Power Consumption Trade-offs for Wireless Audio Access

Johan Pouwelse* Koen Langendoen Henk Sips
Faculty of Information Technology and Systems
Delft University of Technology
Mekelweg 4, 2628 CD Delft, The Netherlands
Tel: [+31]15-2782539; fax: [+31]15-2786190
E-mail: `{pouwelse,koen,sips}@ubicom.tudelft.nl`

Abstract

This paper studies the power consumption and trade-offs of a wearable device playing audio streamed over a wireless connection. Our experimental platform is a credit card sized device, capable of receiving, decoding, and playing compressed audio. We present the power consumption of four audio decoders and four Wireless LAN cards under various conditions. Decoding MP3 files takes just 26 mW of processor power. The best Wireless LAN card (WaveLAN) consumes 74 mW when idle, and an additional 1.1 mJ/Kb when receiving data. By combining the decoding cost for several compression algorithms and the corresponding wireless link cost we determine the optimal configuration (MP3 with WaveLAN).

Keywords: power consumption, wireless LAN, audio decoding, power measurements, low-power trade-offs

1 Introduction

Technological advances fuel the interest in small wearable devices and high-bandwidth wireless Internet. Technology has arrived that can deliver wireless access to multimedia information such as hypertext, audio, images, video, and 3D graphics. This paper focuses on the *wireless audio access* scenario where a server on the Internet streams compressed audio information to the user, who is carrying a wearable device with a high-bandwidth wireless link. Since the primary concern of any wearable is the limited amount of battery energy available, it is important to understand the trade-offs involved in designing and configuring a wearable audio playback device. It might even be necessary to adapt the configuration dynamically, because the quality of the wireless link depends on the environment. The following three components account for most of the energy consumed in a wearable playback device:

- Receiving compressed audio over the wireless link
- Decoding the compressed audio
- Outputting the analog audio signal

Within each of these components there are trade-offs to be made for the quality and the cost of resources. More interesting, however, is that trade-offs *between* the first two components can be made; heavily compressed audio requires extensive decoding with moderate data rates across the wireless link, while lightly compressed audio requires little processing for decoding but induces a high data rate. Which scenario is best depends on the power consumption of the available audio decoders and the effectiveness of the wireless link (at the current time) in handling the associated data rates.

The trade-off in power consumption between the wireless link and audio decoding is the focus of this paper. We present detailed measurements of various IEEE 802.11 compliant wireless PCMCIA cards and multiple audio decoders including MPEG1 (layers 1, 2, and 3). The measurements are the necessary prerequisite to explore the range of low-power configurations. We are especially interested in adapting the configuration of the wearable playback device during operation in response to changes in the environment. In particular, we take distance and network type into account.

2 Measurements

To determine the power consumption of a wearable device with wireless audio access we conducted measurements on a working system and four different wireless LANs. At Delft University of Technology we have been developing a small embedded device, called *LART* (Linux Advanced Radio Terminal) [7]. The

---

[*] Supported by the Dutch organization for Applied Scientific Research (TNO), Physics and Electronics Laboratory.

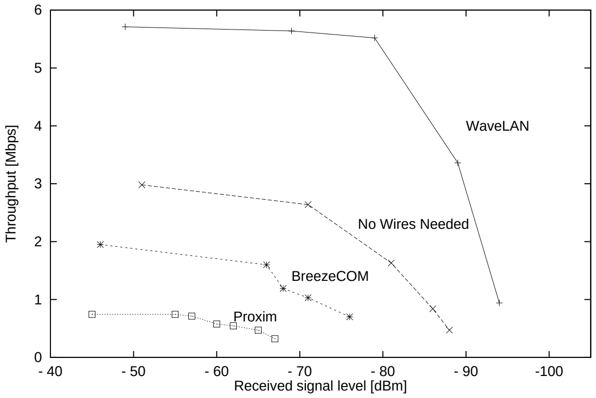

Figure 1: Throughput vs. wireless link attenuation.

