Acoustic Laptops as a research enabler*

Michael Allen
Cogent Computing
Applied Research Centre
Coventry University, UK
m.allen@coventry.ac.uk,

Lewis Girod
CSAIL
Mass. Inst. of Technology
Cambridge, MA, USA
girod@csail.mit.edu,

Deborah Estrin
CENS
UC, Los Angeles
Los Angeles, CA, USA
destrin@cs.ucla.edu,

Abstract

The Acoustic ENSBox [1] is an embedded platform which enables practical distributed acoustic sensing by providing integrated hardware and software support in a single platform. It provides a highly accurate acoustic self-calibration system which eliminates the need for manual surveying of node reference positions. In this paper, we present an Acoustic Laptop, that enables distributed acoustic research through the use of a less resource-constrained and more readily available platform. It runs exactly the same software and uses the same sensor hardware as the Acoustic ENSBox, but replaces the embedded computing platform with a standard laptop.

We describe the advantages of using the Acoustic Laptop as a rich prototyping platform for acoustic source localization and mote-class node localization applications. The Acoustic Laptop is not intended to replace the Acoustic ENSBox, but to compliment it, by providing an easily replicated prototyping platform that is extensible and resource-rich, and suitable for attended, pilot deployments. We show that the benefits gained by a laptop's extra resources enable intensive signal processing in real-time, without optimization. This enables on-line, interactive experimentation with algorithms such as Approximated Maximum Likelihood. Applications developed using the Acoustic Laptop can subsequently be run on the more deployable Acoustic ENSBox platform, unmodified apart from performance optimizations.

1 Introduction

Distributed acoustic sensing is one of the early and persistent challenges for distributed sensing. Approaches using coherent array processing, beamforming and Approximated Likelihood methods [2, 3, 4, 5] have applications in the scientific, military and commercial fields, for example tracking calls of animals or vehicles [6, 7].

However, development and deployment of systems to investigate these areas has been somewhat slowed by the lack of an integrated platform to support rapidly deployable distributed acoustic sensing. In response to this, the Acoustic Embedded Networked Sensing Box (ENSBox) [1] has been developed to enable high quality distributed acoustic sensing for these applications. It provides the software and hardware to prototype, experiment with and deploy distributed acoustic sensing applications [5, 8].

Any acoustic monitoring, detection or tracking application requires that the reference positions of observing nodes are known as accurately as possible; if these positions are poorly determined, then the estimates for the unknown acoustic sources will be adversely affected. The Acoustic ENSBox provides a sophisticated array self-calibration system that establishes precise positions and orientations: 5 cm average 2D position error and 1.5 degree average sub-array orientation error over a partially obstructed 80x50 m outdoor area [1]. In an ENSBox deployment there is no need to manually survey the positions of nodes to establish them as reference points, because the calibration system estimates the positions with high accuracy. This is critical in many environments where GPS cannot acquire lock, such as heavily forested areas.

The Acoustic ENSBox hosts a sensor board comprised of a four-microphone sub-array and speaker unit. The microphones are arranged in a tetrahedral configuration (12cm apart), and are connected to a four channel sound card that samples at rates of up to 48 KHz simultaneously on each channel. Through the Reference Broadcast Synchronization (RBS) technique [9], a network of ENSBoxes can be globally synchronized to within 10μs. This means events of interest detected by ENSBoxes can be easily correlated. These features provide good support for distributed acoustic sensing applications, as evidenced by a recent empirical study of collaborative source localization of marmots in their natural habitat [7].

While the ENSBox is resource-rich in comparison to motes, application development can still be challenging. As a compliment to the Acoustic ENSBox platform, we developed the Acoustic Laptop; a version of the Acoustic ENSBox that runs on a standard laptop, sharing the same sensor sub-array and communication hardware. The Acoustic Laptop combines all of the features that make the Acoustic ENSBox desirable (sensor array, wireless network services, time synchronization and precise self-calibration) with the advantages of a laptop: a screen, more memory and disk, a fast processor with floating point, easy integration of other peripherals via

*Support for this work was provided by NSF Cooperative Agreement CCR-0120778 and NSF Award Number CNS-0520032.

