclusions and future directions.

2. Energy Management in Multi-Hop Wireless Networks

Battery capacity is not increasing in step with the energy requirements of new mobile computing technology. Therefore, methods of saving energy must be designed to enable the use of mobile computing devices. Our approach uses application framing-aware link layer mechanisms to reduce the number of incomplete frames transmitted through the network. In this section, we discuss our approach in the context of current research in energy management and dropping policies in single-hop and multi-hop wireless networks.

2.1 Energy-Aware Communication

The concern for saving energy has spawned a large body of work on energy-saving routing protocols. In general, energy-aware routing protocols for multi-hop wireless networks focus on load balancing based on energy consumption and routing efficiency [3, 4, 23, 25]. However, such work does not consider efficient transmission of data once the energy efficient route has been found. Essentially, energy-efficient routing and our data-centric approach are orthogonal, so their benefits can potentially be combined.

FEC-based error recovery solutions [1, 10, 12, 14, 19, 24, 26] add overhead in the form of extra data needed to recover from errors and in the form of computation necessary to generate the codes for such recovery. For an FEC-based solution to be optimal, the code must match the error rate of the link. If the error rate is over-estimated, too much extra data will be included, wasting bandwidth and adding to the delay at each hop. However, if the error rate is under-estimated, all errors will not be corrected. For the estimation to be accurate across a multi-hop wireless environment, the FEC-code must be recalculated at each hop, due to the fact that each hop has varying error rates.

Another energy-saving is reducing the transmitter energy [9, 18, 20], which reduces the amount of energy used during transmissions at the cost of reducing the effective transmission range, rate, and/or reliability. This type of solution has no concern for data relations between packets or for loss recovery.

2.2 Supporting End-to-End Communication with Hop-by-Hop Mechanisms

The judicious use of hop-by-hop mechanisms has been discussed in the context of both congestion control and support for end-to-end communication in last-hop wireless environments. While research in both these areas does not address energy management, the techniques are similar.

The problem of fragmentation wasting resources has been known for a long time: *The loss of any one fragment means that the resources expended in sending the other fragments of that datagram are entirely wasted [13].* While such waste could be reduced by ensuring the transmission of one application layer frame per link layer packet, such intelligent fragmentation may not always be possible. For example, the problem of having larger application data frames than ATM cells has been explored by Floyd [22]. In general, *fragmentation-based congestion collapse* is caused by bandwidth being wasted through the transmission of fragments of packets that cannot be reassembled at the receiver into valid packets due to the loss of some of the fragments during a congested period of the network. ATM with partial packet discard was shown to be helpful in combating such fragmentation-based congestion collapse [2, 16, 17]. One major difference between the partial packet discard approach and our approach is the desire to eliminate the transmission of partial frames. With partial packet discard, the initial packets (or cells) are transmitted until a loss is encountered. These packets traverse the entire path and are discarded at the receiver. While the latter packets do not add to congestion, these initial packets are not dropped. In our approach, we add a minimal amount of buffering which allows us to delay sending packets from any frame in which some packets are missing to prevent energy consumption from the initial packets.

In a similar vein, [21] proposed dropping subsequent packets of a video frame after the loss of one of the packets containing a fragment of the video frame. The dropping mechanism presented in Section 4 is an application of this technique to multi-hop wireless networks. Essentially, the focus of previous applications of packet discard techniques has been to avoid congestion collapse, however, even in the absence of congestion, when energy conservation is also a concern, packet discard techniques can be used to achieve energy savings, as will be shown in Section 5.

3. Protocol Effectiveness and Energy Efficiency

The difficulty in designing MAC protocols is finding a balance between effectiveness and energy-consumption. We define *effectiveness* to be the goodput, or percentage of the data received at the end host that is usable by the end host application. It is clearly possible for an energy-conserving protocol to save energy but at the cost of very poor protocol effectiveness. For example, if an energy-aware protocol indiscriminately drops packets based on battery levels of the mobile nodes and a certain application has a frame size that spans two packets, it is possible that many of the packets that are received at the end host are unusable due to the fact that half of the frame was dropped by the protocol. In this situation, many of the received packets actually constitute wasted energy since the data in the packets is useless to the receiving application.