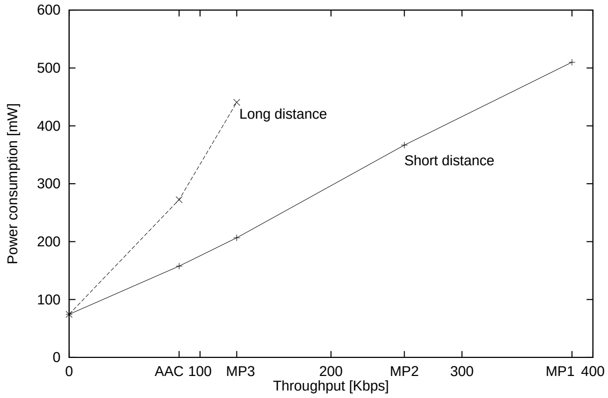

Figure 2: WaveLAN performance and cost.

size of the LART is roughly equal to two credit cards, and it can be powered by three pen-light batteries. The LART uses a power efficient 190 MHz StrongARM1100 processor, and can decode compressed audio and play it with CD quality. The LART supports frequency and voltage scaling, so power consumption can be minimized by matching CPU performance to decoding requirements. We will refer to the combination of a LART, wireless link, and audio decoding software as the *LARTman*.

We tested two types of audio compression algorithms on the LARTman: A-law and MPEG1 (layers 1, 2, and 3). The MPEG1 decoder is the publicly available splay decoder by Woo-Jae Jung, optimized and rewritten for fixed point by Nicholas Pitre. The A-law decoder was written from scratch and fine-tuned for the LARTman. A-law yields the lowest compression factor and reduces the audio steam from 1.4 Mbps to 705 Kbps. The three MPEG1 layers offer medium to strong compression: MP1=384 Kbps, MP2=256 Kbps, MP3=128 Kbps.

2.1 Wireless link

The wireless link is responsible for the transfer of information with minimal or no corruption. The IEEE 802.11 Wireless LAN (WLAN) standard [6] defines a method to transfer several Mbps across a distance of up to a few hundred meters. Several products, all PCMCIA cards, are on the market that implement this standard. We measured the performance of four of those products operating in the 2.4 GHz band: WaveLAN (recently renamed to Orinoco), No Wires Needed, BreezeCOM, and Proxim.

The performance measurements were conducted by connecting the specific WLAN PCMCIA card to the corresponding base station through a link with controllable and calibrated attenuation. The base station and WLAN card are placed in two separate, shielded metal boxes; the antennas are bypassed and the output signals are fed directly to the attenuator. Figure 1 shows the performance of the four WLAN cards under different conditions. The performance of WaveLAN is superior to the others, both in terms of bandwidth and maximum attenuation. Other measurements show that at the same time the power consumption of the WaveLAN is also the lowest by far. Therefore we will only consider the WaveLAN implementation in the remainder of this paper.

Note that the bandwidth provided by WaveLAN ($\geq$ 5 Mbps) is much larger than needed for streaming compressed audio ($\leq$ 705 Kbps). Fortunately, the 802.11 standard defines a method to periodically power down WLAN cards. A WLAN card informs the base station of its decision to power down and states an interval period. A typical interval period lies between 100 and 1000 ms. The base station will buffer all packets destined for the WLAN card. Periodically the base station transmits a beacon containing information about buffered packets. When the WLAN card wakes up, it listens to the beacon and requests packets destined for itself, if any. This power down method conserves power, but increases delay. For the following measurements the power down interval was set to 250 ms.

In Figure 2 the power consumption of the WaveLAN card is given for several link loads (i.e. compressed audio files in different formats) and two attenuations (i.e. short and long distance). The power consumption is measured by the voltage drop across a small and accurate resistor in the path of the power supply. The voltage drop is sampled at 20 kHz and converted into the average power consumption. We compensate for voltage drops in the power supply to the WaveLAN card.

The zero load indicates the power consumption when idle, which is 74 mW irrespective of the distance. For non-zero link loads error-free transmission over a short distance is much cheaper than error-prone transmission over a long distance. Erroneous packets are discarded, and timers within the 802.11

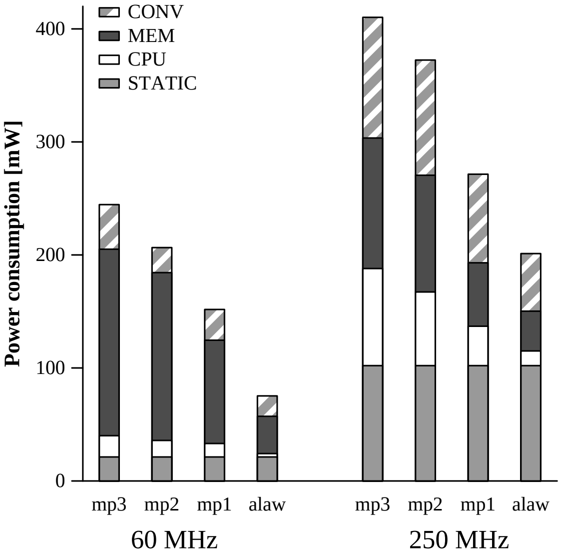

Figure 3: Costs of audio decoding on the LARTman.