EmNets'07 June 25-26, 2007, Cork, Ireland

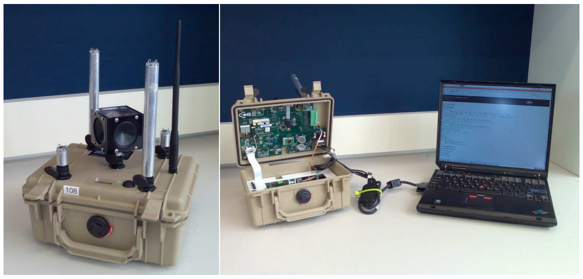

Figure 1. Left, the Acoustic Ensbox and right, the Acoustic Laptop using the same sensor array. The array is integrated into a 'Pelican' box which houses the battery compartment and optionally the Ensbox platform.

USB, broad availability and a large collection of packaged, pre-compiled software. This processing power allows on-line evaluation of intensive algorithms, enabling interactive experimentation and real-time analysis. However, the Acoustic Laptop is more cumbersome to deploy, and much less energy-efficient and weather-resistant. A laptop-based system is unlikely to last more than about 3 hours on batteries, and the keyboard, fans and vents are more likely sources of mechanical failure in harsh conditions. As such it is suitable only for development, evaluation, system planning and brief pilot deployments—but the similarity of the two systems minimizes the effort required to switch back to the ENSBox for longer deployments. The Acoustic Laptop is not intended to replace the Acoustic ENSBox, but compliment it by providing an even less constrained and readily available rapid prototyping environment. The relative ease of adoption for researchers allows exploratory investigation and analysis of distributed sensing systems without requiring a full adoption of a specfic embedded platform. Because of the framework in which the software is developed, transferring to the embedded platform is a straightforward process.

In the rest of this paper we describe our experience building and using the Acoustic Laptop. Section 2 describes how the Acoustic Laptop is implemented. Section 3 describes two example applications for which it is well-suited. Section 4 describes the results of evaluating the performance of the ENSBox relative to Acoustic Laptops with varying hardware specifications.

2 The Acoustic Laptop

Building an Acoustic Laptop entails replacing the embedded processor in the ENSBox with an external laptop. The mechanical details and analog electronics of the ENSBox remain constant, connected to the laptop by an umbilical, as in Figure 1. (all the required software is available for download, and all hardware for purchase). In this section we describe how this is accomplished.

2.1 Hardware Requirements

Any laptop with two free PCMCIA slots can be used for an Acoustic Laptop. The PCMCIA slots are used to accommodate an SMC networks 802.11b wireless card[1] and a Digigram VXPocket440 four-channel sound card[2], both of which can be be bought off-the-shelf and used 'as-is'. These two PCMCIA cards are normally plugged into the ENSBox's embedded processor board; to use the laptop as the main processor, they are instead inserted into the laptop. The rest of the system, including the analog sensor board and the microphone sub-array, is used without modification.

2.2 Software Requirements

The software on the Acoustic ENSBox that supports both self-configuration and distributed sensing is built over Linux, based on the Emstar [10] framework. The Acoustic Laptop version is a direct recompilation of the ENSBox software for the x86 architecture. Since the software is identical, laptop-based nodes can interoperate with ENSBoxes.

We chose Ubuntu Dapper Drake 6.06 LTS as the Acoustic Laptop's Linux distribution of choice, due to its ease of installation and management of software packages, which made upgrading and patching drivers relatively easy. The Acoustic ENSBox runs a 2.6.10 Linux kernel, with custom patches for the sound and network card drivers, which had to be upgraded to be compatible with Dapper Drake's 2.6.15 kernel and drivers. This was achieved without a full kernel recompilation; the relevant ALSA and HostAP drivers are recompiled as external kernel modules.

The software patches we developed specifically for the Acoustic Laptop on the 2.6.15 Linux kernel are available as part of the standard Emstar checkout, and a comprehensive tutorial is provided to ease the installation process. As the system stabilizes, we plan to construct Debian packages.

3 Applications

Many applications of distributed acoustic sensing require responses in real-time. In embedded hardware, these computationally intensive algorithms typically require heavy optimization. In this section, we describe two applications that show how the Acoustic Laptop can speed development.