To factor the effect of useless packets into energy consumption analysis, we use an *energy efficiency* metric. Energy-efficiency (E_P) is the ratio of the number of usable packets (P_G) received at the end-host to the total energy (E_T) used in the transmission of a data stream ($E_P \equiv \frac{P_G}{E_T}$). Energy-efficiency ($E_P$) can be used to evaluate the effect of an energy-aware protocol on the application as well as its effect on the total energy consumption of the network and so, deciding whether or not the protocol is actually useful.

4. Application-Aware Link Layer Protocol

Our approach to providing energy-efficient transport of data in a multi-hop wireless network uses information about application layer framing at the link layer. Through the use of knowledge about application layer framing, the link layer makes intelligent decisions to improve protocol effectiveness and energy-efficiency. To this end, we use two mechanisms, an intelligent dropping mechanism described in Section 4.2 and a link layer retransmission mechanism described in Section 4.3. The intelligent dropping mechanism achieves better energy-efficiency by reducing the number of unusable packets transmitted by dropping partial frames at each hop in the path. Furthermore, enabling retransmissions at the link layer achieves better energy-efficiency by reducing the number of unusable packets received at the end host (*e.g.*, as in IEEE 802.11b [11]). The transport protocol used on top of the proposed link layer protocol is described in Section 4.1.

The parameters that affect the performance of the link layer mechanisms are: the application layer frame size, the loss rate of the links, the hop count, and the mobility rate. The effects of variations in these parameters are explored in the rest of this section and in Section 5. Essentially, the proposed mechanisms achieve protocol effectiveness and energy-efficiency at the expense of an increase in delay and a small amount of extra buffer space at each node. These details are explored further in the remainder of this section.

4.1 Transport Protocol Support

To develop an energy-efficient link layer protocol for wireless multi-hop networks, some information about application layer frames needs to be exposed to the link layer. To this end, a two field header is added to each packet, which is filled at the transport layer. The first field, *frame_no*, contains the application frame number of the data. The second field, *frame_size*, contains the number of packets for this application layer frame. These fields contain the main parameters that are used by the link layer mechanisms. It is important to note that if the fields are not present, the link layer performs like a traditional link layer. This provides backwards compatibility. Also, packets sent with *frame_size* equal to one are treated as if being sent by a regular link layer and are transmitted immediately.

4.2 Intelligent Dropping Mechanism

The dropping mechanism is based on the simple idea that if any part of a frame is lost, the entire frame is useless. Therefore, each node only sends packets containing fragments of complete frames by using a simple mechanism previously proposed for wired networks for fragmented packets. Essentially, an extra buffer capable of holding two frames of data is kept at each node, where packets are buffered until a full frame is received. Once all the packets of a frame have been received, the packets in the frame are placed at the the tail of the send queue. Using this mechanism, no node sends any packets of a frame for which it is not in possession of the entire frame. This method dramatically reduces the number of incomplete frames received at any node in the path at the cost of increased delay from buffering the packets of a frame until all are received.

