Audio Engineering Society

Convention Paper

Presented at the 118th Convention
2005 May 28–31 Barcelona, Spain

Audio-Based Self-Localization for Ubiquitous Sensor Networks

Ben C. Dalton and V. Michael Bove Jr.

MIT Media Laboratory, E15-368B, 20 Ames Street, Cambridge MA 02139-4307 USA
{bcd,vmb}@media.mit.edu

ABSTRACT

An active audio self-localization algorithm is described which is effective even for distributed sensor networks with only coarse temporal synchronization. A practical implementation of a simple method of estimating ranges from recordings of reference sounds is used. Pseudo-noise "chirps" are emitted and recorded at each of the nodes. Pair-wise distances are calculated by comparing the difference in the audio delays between the peaks measured in each recording. By removing dependence on fine grained temporal synchronization it is hoped that this technique can be used concurrently across a wide range of devices to better leverage the existing audio sensing resources that surround us. An implementation of this method using the Smart Architectural Surfaces development platform is described and assessed. The viability of the method is further demonstrated in a mixed-device *ad-hoc* sensor-network case using existing off-the-shelf technology.

1. INTRODUCTION AND MOTIVATION

We are surrounded by digital devices that can communicate, listen and learn. Many of the mobile phones, music players, laptops, cameras and personal assistants we carry are equipped with microphones, speakers and some form of wireless communication. As we sit with colleagues, family or friends, we create the potential for *ad-hoc* microphone and speaker arrays. These clusters of devices offer the resources of a distributed audio sensor network. If their owners desired, the impromptu collection of sound sensing nodes could share information, detect the sounds they each produce, and establish a common spatial distribution.

It seems strange that the computers we use do not know the physical position of people and things - something that is so natural to us. Audio affords rich contextual information of surroundings, movement and position. Sound accompanies most human activity, providing signatures that can inform emerging mobile computing systems of the spaces they occupy and the people that use them. A network of sensors benefit from their spatial distribution, but must establish a common co-ordinate system in order to use this information.

Active audio self-localization, in which nodes can both produce and sense sounds as a means of measuring the distance over which the sounds travel, has been demonstrated to be an increasingly robust approach for calibration, tracking and mapping in a number of

dedicated hardware systems (see Bachrach *et al.*[1] for an overview of recent work). Here a simple ranging method[2] is used to extends these localization approaches to a disparate system of sensors in which fine-grained synchronization, speaker performance and microphone calibration cannot be relied upon. Presented in this paper is a description of a practical implementation of this technique and an analysis of the spatial co-ordinates derived from the measurements of inter-node time of flight (ToF).

With advances in mobile computing and the adoption of several effective communication protocols, and as development environments for consumer electronics open up, personal devices are becoming potent components in future distributed ubiquitous systems. It is only recently that the possibility of forming sensor networks across multiple platforms has been discussed (for example[3]), and yet the ability of a distributed sensing system to make use of unanticipated resources as they become available in an environment is an appealing one. Consider a scenario in which several sensor-network-ready devices happen to be within range, phones placed on a table during a meeting for example, or the devices scattered throughout a home. They discover and establish contact with one another through group forming protocol and then use a distributed audio localization technique to establish their own locations. This information is invaluable in developing spatial awareness in these devices and provides a basis for beam forming, sound tracking and speaker array algorithms. By requiring only a microphone, speaker and some form of inter-node communication, but no access to low-level hardware, it becomes possible to utilize any device with these basic resources as part of an auditory scene analysis sensor network. A microphone array can then be built on the fly from these pervasive components with little infrastructure.

