Top Five Myths about the Energy Consumption of Wireless Communication

Rex Min and Anantha Chandrakasan
Massachusetts Institute of Technology, 50 Vassar St, Room 38-107
Cambridge, MA 02139
{rmin, anantha}@mtl.mit.edu

Energy consumption has recently become an important consideration for wireless communication protocols. The shrinking size and increasing density of next-generation wireless devices imply reduced battery capacities, meaning that emerging wireless systems must use energy more efficiently than ever before. The energy-intensive nature of wireless communication has recently spurred protocol and MAC developments that explicitly seek to reduce the energy consumed by the communication subsystem [1,2].

Our recent work with integrated hardware, API, middleware, and software solutions for high-density wireless networks have revealed two general conclusions regarding energy-efficient communication. First, it is crucial that energy-efficient communication software layers be based upon sound, accurate models of the hardware on which they will operate. Incomplete or inaccurate energy models lead to surprising discrepancies between designers' expectations and real-world realities. This becomes especially true in emerging applications such as microsensor networks [3] where application and hardware characteristics differ greatly from today's norms. Second, two power management techniques used widely by hardware designers hold great promise for protocols: application-specific design and energy-quality scalability. We call upon these principles to dispel our version of the "Top Five Myths" about energy-efficient communication, which follow below.

5. "Communication energy scales with distance as d^n."

A power law correctly describes the radiated power necessary to transmit over a distance *d*. However, this path loss term alone fails to consider the energy overheads of the hardware. These components include the startup energy for the transceiver, the static (distance-independent) power drawn by the transmitter and receiver, power amplifier inefficiencies, coding energy, and protocol overhead. For short-range, high-rate radios, receiver energy will frequently exceed the energy of transmission. For sensor networks that transmit with short packet sizes, even the startup energy of the transceiver has been shown to exceed the energy of transmission [4]. To avoid surprises, energy-efficient communication protocols must be designed around accurate energy models of the targeted hardware.

4. "'Graceful degradation' doesn't apply to communication."

Low power digital hardware and computational algorithms frequently trade energy for quality. For instance, digital processing occurs more slowly—and consumes less energy—when the circuits' supply voltage is reduced. Dynamic voltage scaling therefore enables a graceful energy *vs.* latency trade off that can be adjusted to meet an application's changing needs. A system that is able to degrade performance gracefully in exchange for energy savings is ensuring that energy is not wasted by providing performance in excess of an application's needs.

Graceful energy *vs.* quality scalability for wireless communication can be achieved once the notion of communication "quality" is defined. Hence, we define communication quality by four of its fundamental metrics: *range, latency, reliability,* and *energy.* We then introduce an API that allows an application to specify bounds on these metrics. The latter three metrics can be bounded by direct specification:

- `set_max_latency(double usecs)`
- `set_min_reliability(double ber)`
- `set_max_energy(double ujoules)`

With cooperation from a protocol layer that maintains approximate distances to—and numbers of—neighboring nodes, the communication range desired by an application can be expressed in a variety of ways, whichever is most convenient to the application:

- `set_range(Distance d)`
- `set_range(int numberOfNearestNeighbors)`
- `set_destination(Node n)`
- `set_destination(Nodes n[])`

3. "Abstraction gets in the way of energy savings."

Abstraction layers are only as inefficient as their definitions. Our communication API allows an application to expose tolerable bounds on latency, reliability, range, and energy. Application designers that utilize wireless communication have historically been reluctant to manipulate hardware energy knobs such as processor voltages or transmit power. Introducing proper abstractions between communication software and hardware can therefore *encourage*—not hinder—energy savings.

To bridge the gap between these performance parameters and the actual hardware "knobs" for energy scalability, we introduce a power-aware middleware layer as illustrated in Figure 1. The power awareness manager is empowered with accurate hardware energy models for the node's digital processing circuits and radio transceiver, allowing this layer to select the minimum-energy hardware settings for the performance level commanded through the

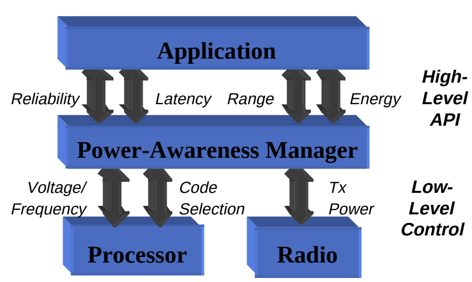

Figure 1: A power-aware interface bridges an application's quality requests for communication and the hardware's energy scalability.

API [5]. Figure 2 illustrates a sample operational policy for a wireless microsensor node [6], in which an API-specified reliability and range (x- and y-axes) are mapped to the radio transmission power and convolutional coding scheme that result in minimum energy consumption.

2. *"It is wise to refine 802.11b for energy-efficiency."*

In the VLSI world, *application-specific integrated circuits* (ASICs) consume far less energy than general-purpose circuits such as microprocessors. Reducing unnecessary functionality consequently reduces complexity and energy. With this in mind, we advocate the design of *application-specific protocols* that provide concise, energy-efficient solutions for specific applications. 802.11b and many popular *ad hoc* routing protocols, in contrast, are general-purpose solutions with overheads and capabilities that may not be necessary for specialized and emerging applications.