The link layer using this mechanism does not fail to send any packets that will be usable by the end host. Therefore, the only decrease in transmissions is due to the reduction in useless packets sent. There are two costs that are incurred by this protocol. The first is a delay cost due to the fact that packets are only sent after an entire frame has been buffered. The application layer frame size and the hop count directly affect this cost. As the application layer frame size increases the delay increases due to the fact that larger application layer frames are fragmented into a greater number of link layer packets that need to be buffered. Furthermore, as the hop count increases, the delay increases due to gathering all packets before forwarding. If the number of hops is k, the number of flows is f, and the number of packets in one application layer frame is N, for a shared channel we have the following: Normal Delay $\equiv f \times (N + k - 1)$ and Dropping Delay $\equiv f \times (N \times k)$. Therefore, the dropping mechanism creates a multiplicative delay, while a non-buffering link layer protocol incurs a linear delay with respect to the hop count and frame size. However, Normal Delay assumes perfect pipelining of packets at each hop, which may not be achievable in practice due to contention between nodes at both ends of the link. The increase due to the dropping mechanism is due to the collection of all packets in a frame at each hop, which may also potentially reduce such contention between nodes. The second cost incurred is the need for more buffer space at the nodes. The dropping mechanism requires enough buffer space to hold two frames. Therefore, the needed buffer size consequently increases as the frame size increases.

The higher the loss rate of each link, the more likely that partial frames will be created (through the loss of some of the packets in each frame). Therefore, as the loss rate increases, better energy-efficiency gains are expected. The overall results of this mechanism in Section 5 show not only better energy-efficiency than a standard link layer, but also lower total energy expenditures. As the rate of mobility increases, the chance of one half of a frame traversing one path, and another traversing another path increases. However, since this protocol is not currently mobility-aware, this causes frames that could have been transmitted successfully to be dropped. However, the benefits return once the flow settles to a new route.

4.3 The Retransmission Mechanism

To increase the number of usable frames received at the end host, the proposed link layer protocol utilizes link layer retransmissions. Essentially, MAC layer information about transmission errors is exposed to the link layer. When a transmission error is detected by the MAC layer, the packet involved in the error is retransmitted. An additional parameter for the retransmission mechanism is the number of times to attempt to retransmit a packet lost due to transmission error. For this paper each packet retransmission at the link layer is only attempted once. The packet retransmission limit is set to 7 for short packets and to 4 for long packets in IEEE 802.11b standard [11]. However, only one retransmission attempt is performed by our approach since multiple retransmissions cause additional delay and increase the power expenditure. If the retransmission of a packet fails, the rest of the remaining packets in the frame are dropped.

The cost of link layer retransmissions is extra delay added onto the delay incurred by the intelligent dropping mechanism. This delay is bound by the total number of retransmission attempts r for all flows: Retransmission Delay $\equiv (f+r)(N*k)$. The effects of the parameters considered on this mechanism are essentially the same as the effects noticed for the intelligent dropping mechanism. As the hop count and the application layer frame size increases, the delay cost increases. The retransmission mechanism is also not mobility-aware and, therefore, performance degrades as mobility speeds increase. As the loss rate increases, the total number of retransmissions also increase. This leads to an interesting difference. Namely, while the total energy used by a link layer protocol using retransmissions may be higher than a standard link layer protocol, the energy-efficiency improves due to higher goodput performance (See Section 5). Therefore, a link layer protocol using retransmissions is more effective than a standard link layer protocol.

5. Evaluation

In this section, we present results from our simulation of three different link layer protocols. The first protocol is the *standard* link layer which does no buffering, dropping or retransmissions. The second is a application-aware link layer protocol, which implements the intelligent *dropping* mechanism. The third is a link layer protocol with a *retransmission* mechanism. All are simulated with the ns-2 network simulator [6]. For the application data, we use a CBR stream modified to include the two field header described in Section 4.1. All simulations use the IEEE 802.11 MAC layer. The simulations run in a $1000 \times 1000m^2$ area with 50 nodes. AODV is used for routing. Additionally, we use *random waypoint* mobility model. The effect of three parameters are tested in the simulations: the application layer frame size, the mobility rate, and the link error rate. The link error rate is modeled using a random probability of each packet being dropped. Our simulation results represent an an average of five runs with identical traffic models but different randomly generated network topologies. The effectiveness of the protocols are evaluated using the following metrics: the number of complete frames received at the end host, the number of partial frames received at the end host, the average end to end delay, the total number of MAC layer transmissions, and the average hop count. Additionally, we provide comparisons based on the *energy-efficiency* metric described in Section 3.