2. BACKGROUND AND RELATED WORK

Ranging between sensors provides one component of robust multi-modal location-sensing frameworks that benefit from a combination of the properties, and difference in failure modes, of triangulation, proximity sensing and scene analysis techniques[4]. Audio is well suited to room range measurements because the speed of sound is slow enough for unspecialized hardware to measure, with sufficient accuracy, the time of flight at these distances. Long sound files can be stored easily and processed with relative computational simplicity. Architecturally, acoustics tend to be restricted to single rooms and spaces, leading to a very human scale to sound sensing. Sound is physically easy to produce and detect, and features prominently as a core interface, both input and output, in many devices. Further to using the linear relation of distance and time of flight to determine spatial information, the angle of a sound's arrival and the reduction in amplitude, that falls off with the radius from the source squared as the pressure wave spreads, can also be used. However, angle must be measured with specialized directional sensors, and volume is difficult to calibrate between devices[5]. The performance of any sound sensing approach depends on environmental changes to the speed of sound, physical occlusion, background noise and multi-path reverberation[6]. Human factors must also be considered; audible chirps can be distracting and pollute the sound environment. High frequency noise triggers a natural reflex that draws our attention. Madhavapeddy *et al.*[7] have investigated using tonal qualities to better encode and transmit information via sound in the presence of humans.

Sound flight time measurement has matured in the fields of location and imaging using echo return time, such as medical ultrasound and marine sonar, and position estimate and tracking of unknown sound sources, such as passive sonar and ballistics location. For the cases in which signals can be produced or detected at both ends of a line to be measured, the task is greatly simplified. The challenge with this approach is one of timing. Sensor equipped devices must establish some common time frame or comparison model in order to calculate time of flight between emission and detection. One common approach is to estimate unknown timing jitter in the communication between devices in order to synchronize. Another approach is to use precise emission and arrival times of an electromagnetic reference pulse, emitted synchronously with the sound pulse. As the flight time at the speed of light can be considered instantaneous over the ranges for which audio can be detected, the difference in arrival times yields the acoustic ToF. This is a popular choice for low resource systems, in which memory for storing sound and processing is limited, but access to low-level hardware timing allows exact arrival and emission times to be measured. Examples of self-localization systems using dedicated *ad-hoc* wireless sensor network hardware layers such as the Mica motes or Crickets have demonstrated 1-2 cm accuracy with this approach, and robust performance in real-room[8], urban[9] and natural[10] environments. These systems use either

audible or ultrasound pulses. Another approach is to determine ranges from a number of sensors to a reference beacon, and establish a common co-ordinate system using lateration[11]. Bachrach *et al.*[12] perform Simultaneous Localization And Tracking (SLAT) with a single moving reference node.

In cases of absolute timing comparison, synchronization must be established between nodes. Network Timing Protocol[13] has proven effective in networks; using jitter estimation, redundancy and averaging to set and maintain synchronization with a reference server. This method, however, guarantees only on the order of a hundred millisecond accuracy at points of high network traffic. To improve on this, one successful approach is to use a reference Radio Frequency (RF) pulse (for example[14]). Comparing the difference in arrival time at each node establishes a common time frame. This method generally requires low-level access to time-stamp the incoming signals in order to overcome variation in delays due to buffering in the communications layer. While this is convenient in specialized hardware and operating systems, it requires a solution specific to each processor, and so does not migrate easily from one platform to the next.

Raykar *et al.*[2] have presented an approach to positioning calibration with goals and constraints close to those described here. Their work establishes separate co-ordinates of both the microphones and speakers for a collection of heterogeneous devices, which they term General Purpose Computers (GPCs). They introduce the technique used in this paper; a closed-form position estimation of sensor-emitter pairs, using it as a close initial guess in an iterative nonlinear least squares minimization of position error functions, accounting for the synchronization offset in each measurement. The method is demonstrated in a 5 laptop, two-dimensional case. The current work does not apply this optimization step, assuming that the separation of the microphone and speaker at each device, and the uncertainty introduced by this, is small.

