DSP IMPLEMENTATION OF A DISTRIBUTED ACOUSTICAL BEAMFORMER ON A WIRELESS SENSOR PLATFORM[1]

J.C. Chen ♯, *L. Yip* ♯, *H. Wang* †, *D. Maniezzo* ♯, *R.E. Hudson* ♯, *J. Elson*†, *K. Yao*♯, *D. Estrin*†

† *Computer Science Department, UCLA*, ♯ *Electrical Engineering Department, UCLA*
Los Angeles, CA 900095-1594, USA
jcchen@ee.ucla.edu; hbwang@lecs.cs.ucla.edu; leyip@ucla.edu; dmaniezzo@ieee.org;
ralph@ucla.edu; jelson@cs.ucla.edu; yao@ee.ucla.edu; destrin@cs.ucla.edu

ABSTRACT

In this paper, we consider the use of Compaq iPAQ 3760s, equipped with a built-in microphone and an external wireless card, for acoustic acquisition and processing to perform a distributed acoustical beamforming. Time synchronization among the microphones is achieved by the Reference-Broadcast Synchronization method. Two beamforming algorithms, based on the time difference of arrivals (TDOAs) among the microphones followed by a least-squares estimation, and the maximum-likelihood (ML) parameter estimation method, are used to perform source detection, enhancement, localization, delay-steered beamforming, and direction-of-arrival estimation. Experimental beamforming results using the iPAQs and the wireless network are reported.

1. INTRODUCTION

Recent developments in integrated circuit technology have allowed the construction of low-cost small sensor nodes with signal processing and wireless communication capabilities that can form distributed wireless sensor network systems. These systems can be used in diverse military, industrial, scientific, office, and home applications [1]-[2]. In this paper, we propose to perform beamforming based on coherent processing of acoustical waveforms collected from the sensor nodes for detection, localization, tracking, identification, and signal-to-noise-ratio (SNR) enhancement of acoustical sources, counting the number of such sources, and estimation of the impulse responses of the acoustical channels. In order to perform coherent processing of these waveforms, the signals collected from the nodes must be time-synchronized with respect to each other.

In the past, most reported systems performing these beamforming operations involved custom-made hardware. In the paper, we propose to use Compaq iPAQ 3760s, which are handheld, battery-powered devices normally meant to be used as PDAs. We selected the iPAQ because it is small, has reasonable battery life, supports Linux OS, and is readily available commercial-off-the-shelf (COTS). Synchronization among the iPAQ's CPU clocks is achieved using Reference-Broadcast Synchronization (RBS), described by Elson et al in [8]. Two of our previously proposed and verified beamforming algorithms [4]-[6] can be implemented on the iPAQ based sensor network. The first class of beamforming algorithms exploits the time difference of arrivals (TDOAs) among the sensors, while the second class of beamforming algorithms uses maximum-likelihood (ML) parameter estimation method to perform source localization for near-field scenarios and source direction of arrival (DOA) for far-field scenarios. TDOAs approach not only collects the maximum power of the dominant source, but also provides some rejection of other interferences and noises. The relative phase information among the weights yields the relative propagation time delays from the dominant source to the sensors. In the ML approach, several sub-arrays yielding cross bearing DOAs can also be used to perform accurate source localization.

We compare the operations of these beamforming algorithms on the wireless sensor network. Complexity and performance of these algorithms vary greatly depending on the algorithms, type of sources, the geometric relationships among the sources and the sensor nodes, the signal strengths of the sources, and other system parameters including time-synchronization errors in the sensor nodes, sampling rate of the iPAQ, etc.

2. TEST BED DESCRIPTION

The test bed is a wireless sensor network that acquires acoustic source signals capable of performing various signal, array, and sensor networking experiments. In this paper, we intentionally process data offline in order to study the effects of source bandwidth, ambient noise, and array config-

---

[1] This work is partially supported by the NSF CENS program, Intel Corp., AFRL grant N374, and NASA-Dryden grant NCC2-374.

uration on the performance of our beamforming algorithms; but real-time operation for practical applications is of eventual interest.