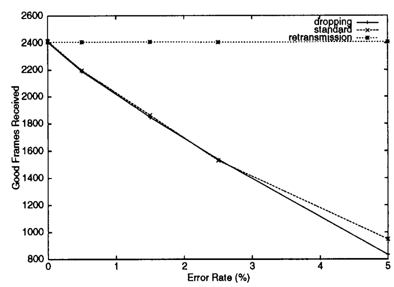

Figure 1. Complete frames at receiver.

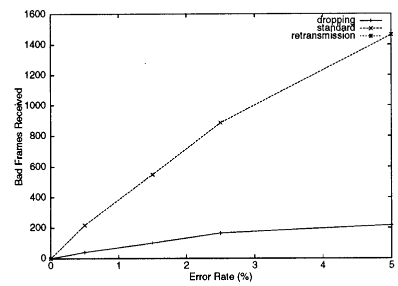

Figure 2. Partial frames at receiver.

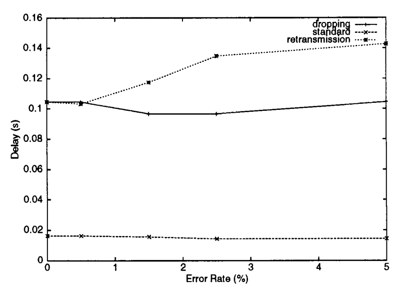

Figure 3. End-to-end Delay

5.1 Effects of Error Rate on Performance

To evaluate the impact of error rate, simulations were run that maintain an average hop count of five, a frame size of four, and a mobility rate of 0.1 m/s. The error rates used are 0%, 0.5%, 1.5%, 2.5%, and 5%. The simulation results show that the dropping mechanism delivers slightly fewer complete frames to the receiver than the standard link layer (see Figure 1). Since nodes do not forward partial frames, any loss on each link is from complete frames. As expected, the link layer with retransmissions delivers significantly more complete frames as the loss rate increases. However, the dropping mechanism successfully limits the number of partial frames delivered to the receiver (see Figure 2). The partial frames that do get to the receiver are due to errors on the last link. These can obviously not be avoided by a dropping mechanism.

The main cost of the proposed mechanism is the delay incurred by buffering. The results, as expected, show that the delay is proportional to the frame size (see Figure 3). This is due to the fact that at each hop, before the first packet in a frame can be sent, the rest of the packets in the frame must be received. For the 5 hop network with a frame size of 4, the delay cost is 80ms. As expected, our experiments show that the delay increases linearly with the size of the

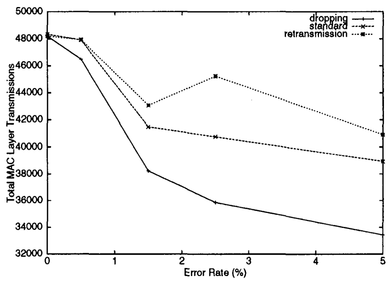

Figure 4. MAC layer transmissions.

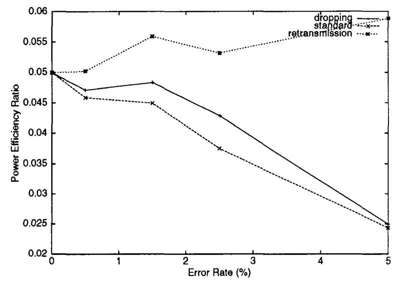

Figure 5. Power-efficiency.

application layer frames. As the error rate increases, the link layer using the retransmission mechanism incurs more delay as the need for retransmissions increases.