3. ALGORITHM AND METHOD

The approach presented here is intended for devices with sufficient computational and storage resources to record and process several seconds of sound. Full duplex audio is required for simultaneous input and output of sound, allowing a device to record its own emitted sounds. Some form of communication protocol is necessary; 802.11* WiFi, short range Bluetooth, mobile telephone protocols such as GSM (Global System for Mobile Communications), or even infrared are all suitable candidates, as are combinations of these with multi-modal devices to act as intermediaries. It is even conceivable, although inefficient, to use an entirely audio based system[7].

Unique reference signals for each of the nodes are generated by calculating pseudo-random sequences. These are phase-shift encoded onto a carrier frequency and emitted as noise-like chirps. Maximal-length sequences are used because of high auto-correlation and bounded cross-correlation properties[15]. Using 511 bit length sequences yields 48 unique identifiers, which are assigned sequentially to the nodes. This assumes that initial device discovery and group forming steps have established the available resources in an environment and given a count of the nodes, assigning an arbitrary sequential ordering to the devices. A synchronization technique such as Network Timing Protocol (NTP) is then used to establish initial coarse timing agreement allowing each device to chirp in order without overlap. It is also possible to use other unique information (for example least significant digits of IP address in a local subnet) to establish chirp order. It is desirable to have as accurate synchronization as possible, to reduce the overall recording buffer length. However discrepancies in synchronization can be tolerated with sufficient recording memory. The limiting assumption, however, is that any drift in synchronization between the clocks in each device, over the length of the recording, must be negligible. With a long recording, this may not be the case. The orthogonal chirp sequences can be detected even when two collide, further reducing the need for exact synchronization.

A single recording of all of the sequentially arriving chirps is made on each device to remove the significance of any audio circuitry buffer latency that can otherwise cause an unknown delay between the arrival and detection of a sound. By correlating the recordings against each of the known pseudo-random sequences, strong peaks are identified at the arrival of each of the sounds. Each device first finds the chirp it emitted, this significantly louder part of the audio is then suppressed before searching for the other chirps. Following the approach of Girod *et al.*[6], a running threshold, found by correlating the recording with a noise-like sequence first, suppresses false peaks, despite varying background noise levels, and so improves robustness. The most direct path for an arriving sound is found in most cases, as the first peak is generally

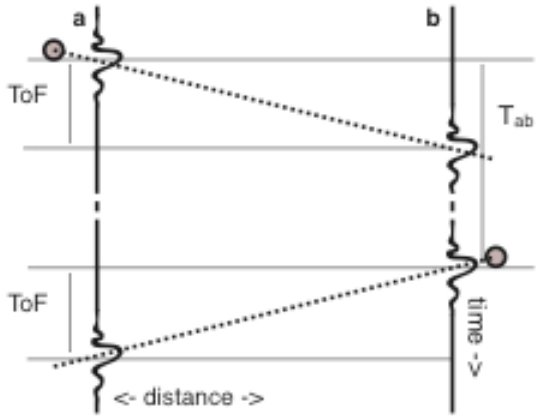

Figure 1 Emission and detection of chirps recorded at each node. Device *a* chirps, and some arbitrary time T_{ab} later, device *b* chirps. The difference in inter-peak delay between the two nodes yields twice the time of flight.

detected even in reverberant and noisy real-room conditions.

The time delay, in samples, is measured between the chirp emitted at a node and each of the remaining peaks detected in the recording. As Figure 1 shows, for two nodes *a* and *b*, the time delay between peaks measured at the device that chirps first (labeled *a*) is the arbitrary time T_{ab} between when the two chirps actually occurred, plus the time of flight, ToF, between *a* and *b*, $A=T_{ab}+ToF$. While the delay recorded at *b* is T_{ab} minus the ToF from *a* to *b*, $B=T_{ab}-ToF$. Clearly, calculating the difference between the peak-to-peak delays measured at *a* and *b* therefore yields twice the time of flight, $A-B=T_{ab}-T_{ab}+2ToF$, thus essentially removing the arbitrary time between the two auditory events from the calculation without any knowledge of the absolute synchronization of the two recordings. The error in this measurement depends on the uncertainty in the position of each of the four peaks detected and any drift between the clocks at the two nodes. All measurements are passed to a single device to iteratively calculate likely relative positions. A multidimensional scaling approach[16] is used to determine a likely three-dimensional conformation from the Euclidean range information between each pair of nodes in the graph.