3.1 Acoustic Source Localization

The Acoustic ENSBox's capability to support distributed acoustic source localization has been demonstrated in [7], where an on-line detector is used on each node to identify marmot calls and capture audio for analysis. All of this audio is consolidated to another machine, and off-line position estimates for the marmot calls are determined by identifying the combined direction of arrival (DoA) estimates from each node's audio recordings. The most likely DoA is determined using the Approximated Maximum Likelihood algorithm (AML) [2, 3, 4].

Although the processing carried out by the on-line detector is not heavily optimized when compared to a mote implementation [8], it is still optimized to fit the platform. Three-quarters of the samples taken for analysis of a detection are discarded to reduce processing load (not to the detriment of the detector's performance). An implementation of the detector on the Acoustic Laptop would not have to enforce such an optimization, and hence would be suitable for early prototyping of similar (or very different) on-line detectors.

[1] `http://www.smc.com`

[2] `http://www.digigram.com`

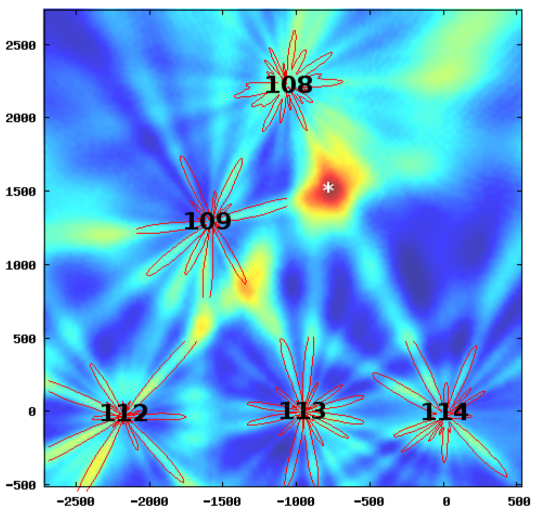

Figure 2. A pseudo-likelihood map, which is formed by 'fusing' together the results of each node's AML calculation (at each angle how likely it was the signal came from that direction). The map represents a top-down geographical view of the network (the numbers are node IDs showing each node's physical position), with a relative coordinate system in centimeters. At each node position, the AML result is overlayed as a polar plot. The estimate of the unknown source is shown as an asterisk.

The AML algorithm determines the DoA by assessing the likelihood of every possible angle from the acoustic signal could have arrived. This makes it so computationally expensive for the Acoustic ENSBox that a real-time implementation is unreasonable without heavy optimization. A laptop's processing power allows for on-line evaluation of AML, meaning the acoustic source can be analyzed in real time, and a position estimate determined on-line, enabling interactive experimentation.

We created a single-laptop AML visualizer by linking the on-line event detector to our implementation of the AML algorithm. The result of a detection and AML calculation was visualized as a polar plot on the laptop, in real-time. This prototype allowed us to debug our code and validate results rapidly, due to the real-time feedback of the system. From this, we prototyped an on-line version of the acoustic source localization system described in [7], using a network of Acoustic ENSBoxes and laptops. We deployed five Acoustic ENSBox nodes running online acoustic event detectors, and an Acoustic Laptop acting as an AML-server, which was responsible for back-end processing. Upon detecting an acoustic event, each of the Acoustic ENSBoxes would send a 32Kb chunk of data back to the AML-server (via a TCP connection); this represented a window of 4096 16-bit samples from four channels at 48KHz (around 85ms). The server performed the AML calculations on behalf of the ENSBoxes, subsequently fusing all AML results together with position information to make a pseudo-likelihood map, as shown in Figure 2.

The Acoustic Laptop allowed us to protoptye this system in an unconstrained manner, which allowed for a rapid exploratory development. A more optimized system for real-time AML evaluation would use a standard ENSBox as the AML-server. This ENSBox could be determined dynamically on each detection, depending on some arbitrary metric, for instance the ENSBox for which the signal was loudest (hence most likely nearest to the acoustic event).

3.2 Mote Localization

Node localization is an important, and ongoing research effort in WSNs. Attaching positional context to motes is important, especially if these positions are to be used as reference points for an application, such as seismic monitoring [11] or acoustic signal detection [5]. However, mote-class sensor nodes have hardware constraints in terms of processing capability, memory and ranging resources, so node localization based on acoustic ranging in such a constrained environment is not a trivial problem to solve.