To evaluate the energy consumption of each protocol, the total number of MAC layer transmissions is used (since this metric translates directly to energy consumption). Figure 4 illustrates that the standard link layer falls in between the augmented link layers. The dropping mechanism has significantly fewer transmissions than the others due to the fact that it drops any incomplete frame. It is important to combine these results with the results depicted in Figure 1 to see that while many packets were dropped, this is not noticed by the end application since the number of usable frames remains approximately the same. Obviously, the retransmission mechanism uses more transmissions. Although this may be interpreted as failure of the retransmission mechanism to achieve the goal of being energy-efficient, the comparisons based on the energy-efficiency metric prove otherwise. Based on the definition of the energy-efficiency metric, higher energy-efficiency means a more efficient protocol. It is observed that the dropping mechanism achieves a high energy-efficiency by eliminating useless transmissions. The retransmission mechanism shows the best energy-efficiency ratio. This is accomplished by retransmitting lost packets and dropping frames that cannot be rebuilt.

5.2 Effects of Mobility on Performance

To evaluate the effects of mobility, the error rate is held constant at 1.5%, the average hop count is 3.8 and the frame size is four. Because neither of the link layer mechanisms are mobility-aware, we expect performance degradation as mobility increases. As the mobility rate increase to around 6 m/s, the dropping mechanism begins to drop frames that have parts that travel different paths (see Figure 6). The retransmission mechanism sustains its performance longer due to the fact that link breaks are fixed by the time the retransmission

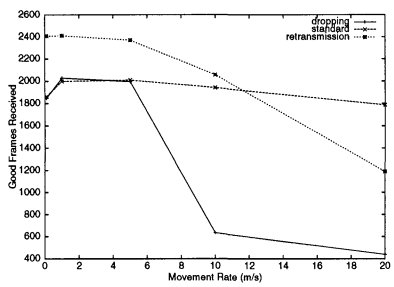

Figure 6. Mobility vs. goodput.

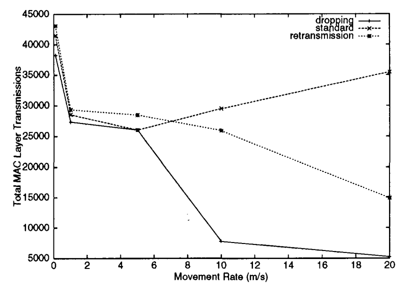

Figure 7. Mobility vs. MAC layer transmissions.

is attempted. However, the retransmission mechanism begins to deliver fewer complete frames to the end host than the standard link layer as the mobility rate increases beyond 11 m/s. Because the retransmission mechanism only attempts one retransmission, when that transmission fails, the simple dropping mechanism takes over.

Figure 7 compares the number of MAC layer transmissions in the presence of mobility. The link layer using the dropping mechanism has a significant drop in transmissions as it begins to drop most of the frames. However, these savings in energy are offset by the ineffectiveness of the protocol. The retransmission mechanism achieves fewer transmissions as the retransmissions begin to fail, and the simple dropping mechanism takes over. Again, the savings in energy is overshadowed by the ineffectiveness of the protocol.

6. Conclusions

Energy conserving protocols are essential for the operation of multi-hop wireless networks due to resource constraints (*e.g.*, battery power). However, such protocols should not only focus on saving energy without any concept of the effect on the application. We define a energy-saving protocol as effective if it maximizes the percentage of usable data received at the end host. Furthermore, we present a new metric for the energy-efficiency defined as the ratio of the number of usable packets received by the end host to the total energy used in transmission. Essentially, the energy-efficiency of a protocol takes into account not only the energy consumption of the protocol but also the effectiveness of the protocol. To this end, an application-aware link layer mechanism has been presented. This mechanism combined with a link layer retransmission mechanism makes use of application layer framing information to achieve significant gains in energy-efficiency. Future work involves adding mobility-awareness into the MAC layer mechanisms to handle route breaking and recovery.

References

[1] S. Aramvith and M.-T. Sun. A coding scheme for wireless video transport with reduced frame skipping. In *SPIE Visual Communication and Image Processing Conference*, 2000.

[2] G. Armitage and K. Adams. Packet reassembly during cell loss. *IEEE Network Magazine*, 7(5):26–34, September 1993.