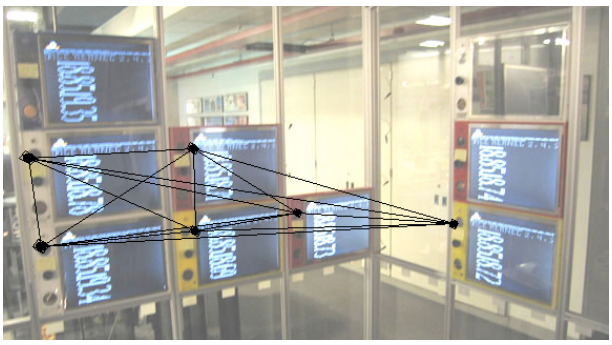

Figure 2 Smart Architectural Surfaces experimental set–up. 3D model generated from the audio range measurements translated and rotated over sensor layout (3.3cm average error).

4. IMPLEMENTATION AND RESULTS

The Smart Architectural Surface[17] is a distributed sensor network test bed consisting of many large, screen-sized, "tiles" lining the walls of a "smart" room. Each component is intended to be an easily replaceable node in a large sensor network, thus gaining robustness from the redundancy of detectors and actuators, and distributed processing in a room setting. The nodes consist of sensor-equipped XScale processors running Linux 2.4.27, and include a microphone, speaker and 802.11b wireless connection as well as video, ultrasound, temperature and humidity input. The internal hardware is similar to that in a PocketPC, and algorithms developed on the tiles are expected to be applicable to commodity handheld devices. The ranging algorithm has been initially implemented and tested on this system. It is written in C++ using the RtAudio[18] cross-platform audio API (application programming interface).

0	22	38	47	91	127
22	0	46	36	87	125
38	46	0	26	52	90
47	36	26	0	41	86
91	87	52	41	0	57
127	125	90	86	57	0

Table I Calculated pair-wise range measurements (cm).

Figure 3 shows the results for ranging measurements over the length of a room. The two devices were moved apart in the plane perpendicular to the direction the microphone and speaker are mounted. The pseudo-random sequences are phase shift encoded onto a 11025 Hz bit rate carrier wave, and are emitted and detected at 44100 samples per second. One sample corresponds to a distance of 8 mm. The background sound level in the room was measured giving a signal to noise ratio of 16 dB. The ranging has an uncertainty of ~3 cm. The peak finding performs well despite a stronger reflection than the line of sight signal, from a flat surface parallel to the plane in which the tiles lie.