2.1. Hardware Platform Description

We selected COMPAQ iPAQ H3760 Pocket PC as the test bed node because it has integrated sensing, processing, and communication capabilities. It has a built-in microphone for recording acoustic signals. Its codec supports a sampling rate ranging from 8 kHz to 48 kHz and a sample format of signed 16-bit integer. Its 206 MHz StrongARM-1110 CPU, 32 MB ROM and 64 MB RAM provide reasonable computational resource for digital signal processing. We also equipped each node with an 11-Mbps ORiNOCO Silver PC Card that implements IEEE 802.11b. In addition, we chose Linux operating system [7] for the test bed node because the open source nature makes it convenient for development.

2.2. Time Synchronization

Beamforming requires synchronized sensor nodes. In the test bed, fine-grained time synchronization is realized by an implementation of Reference Broadcast Synchronization (RBS). Briefly, RBS synchronizes a set of receivers of reference broadcast with one another, in contrast to traditional time synchronization protocol in which a receiver synchronizes with a sender; RFS removes the nondeterministic round-trip delay from the critical path and achieves significantly more precise synchronization [8].

On each node in the test bed, there is a RBS demon that periodically, broadcasts a reference packet with a sequence number, ID. It also listens for arrival of such reference packets from other RBS daemons. Whenever a RBS demon receives a reference packet, it reports the arrival time-stamp along with its ID back to the reference packet sender. The reference packet sender (RPS) collects all reception reports and computes clock conversion parameters between each pair of nodes. The RPS broadcasts back the computed parameters to reference packet receivers that will make them available to users by providing an appropriate library function. In practice, iPAQ-to-iPAQ time synchronization via 802.11b has an error of about 1.5 μs, which is less than the transit time over a distance of 0.02 inches, and is adequate for our intended coherent beamforming operation.

2.3. Data Collection

In the testbed, nodes are organized into clusters. The cluster head commands other nodes to collect the same number of acoustic data samples starting from the same time. Due the low-cost consumer-grade audio codecs on iPAQ 3760s we measured a large nondeterministic latency when they are requested to start recording even if all sensor nodes's CPU are perfectly synchronized; moreover those codecs sample acoustic signals at slightly different rates across different nodes although they are set to the same sampling rate. We avoid the first problem by using "audio server", a demon that runs the audio codec continuously for recording, time-stamps and buffers the most recent 10 s of audio data, and makes it available to user applications through a library function [9]. The cluster head sends to each sensor node an audio data request along with the specified sample numbers and the specified starting time in term of the sensor node's local time. The sensor node grabs from the audio server the specified number of audio data samples and sends them back to the cluster head. In this way, the cluster head collects from all sensor nodes the same number of audio data samples starting from the same time. The request and data transfer between cluster head and sensor nodes are realized by a client-server model. Each sensor node is configured as a server that is waiting for requests from the cluster head. The cluster head creates one client thread for each sensor node requested for audio data. All client threads run concurrently on the cluster head, thus the request and data transfer between the cluster head and each sensor node proceeds concurrently and independently improving the efficiency.

2.4. Beamforming method for source localization and DOA estimation

The collected acoustic data was processed offline by our previous proposed beamforming methods. The first beamforming method, (TDOA), involves a relative time delay estimation step then followed by a least square (LS) fit to the source location in the near-field case or the source DOA in the far-field case. The TDOA can be estimated by using the conventional correlation operation among sensors or a blind beamforming method proposed in [4].

In contrast to the TDOA-LS method, where the data is processed in the time domain, the approximated Maximum Likelihood (AML) estimator [6] does the processing in the frequency domain. Due to the broadband nature of the signal, the ML metric results in a coherent combination of each subband.

3. RESULTS OF OUTDOOR EXPERIMENT

We conducted several outdoor experiment with different sources as well as different array configurations to demonstrate the effectiveness of the proposed wireless time synchronized COTS sensor network for beamforming applications. We consider two scenarios: direct source localization for near-field and DOA estimation of the source for far-field. In the near-field case, we assume high coherency of the received signal at all sensors and apply either TDOA-CLS or AML source localization to find the source location. In the far-field case, we perform TDOA-CLS or AML DOA estimation to find source DOA at each subarray. Source location can then be estimated by crossing bearings obtained from these subarrays.