[3] S. Banerjee and A. Misra. Minimum energy paths for reliable communication in multi-hop wireless networks. In *3rd ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, 2002.

[4] J.-H. Chang and L. Tassiulas. Energy conserving routing in wireless ad-hoc networks. In *IEEE INFOCOM*, 2000.

[5] D. D. Clark and D. L. Tennenhouse. Architectural considerations for a new generation of protocols. In *ACM Symposium on Communications Architectures and Protocols*, 1990.

[6] K. Fall and K. Varadhan. *ns Notes and Documentation*. The VINT Project.

[7] L. M. Feeney and M. Nilsson. Investigating the energy consumption of a wireless network interface in an ad hoc networking environment. In *IEEE INFOCOM*, 2001.

[8] P. Gauthier, D. Harada, and M. Stemm. Reducing power consumption for the next generation of PDAs: It's in the network interface! In *Workshop on Mobile Multimedia Communications MoMuc*, 1996.

[9] J. Gomez, A. T. Campbell, M. NaghShineh, and C. Bisdikian. PARO: supporting dynamic power controlled routing in wireless ad hoc networks. *winet*, 9:443–460, 2003.

[10] R. Han and D. Messerschmitt. A progressively reliable transport protocol. *Multimedia Systems*, 1999.

[11] IEEE 802 LAN/MAN Standards Committee. Wireless LAN medium access control (MAC) and physical layer (PHY) specifications. IEEE Standard 802.11, 1999.

[12] P. Karn. Toward new link layer protocols. In *Tucson Amateur Packet Radio (TAPR) symposium*, 1994.

[13] C. Kent and J. Mogul. Fragmentation considered harmful. In *ACM SIGCOMM '87*, August 1987.

[14] B. Kim, Z. Xiong, and W. Pearlman. Progressive video coding for noisy channels. In *Proceedings ICIP 98*, 1998.

[15] R. Kravets and P. Krishnan. Application-driven power management for mobile communication. *Wireless Networks*, 6(4):263–277, 2000.

[16] M. Labrador and S. Banerjee. Enhancing application throughput by selective packet dropping. In *IEEE ICC*, 1999.

[17] L. Leslie and D. McAuley. Fairisle: An ATM network for the local area. In *ACM SIGCOMM '91*, 1991.

[18] J. Monks, V. Bharghavan, and W. Hwu. A power controlled multiple access protocol for wireless packet networks. In *IEEE INFOCOM*, 2001.

[19] C. Parsa and J.J. Garcia-Luna-Aceves. Improving TCP performance over wireless networks at the link layer. *Mobile Networks and Applications*, 1:57–71, 2000.

[20] R. Ramanathan and R. Rosales-Hain. Topology control of multihop wireless networks using transmit power adjustment. In *IEEE INFOCOM*, 2000.

[21] S. Ramanathan, P. Rangan, and H. Vin. Frame-induced packet discarding: An efficient strategy for video networking. In *Fourth International Workshop on Networking and Operating System Support for Digital Audio and Video*, 1993.

[22] A. Romanow and S. Floyd. Dynamics of TCP traffic over ATM networks. *IEEE Journal on Selected Areas in Communications*, 13(4):633–641, May 1995.

[23] S. Singh, M. Woo, and C. S. Raghavendra. Power-aware routing in mobile ad hoc networks. In *4th Annual International Conference on Mobile Computing and Networking (MobiCom)*, 1998.

[24] S. Tabbane. *Handbook of Mobile Radio Networks*. Artech House Mobile Communications Library, 2000.

[25] C.-K. Toh. Maximum battery life routing to support ubiquitous mobile computing in wireless ad hoc networks. *IEEE Communications Magazine*, 39(6):138–147, 2001.

[26] M. Zorzi and R. R. Rao. Error control and energy consumption in communications for nomadic computing. *IEEE Transactions on Computers*, 46:279–289, 1997.