To this end, acoustic node localization is commonly approached in a range-only context, where acoustic time of flight (ToF) distance estimates are taken between nodes and used as input to a localization algorithm, which may simply be a lateration calculation. There have been numerous *practical* investigations of acoustic ranging on the Mica2 [12, 13], and Cricket [14, 15] platforms, although node positions are frequently calculated offline, with the exception of [15].

The Acoustic Laptop can be used to provide a rapidly deployed, distributed surveying system, where the positions of nodes can be quickly established and recorded (either separately or in collaboration with the motes) and periodically rechecked.

If a mote can be equipped to emit a characteristic acoustic signal (or chirp), it can be identified by an on-line detector running on an Acoustic Laptop (as in the previous section). These mote chirps can be treated as unknown sound sources; a network of Acoustic Laptops can therefore combine the estimated directions of arrival of the chirp (from the AML algorithm) to see where they best converge to estimate the position of the mote.

Because the Acoustic Laptop network has tight time synchronization, time difference of arrival (TDoA) range information can also be determined by each Acoustic Laptop. The accuracy of TDOA estimates can be increased if a mote encodes some information in its acoustic chirp, such as a peusdo noise (PN) code, as in [16]. The signal identified by the on-line detector can be correlated with a reference code generated on the Acoustic Laptop.

Both of these applications have shown the benefits which come from having an x86 class platform for developing and and prototyping applications. In the next section we demonstrate the leap in processing power that using the Acoustic Laptop can provide for prototpying.

4 Experimentation

To help demonstrate that a less-constrained platform enchances, or is well suited to, prototyping, we compared the performance of the Acoustic Laptop and Acoustic ENSBox by measuring time taken to process a ranging 'chirp' sent from one 'master' node to two other nodes. Acoustic ranging between nodes is an important part of the self-calibration process in the Acoustic ENSBox and Laptop networks— it provides both direction of arrival and time of flight estimates between nodes, which are used in the computation of a relative coordinate system. The ranging process (described in more detail in [1]) is indicative of intensive signal processing—it involves calculating the angle of arrival from the relative phase differences between the four microphones, as well as beamforming to increase the signal-to-noise ratio of the pseudo-noise 'chirp' emitted by a node.

4.1 Method

In our experiments, two separate networks were set up, one for Acoustic Laptops and one for Acoustic ENSBoxes. Each network contained three nodes: a 'master', who was responsible for generating the chirp and for time synchronization; and two other nodes which received and processed the chirp. The acoustic software uses the RBS time synchronization service [9] to accurately synchronize nodes to microsecond accuracy, which is important for correlation of sampling in both acoustic source localization and self-calibration. In order to do this, RBS requires a minimum of three nodes in the network.

In our Acoustic ENSBox network, the hardware capabilities of each node are exactly the same - each node has 64MB RAM and an Intel PXA255 400 MHz processor, but in our Acoustic Laptop network, the two nodes are each different models of IBM Thinkpad, as described below:

- IBM Thinkpad T30 with 256 MB RAM and P4 2.00 GHz processor
- IBM Thinkpad 600X with 256 MB RAM and P3 500 MHz processor

We performed two separate experiments, one for the Acoustic Laptops and one for the Acoustic ENSBoxes - in both cases, we used an Acoustic Laptop as the 'master' node (for consistency), with either two laptops or two Acoustic ENSBoxes to complete the network. The 'master' Acoustic Laptop was an IBM Thinkpad T30 with 128 MB RAM and a P4 2.00 GHz processor, set up in exactly the same way as the other Acoustic Laptops. All nodes used in the experiments were running the same version of the software (albeit compiled for different platforms).

In each of the networks, the master node sent n acoustic chirps, which the other two nodes received and processed to calculate distance estimates. We measured the time taken to process a chirp by taking a timestamp at the entry to the chirp processing function (responsible for coordinating the chirp processing), and again when it exits. To fairly compare processing times, we consider only successful range estimates made by the Acoustic Laptops and Acoustic ENSBoxes, as these constitute a full processing run. In an unsuccessful range processing iteration, only part of the calculation is carried out, which reduces the processing time. For both the Acoustic Laptops and Acoustic ENSBoxes, we compare the mean and median processing times of ten successful ranging estimates.