MAC layer of the WaveLAN card trigger packet retransmissions. These retransmissions take time and cause the power consumption to rise significantly. For example, at 128 Kbps (MP3) the power consumption more than *doubles* when retransmissions are needed. The maximum bit rate is only about 640 Kbps with the power-down option, which is smaller than required by A-law (705 Kbps). A-law can only be supported when WaveLAN operates continuously and consumes about 1 W, irrespective of the actual link load and distance.

2.2 Audio decoding

Figure 3 shows the power consumption of the different audio decoders on the LARTman. The measurements at 60 MHz represent the case that the LARTman is used solely for decoding; the 250 MHz case represents a loaded LARTman running other applications requiring maximal CPU performance. To better understand the decoding costs, the power consumption is broken down in four components: *static*, *processor*, *memory*, and *voltage conversion*. They will be discussed in detail below.

The differences in behavior between decoding audio on an empty and a full system are caused by the exploitation of frequency and voltage scaling on the LARTman. In [12] we showed that the power consumption of the processor can be minimized effectively by matching the clock frequency to the actual workload, and lowering the voltage accordingly. The clock frequency can be varied between 60 MHz, which is already enough for audio decoding, and 250 MHz. The impact on power efficiency is quite large: an instruction at 250 MHz consumes five times as much energy as its counterpart at 60 MHz.

The *static* component in the power breakdown of Figure 3 is the fixed power consumption of the LARTman when idle, without any useful activity of the processor or memory. This value lies between 700 and 1200 mW for typical handhelds [14]. The LARTman, however, is much more power efficient: the static component lies between 21 mW (60 MHz) and 102 mW (250 MHz).

The *processor* component is the power consumed by the processor core (including cache) of the StrongARM. At 60 MHz the differences between the various audio decoders are small. At 250 MHz, however, it is clear that MP3 is the most complex decoder demanding the most processing power (86 mW), while A-law is the most simple one (13 mW).

Besides inducing a load on the processor, audio decoding also uses the memory subsystem of the LARTman quite intensively. In fact, the *memory* component is the dominating factor at 60 MHz and is even larger than at 250 MHz. The reason is that the memory access signals are derived from the processor clock, whose granularity is too coarse at low frequencies, resulting in poor memory bandwidth: 38 MB/s at 60 MHz versus 94 MB/s at 250 MHz. Nevertheless, even at 250 MHz the memory component is larger than the processor component, showing that audio decoding cost is dominated by memory access. This is somewhat unfortunate since memory operates at fixed (3.3) voltage and cannot be scaled down as with the processor.

The final component in the power breakdown of Figure 3 is the loss due to *voltage conversion*. Converting the unregulated (battery) input into stable voltages for the processor and memory costs between 10 and 25 % of the total power. It is significant at 250 MHz.

2.3 Analog audio output

The LARTman contains special add-on cards with analog circuitry to output audio signals to a headset. Two alternatives are available: *near studio* quality with minimal distortion, and *walkman* quality with a small amount of signal distortion. Generating the analog signals is in both cases rather expensive because of the conversion from the digital (decompressed audio) into the analog domain and the required amplification. Generating near studio quality requires 554 mW, walkman quality is less expensive and consumes 239 mW. It is the user who must trade-off audio quality and costs (power consumption). Since even the power consumption of low quality audio is high in comparison to decoding (MP3=223 mW), we are developing an integrated audio output system on the LARTman board itself with more efficient components. This will reduce the power consumption to below 100 mW.