We have designed an application-specific protocol tuned for unidirectional data propagation in high-density microsensor networks. As relay nodes in a sensor network have no need for the data they are relaying, the entire notion of *addressing* a packet to a specific relay node is unnecessary. The only concern is that packets move progressively closer to a base station. Hence, we have designed a routing methodology that replaces explicit next-hop addresses with non-exclusive, high-resolution distance metrics to a base station. This *address-free forwarding* protocol, which bears similarities to gradient routing (GRAd) [7], is illustrated in Figure 3. This surprisingly simple protocol achieves performance comparable to 802.11b-based protocols while permitting greater flexibility in radio receiver shutdown.

1. *"Multihop saves energy."*

Traditional multihop routing seeks to reduce transmission energy by adding intermediate relay nodes. Comprehensive models of transmit and receive energy, however, lead us to reconsider this technique. Figure 4 illustrates the energy required for multihop communication with a 2.4 GHz commercial transceiver with a receive energy of 300 mW. For this transceiver, which is typical of Bluetooth applications, multihop is *less* energy-efficient than direct transmission until the radiated transmission power approaches its upper limit of +20 dBm, a level that requires nearly one watt of energy in the transmitter electronics! Additional overheads incurred by the underlying routing protocol will further reduce the efficiency of multihop. Hence, before immediately adopting multi-

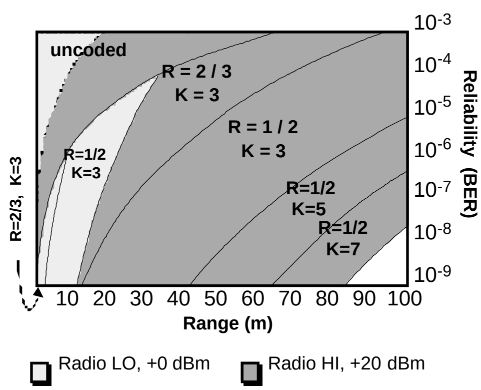

Figure 2: Least-energy hardware policy for single-hop communication given a specified reliability and range, considering both transmit and receive energy. *N* = 1000.

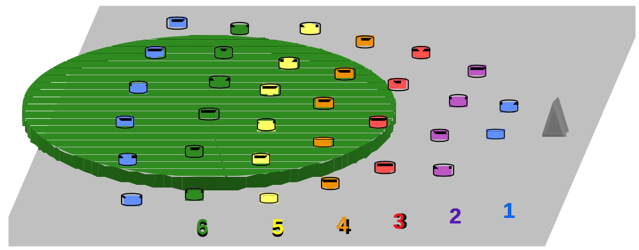

Figure 3: High-resolution distance metrics allow hop lengths to be maximized in a greedy fashion. For each hop, the farthest node from the sender becomes the next relay through the use of a deterministic relay timer whose value is inversely proportional to the distance between sender and receiver. Above, the distance-3 should relay.

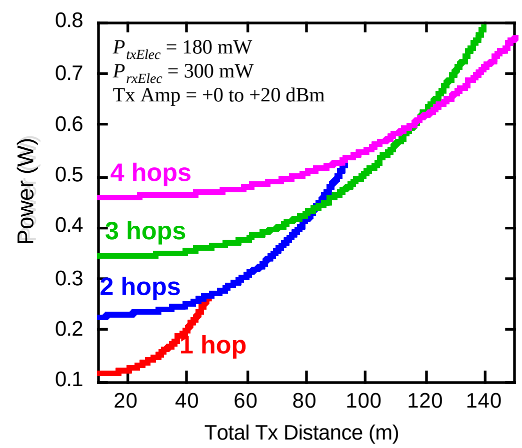

Figure 4: Contrary to popular belief, multihop energy does *not* conserve energy with this commercial 2.4 GHz radio.

hop routing as an energy-saving measure, accurate and thorough energy models of the node hardware must be considered. Multihop saves energy only when d^n path loss (see Myth #5) dominates the energy consumption of hardware, a case that occurs less frequently than is typically believed.

References

[1] B. Chen *et al.*, "Span: an Energy-Efficient Coordination Algorithm for Topology Maintenance in Ad Hoc Wireless Networks," ACM Wireless Networks Jnl, vol. 8, no. 1, 2002.

[2] Alice Wang, W. Heinzelman, and A. Chandrakasan, "Energy-Scalable Protocols for Battery-Operated Microsensor Networks," Proc. SiPS '99, Oct. 1999, pp. 483-492.

[3] G. Asada et al. Wireless Integrated Network Sensors: low power systems on a chip. Proc. ESSCIRC '98 (Sept. 1998).

[4] Andy Wang, S.-H. Cho, C. Sodini, A. Chandrakasan, "Energy efficient modulation and MAC for asymmetric RF microsensor systems," Proc. ISLPED 2001, August 2001.

[5] R. Min and A. Chandrakasan, "A Framework for Energy-Scalable Communication in High-Density Wireless Networks," Proc. ISLPED 2002, August 2002.

[6] E. Shih, et al. Physical layer driven algorithm and protocol design for energy-efficient wireless sensor networks. Proc. MobiCOM 2001 (July 2001).

[7] R. Poor. "Gradient Routing in Ad Hoc Networks," *www.media.mit.edu/pia/Research/ESP/texts/poorieeepaper.pdf*