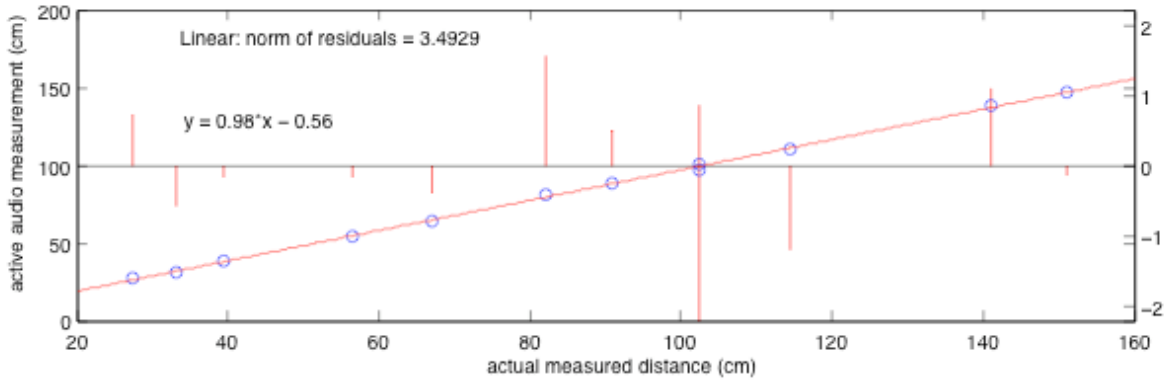

Figure 3 Ranging performance and errors between two SAS tiles facing perpendicularly to the axis of separation.

Using these SAS tiles, a 6 device system, occupying two walls, ran the self-localizing algorithm to establish the position of each device relative to the others. Table I shows the diagonally symmetrical range measurements calculated taking the speed of sound, in air, at sea level to equal 340.29 ms^{-1}. Figure 2 shows the experimental set-up overlaid with the sensor layout determined from the range measurements. Neither the grid layout nor planes of position were used as constraints. It was found that even the furthest nodes were detected despite the noise and reverberation levels in this real-room case, leading to a fully connected graph. The average distance between actual and estimated positions was 3.3 cm.

Although the code has not been fully migrated to a number of small device programming platforms, it has been possible to demonstrate its effectiveness across a range of hardware. As proof of concept, and for brevity, the devices are used to record the signals, but the 44 kHz chirp is generated and played from a co-located SAS speaker. A test set of devices consisted of an HP iPAQ running Familiar Linux, a Dell Axim running Windows Mobile and a Nokia Symbian Series phone, plus two SAS tiles. These devices span a range of microphone performance levels, sound circuit jitter and delays, and compression algorithms. Both the Axim and the Nokia provide basic recording programs that sample at a low 8 kHz rate. Figure 4 shows the performance of cross-correlation peak detection on the 8 kHz Axim recording, correctly detecting the first chirp. The phone also uses the AMR, adaptive multi-rate, lossy speech compression file type. These factors clearly reduce the performance of the peak detection algorithm, and reduce the sample resolution, however, peaks can still often be found over a reduced sensing range. Figure 5 shows the positions of the devices and the calculated estimates. Input compression degrades the detection of chirps, but seems to only contribute a small error to measurements in cases when peaks are found. Figure 6 shows a comparison of tile only and tile-phone ranging.

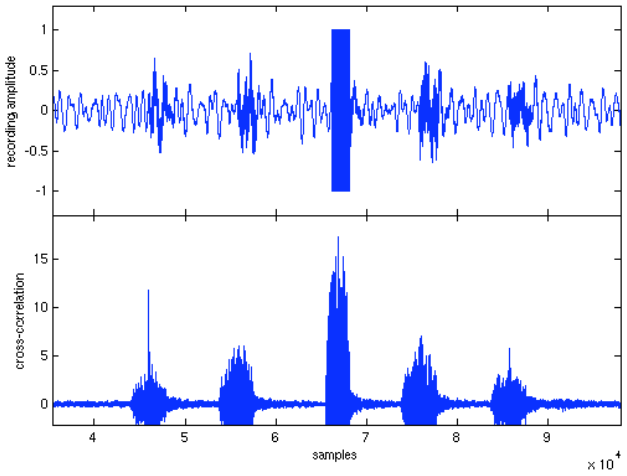

Figure 4 Upper plot shows raw 8 kHz recording up-sampled to 44 kHz. The central block is the co-located chirp. Lower plot shows correlation for the first chirp and corresponding peak.