3 Discussion

The basic power-consumption measurements in the previous section (wireless link, decoding, and analog output) add up to the total costs of wearable audio access. Moreover, we can determine the optimal amount of compression from the power perspective under different circumstances since we have data about multiple decoders, short- and long-range wireless communication, and processing costs on an empty/full LARTman. Figure 4 shows the costs of wireless audio streaming and decoding for three pairs of decoders and associated link loads; it shows the sum of decoding costs (Figure 3) and associated bitrates (Figure 2). The A-law decoder is not shown, because its bit rate exceeds the capacity of the WaveLAN in power-down mode. The power consumption of the WaveLAN is decomposed into a *fixed* part (74 mW) and the actual costs for handling the data traffic across the (short distance) *link*. For each audio decoder, Figure 4 shows the basic *decoding* costs at 60 MHz and the additional power required on a loaded LARTman at 250 MHz.

The best performance is obtained by the MP3 decoder, because the differences in power consumption by the wireless link dominate the trade-off; the penalty of streaming audio at a higher bit rate is not compensated by the gain in decoding costs for MP2 and MP1. The trade-off will not change when considering the power consumption for a long distance link. On the contrary, the difference between MP3 and the others will increase since the cost per transmitted bit increases sharply (see Figure 2). Thus, changes in the wireless link performance have no impact on the choice of the optimal decoding scheme. Likewise, the effect of the load on the LARTman does not affect the choice for MP3. The additional processing costs at 250 MHz for MP3 are not enough to compensate the higher costs of the wireless link incurred by MP2 and MP1.

Based on the above observations it is tempting to conclude that the best choice for a wireless playback device is to compress audio as much as possible to reduce the data rate across the link to the minimum. This, however, is not true as will be shown by two back-of-the-envelope calculations below. First, suppose that we use an audio encoder with a better compression ratio than MP3, for example, Advanced Audio Coding (AAC) operating at 84 Kbps. This would reduce the transmission cost over the (short distance) WaveLAN link with 49 mW (see Figure 2). Thus, the increase in decoding costs must be less than that. We do not have an efficient implementation of AAC on the LARTman, because AAC is dominated by floating-point operations, which are emulated in software. On a Pentium processor (with floating point support), however, AAC is five times as slow as MP3. It is very unlikely that on the LARTman AAC will consume less than 49 mW more than MP3 (221 mW); the ratio should be below 1.2. Taking the long distance numbers into account gives AAC some more slack: 168 mW instead of just 49 mW. Again, it is unlikely that AAC decoding will stay within that margin (factor 1.8).

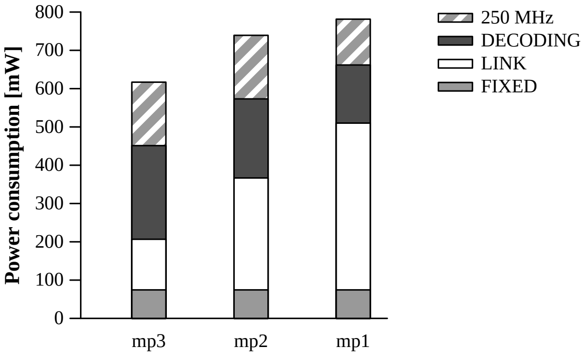

Figure 4: Costs of wireless audio streaming and decoding (excluding analog audio).

Second, suppose that we are using a wireless link with much lower costs per transmitted bit. We will use infrared as an example. With available infrared components an IrDA FIR link would consume around 22 mW when idle, and an additional 0.2 mJ/Kb, which is significantly less than the costs of WaveLAN. With IrDA the advantage of MP3 over A-law is just 124 mW (0.2×(705-84)). The difference in decoding costs, however, is 170 mW (see Figure 3) causing A-law to consume less power overall. The disadvantage of infrared is that the range is limited to about 4 meters. Switching adaptively between WaveLAN and infrared, however, combines the best of both worlds: range and low cost. This implies that the trade-offs for audio streaming change when switching between networks, requiring the system to select another decoder.

4 Related work

Power optimizations at the OS level is an area of active research. A detailed power breakdown of mobile computers is described in [9]. Most research uses the "power down when idle" principle for a single component such as the disk [5], CPU [8, 11], and wireless link [14]. We studied trade-offs between two components (decoding and wireless link).

Odyssey [3, 10] is an excellent example research project aiming at lowering the power consumption by involving the application. Power reductions are obtained by adapting the application performance (video resolution; map resolution) to the available energy through a resource negotiation interface. We can apply these concepts in our LARTman. In [1] a toolkit is described with similar objectives, however, efficiency is lost in the implementation because Java and Corba are used.