4.2 Discussion

The processing times shown in Table 1 were taken over 10 successful trials in each network. As expected, the Acoustic Laptops processed more quickly—the more powerful IBM T30 laptop (which is by no means top of the line by today's standards) showed a mean processing time of 0.23 seconds, and with the less powerful IBM 600x, a mean of 1.11 seconds

	Processing Time (s)		
Platform	Mean	Med.	Max
P4 2GHz, 256MB RAM	0.23	0.22	0.26
P3 500MHz, 256MB RAM	1.11	1.08	1.89
ENSBox 1	19.28	20.76	22.19
ENSBox 2	20.52	21.30	22.42

Table 1. Mean, median and maximum processing time of an acoustic chirp for different architectures.

was observed. Both of these mean processing times demonstrate near-real time processing. The Acoustic ENSBox platforms take around 20 seconds to complete their processing, at least 20 times longer than the Acoustic Laptops.

Intuitively, the difference in performance is to be expected. The results clearly show the difference in having more resources available to carry the load of the intensive signal processing, which the acoustic ranging implementation requires to achieve high accuracy. The processing times demonstrated by each Acoustic Laptop show they are suitable for on-line and real-time processing applications. For example, the Acoustic Laptop can easily be used to estimate ranges from a mobile node.

5 Conclusions and future work

In this paper, we have presented the Acoustic Laptop, a version of the Acoustic ENSBox which can run on commodity hardware, and has minimal custom hardware requirements. We see this as a desirable feature which will widely enable distributed acoustic sensing research at a relatively low cost of adoption.

We have outlined the advantages of using the Acoustic Laptop as a rich prototyping environment with respect to applications, and demonstrated that the benefits gained by extra resources enable real-time and on-line intensive signal processing. This can enable a wider variety of evaluation and research, particularly with respect to real-time processing of data. The Acoustic Laptop's strengths lie in its value as a widely available prototyping platform, as its deployability make it an unsuitable replacement for more application-optimized, embedded solutions.

5.1 Platform developments

Currently, the Acoustic Laptop requires two PCMCIA slots, a luxury which is not commonly available in many new laptops on the market today, so to promote a wider usability across laptops, we would like to relax this restriction. One way to address this is to replace the PCMCIA network card with mini-PCI, the most common interface used for internal wireless cards. Since most modern cards no longer use the Prism-2 chipset, this requires porting the modifications required for synchronization support to a different driver. The most likely candidate is the MadWIFI driver, which supports most modern 802.11G chipsets. The PCMCIA sound card is somewhat more difficult to deal with—the audio server component in the ENSBox software is tightly integrated to the VXPocket440 in order to support tight synchronization with the sample stream. A firewire or USB sound card solution might be adapted, but off-the-shelf sound cards rarely implement the tight synchronization features that we need. Achieving sample-level synchronization is likely to require considerable integration work and possibly some external hardware modifications in order to achieve sample level synchronization.

Power is also inconvenient in our prototype, because the sensor board requires a separate battery source or external power supply. We are currently developing a re-spin of the sensor board that will include a USB slave port that will allow the sensors to be powered from the laptop's USB port.

5.2 Future work

We are in the process of using the Acoustic Laptop to prototype and investigate real-time, on-line collaborative source localization, and are currently integrating an on-line version of the AML algorithm. In addition to this, we are creating a visualization tool which can show a near real-time representation of the AML results for each node in the network. Following the successful integration of an on-line AML estimator, we plan to prototype real-time acoustic node localization, using the combination of AML and TDoA described in this paper.

We encourage interested researchers who wish to adopt the Acoustic Laptop platform to visit our cvs repository[3] and wiki page[4] where instructions on how to create Acoustic Laptops, as well as the software to support distributed acoustic sensing are freely available.

6 References

[1] L. Girod, M. Lukac, V. Trifa, and D. Estrin, "The design and implementation of a self-calibrating acoustic sensing platform," in *SenSys '06: Proceedings of the 4th international conference on Embedded networked sensor systems*, 2006.