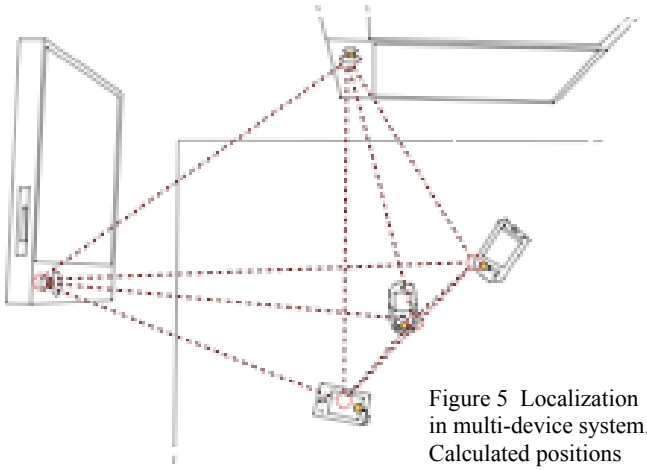

Figure 5 Localization in multi-device system. Calculated positions (dots) overlaid on actual device location.

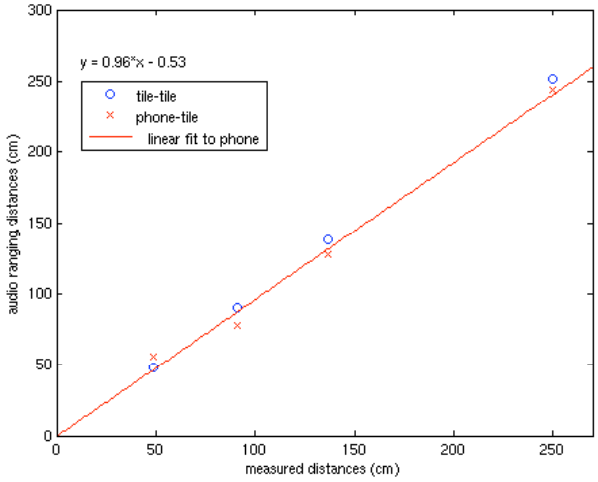

Figure 6 A comparison of ranging using 44 kHz recording on the SAS platform and 8 kHz, lossy recording on a phone.

5. DISCUSSION

A simple method for finding pair-wise range measurements between only coarsely synchronized devices, equipped with microphones and speakers, has been described and demonstrated. Correcting for the effect on the speed of sound of environmental conditions, error in the measurements is close to the uncertainty in the sampling rates used. For 44100 Hz samples in the tests described, this is approximately 3 cm. A lateral sensing case is presented, in which sensors and emitters are directed perpendicularly to their plane of separation. Pseudo-noise chirps are calculated and correlation peaks found in the recordings at each of the nodes, yielding calculations of ranges. These can be passed to a number of existing algorithms[2,8,9,12] to accurately establishing a common co-ordinate system across the sensors in an *ad-hoc* system. It is shown that using a multidimensional scaling approach, the sensor network positions can be retrieved up to global rotation and reflection of the system.

Peak finding on the recordings occurs at each node. The benefit of this is that only a small amount of information, the length of the delays between peaks, has to be passed over the communication channel. The current results were taken using a correlation algorithm that runs over the entire length of the recording and thus proportionally scales the processing time with the number of nodes. Running an initial peak-finding step on a down-sampled version of the recording to identify areas of interest first may increase the speed of the algorithm. As the number of nodes in a densely populated system grows, the approach can be modified so that each sensor looks for peaks in a constrained time window, such as the time for 6 chirps to have occurred before and after the nodes own chirp. Multidimensional scaling methods that improve performance in networks containing a large number of nodes with a degree of connectivity on the order of 10 neighbors or more, such as Moore *et al*[8] which avoids settling on false minima in possible conformation, can be used in this case.