Measuring the power consumption of devices is difficult. We were able to measure the power consumption of individual hardware components directly, other researchers [4, 9] measure only the total power. Odyssey's power-measurement method [4] uses CPU profiling to break up the total power. The uncalibrated battery levels returned by the Apple and PalmPilot hardware are used in [2, 9] to obtain the power consumption.

5 Conclusions and future work

This paper addresses trade-offs involved for wireless audio access in a scenario where audio is streamed from a server on the Internet to a user carrying a wearable device equipped with a high-bandwidth wireless link. In particular, we address the trade-off between audio compression ratio (decoding costs) and wireless communication (transmission costs). We presented performance and power-consumption measurements of different Wireless LAN (WLAN) cards. WaveLAN performs best, and consumes about 74 mW when idle, and an additional 1.1 mJ/Kb when receiving data. We provided a detailed power breakdown for running MPEG1 (layers 1, 2, and 3) and A-law audio decoders on a small wearable device, called the LARTman.

By combining the decoding costs and the costs for the associated bit rate over the WaveLAN link showed that MP3 is the best choice, independent from the load on the LARTman and the condition of the link. Nevertheless, a static choice for MP3 will not yield the lowest power consumption as shown by two back-of-the-envelope calculations involving AAC and IrDA. We conclude that an adaptive system is required to make most efficient use of the limited energy that is available.

The LARTman is still under development, and we will utilize new components as soon as they become available in the future. When the PCMCIA interface becomes available, we will be able to combine the wireless link and decoding software into a single system, and perform integrated measurements. Also, an IrDA link is under construction, which will allow us to experiment with adaptive schemes that switch between networks (WaveLAN and IrDA) and decoders (MP3 and A-law).

Acknowledgements

The authors wish to thank Jan-Derk Bakker for his work on the StrongARM platform and Erik Mouw for implementing the audio drivers on this platform. Without their work this research would not have been possible. We thank Hylke van Dijk for carefully criticizing the draft version of this paper, which greatly improved it.

References

[1] Agin O., et.al., "The Mobiware toolkit: programmable support for adaptive mobile networking", IEEE Personal Communications Magazine, August 1998.

[2] Ellis C., "The case for higher-level power management", Proc. of the 7th IEEE workshop on Hot Topics in OS (HotOS), March 1999.

[3] Flinn J., Satyanarayanan M., "Energy-aware adaption for mobile applications", Proc. of the 17th Symposium on Operating Systems Principles (SOSP), 1999.

[4] Flinn J., Satyanarayanan M., "Powerscope: A tool for profiling the energy usage of mobile applications", Proc. of the 2nd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA), February 1999.

[5] Helmbold D., Long D., Sherrod B., "A dynamic disk spin-down technique for mobile computing", Proc. of the 2nd ACM International Conf. on Mobile Computing (MOBICOM), November 1996.

[6] IEEE Std 802.11, "Wireless local area networks standard", November 1997.

[7] LART homepage, http://www.lart.tudelft.nl/.

[8] Lorch J.R., Smith A.J., "Scheduling techniques for reducing processor energy use in MacOS", Wireless Networks, No. 3, 1997.

[9] Lorch J.R., "A complete picture of the energy consumption of a portable computer", Masters Thesis, University of California at Berkeley, December 1995.

[10] Noble B., "System support for mobile, adaptive applications", IEEE Personal Communications, February 2000.

[11] Pering T., Burd T., Brodersen R., "The simulation and evaluation of dynamic voltage scaling algorithms", ISLPED, August. 1998.

[12] Pouwelse J.A., Langendoen K., Sips H., "Dynamic voltage scaling on a low-power microprocessor", UbiCom-TechnicalReport/2000/4, http://www.ubicom.tudelft.nl/docs/.

[13] Sivalingam K.M., Chen J.-C., Agrawal P., "Design and analysis of low-power access protocols for wireless and mobile ATM networks", ACM/Baltzer Journal on Mobile Networking and Applications (MONET), June 1998.

[14] Stemm M., Katz R.H., "Measuring and reducing energy consumption of network interfaces in hand-held devices", IEICE Transactions on Fundamentals of Electronics, Communications, and Computer Science, 80(8):1125-1131, August 1997.