[2] J. Chen, L. Yip, J. Elson, H. Wang, D. Maniezzo, R. Hudson, K. Yao, and D. Estrin, "Coherent acoustic array processing and localization on wireless sensor networks," *Proceedings of the IEEE*, vol. 91, no. 8, pp. 1154–1162, 2003.

[3] L. Yip, K. Comanor, J. C. Chen, R. E. Hudson, K. Yao, and L. Vandenberghe, "Array processing for target doa, localization, and classification based on aml and svm algorithms in sensor networks.," in *IPSN*, pp. 269–284, 2003.

[4] P. Bergamo, S. Asgari, H. Wang, D. Maniezzo, and R. E. Hudson, "Collaborative sensor networking towards real-time acoustical beamforming in free-space and limited reverberance," *IEEE Transactions on Mobile Computing*, vol. 3, no. 3, pp. 211–224, 2004. Student Member-Len Yip and Fellow-Kung Yao and Fellow-Deborah Estrin.

[5] G. Simon, M. Maróti, Á. Lédeczi, G. Balogh, B. Kusy, A. Nádas, G. Pap, J. Sallai, and K. Frampton, "Sensor network-based countersniper system," in *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems*, (New York, NY, USA), pp. 1–12, ACM Press, 2004.

[6] H. Wang, J. Chen, A. Ali, S. Asgari, R. Hudson, K. Yao, D. Estrin, and C. Taylor, "Acoustic sensor networks for woodpecker localization," in *SPIE Conference on Advanced Signal Processing Algorithms, Architectures and Implementation*, 2005.

[7] A. Ali, T. Collier, L. Girod, K. Yao, C. E. Taylor, and D. Blumstein, "An empirical study of collaborative acoustic source localization," in *IPSN '07: Proceedings of The International Conference on Information Processing in Sensor Networks*, 2007.

[8] W. Hu, N. Bulusu, C. T. Chou, S. Jha, A. Taylor, and V. N. Tran, "A hybrid sensor network for cane-toad monitoring," in *SenSys '05: Proceedings of the 3rd international conference on Embedded networked sensor systems*, (New York, NY, USA), pp. 305–305, ACM Press, 2005.

[9] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts.," in *OSDI*, 2002.

[10] L. Girod, J. Elson, A. Cerpa, T. Stathopoulos, N. Ramanathan, and D. Estrin, "Emstar: A software environment for developing and deploying wireless sensor networks.," in *USENIX Annual Technical Conference, General Track*, pp. 283–296, 2004.

[11] K. Chintalapudi, J. Paek, O. Gnawali, T. S. Fu, K. Dantu, J. Caffrey, R. Govindan, E. Johnson, and S. Masri, "Structural damage detection and localization using netshm," in *IPSN '06: Proceedings of the fifth international conference on Information processing in sensor networks*, (New York, NY, USA), pp. 475–482, ACM Press, 2006.

[12] J. Sallai, G. Balogh, M. Maróti, Á. Lédeczi, and B. Kusy, "Acoustic ranging in resource-constrained sensor networks.," in *International Conference on Wireless Networks*, pp. 467–, 2004.

[13] Y. Kwon, K. Mechitov, S. Sundresh, W. Kim, and G. Agha, "Resilient localization for sensor networks in outdoor environments," in *ICDCS '05: Proceedings of the 25th IEEE International Conference on Distributed Computing Systems (ICDCS'05)*, (Washington, DC, USA), pp. 643–652, IEEE Computer Society, 2005.

[14] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The Cricket Location-Support System," in *6th ACM MOBICOM*, (Boston, MA), August 2000.

[15] D. Moore, J. Leonard, D. Rus, and S. Teller, "Robust distributed network localization with noisy range measurements," in *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems*, (New York, NY, USA), pp. 50–61, ACM Press, 2004.

[16] L. Girod, V. Bychkovskiy, J. Elson, and D. Estrin, "Locating tiny sensors in time and space: A case study," in *In Proceedings of the International Conference on Computer Design (ICCD 2002)*, 2002.

[3] http://cvs.cens.ucla.edu/

[4] http://www.lecs.cs.ucla.edu/wiki/index.php/AcousticLaptop