The Smart Architectural Surfaces platform has been used in these tests, providing distributed processor and sensor resources, and group-forming and communications protocol. The algorithms have been developed to function, with little modification, over a large range of electronic devices. This is demonstrated in a multi-device, real-room, test case. Despite degradation in peak finding performance due to compression of the audio recordings, accurate range measurements are still obtained with outlier rejection and careful tuning of the threshold levels. Optimization of the code to use fixed-point only calculations, which improve speed on the limited resources of processors on the SAS, phones and similar devices, is currently work in progress. The authors are also looking to use the temperature and humidity sensors available in the test platform as a possible way to correct for global variations in sound speed, however it has been observed[6] that local variations, such as air outlets between nodes can have a significant effect in sound based ranging approaches. Statistical analysis of repeat measurements and calibration for a known separation between a sensor pair may provide effective scaling in future tests on generic sensor networks.

6. ACKNOWLEDGEMENTS

This work was supported by the CELab, Digital Life, and Things that Think research consortia, and a joint research program with the Information and Communications University.

7. REFERENCES

[1] Bachrach J. and Taylor C. "Localization in Sensor Networks." Handbook of Sensor Networks, in press Wiley 2005.

[2] Raykar V. C., Kozintsev I. and Lienhart R. "Position Calibration of Microphones and

Loudspeakers in Distributed Computing Platforms." IEEE Transactions on Speech and Audio Processing, 2003.

[3] Lienhart R. and Kozintsev I. "Self-aware Distributed AV Sensor and Actuator Networks for Improved Media Adaptation." ICME2004, Taiwan, 2004 June.

[4] Hightower J. and Borriello G. "Location Systems for Ubiquitous Computing," Computer, vol. 34, no. 8, pp. 57-66, IEEE Computer Society Press, 2001. Aug.

[5] Whitehouse K. and Culler. D. "Macro-calibration in Sensor/Actuator Networks." Mobile Networks and Applications Journal (MONET), Special Issue on Wireless Sensor Networks. 2003. June.

[6] Girod L. and Estrin. D. "Robust range estimation using acoustic and multimodal sensing". Intelligent Robots and Systems, 2001 Volume: 3, 29 Oct.-3 Nov.

[7] Madhavapeddy A., Scott D., Sharp R. "Context-Aware Computing With Sound." International Conference on Ubiquitous Computing, 2003. October.

[8] Moore D., Leonard J., Rus D. and Teller S. "Robust Distributed Network Localization with Noisy Range Measurements" ACM Conference on Embedded Networked Sensor Systems (SenSys '04), 2004. November 3-5.

[9] Gyula S., Akos L., Miklos M., *et al.* "Sensor Network-Based Countersniper System", ACM SenSys 2004

[10] Wang H., Estrin D. and Girod L. "Preprocessing in a Tiered Sensor Network for Habitat Monitoring" EURASIP JASP Special Issue on Sensor Networks, pp. 392-401, 2003. May.

[11] Broxton M., Lifton J. and Paradiso J. "Localizing a Sensor Network via Collaborative Processing of Global Stimuli" EWSN 2005.

[12] Taylor C., Rahimi A., Bachrach J., and Shrobe H. "Simultaneous Localization and Tracking in an Ad Hoc Sensor Network" IPSN 2005.

[13] Mills D. L. "Internet Time Synchronization: The Network Time Protocol." IEEE Transactions on Communications 39 no. 10, p. 1482- 1493, 1991. October.

[14] Elson J. E., Girod L., and Estrin D. "Fine-Grained Network Time Synchronization using Reference Broadcasts." The Fifth Symposium on Operating Systems Design and Implementation, p. 147-163, 2002. December.

[15] Sarwate D. V. and Pursley M.B. "Crosscorrelation Properties of Pseudorandom and Related Sequences" Proc. of the IEEE vol.68, no 5. 1980.

[16] Ji X. and Zha H. "Sensor positioning in wireless ad-hoc sensor networks with multidimensional scaling" Proceedings of IEEE INFOCOM 2004.

[17] Bove, Jr. V. M. and Mallett J. "Collaborative Knowledge Building by Smart Sensors." BT Technology Journal, 22:4, Oct. 2004.

[18] RtAudio, www.music.mcgill.ca/~gary/rtaudio/