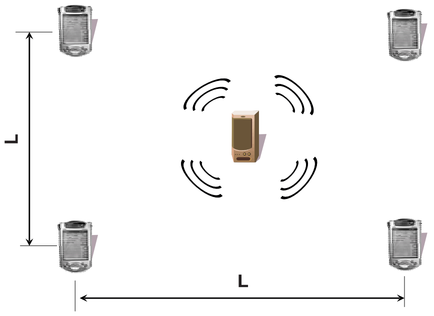

Fig. 1. Square Configuration

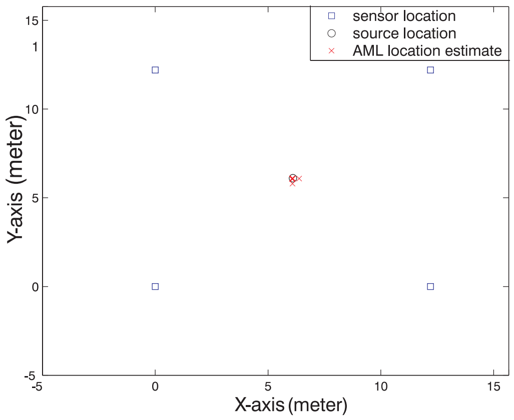

Fig. 2. AML source localization result of a music signal

Our experimental setup for the near-field source localization scenario is shown in fig. 1. The source is placed in the middle of a square sensor array. The inter-sensor spacing **L** is 40 ft (12.2 m). The sound of a prerecorded organ music with a 2 kHz bandwidth and 1.75 kHz central frequency is played through the source and collected by the microphone array embedded in the iPAQs. Fig. 2 shows that, with this configuration, the location of the source can be estimated with an RMS error of 0.0218 m using the near-field AML source localization algorithm. An error of 0.0170 m was observed for the same data using the two-step CLS method. Thus, both beamforming methods are capable of locating the source.

We consider two different subarray configurations for the far-field cross bearing source localization. In one configuration, we used a linear sensor array with three sensors as shown at fig. 3. The cross-bearings from three widely separated subarrays yield the estimated source locations, S1,..,S6, in fig. 3. The same music sound was played at six different locations simulating a source movement. The estimated results of one snapshot are shown in fig. 4 for clear illustration. It can be seen that more accurate estimation can be obtained when the source is inside the convex hull of the subarrays. Comparison of AML and TDOA-CLS can be made. The RMS errors of 0.339 m at location S5 and 1.421 m at location S1 are reported for AML method. With TDOA-CLS, the RMS error is 0.569 m at S5 and 1.57 m at S1.

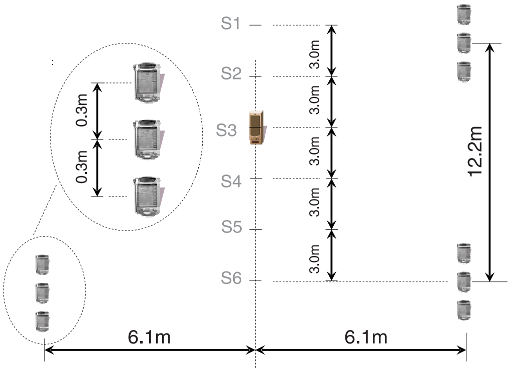

Fig. 3. Linear sensor array configuration

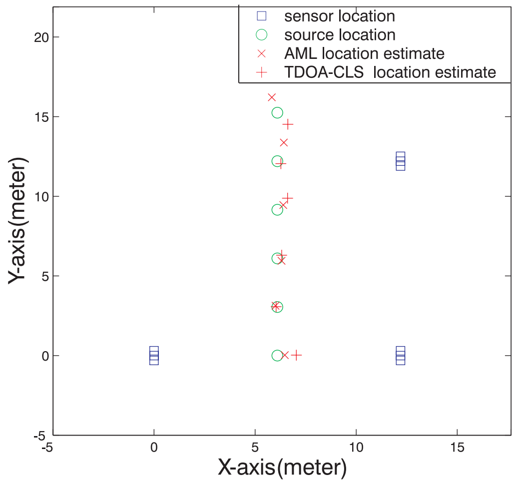

Fig. 4. Cross bearing localizations of a music source at different locations

In the configuration of fig. 5, we use four square subarrays each with four sensors. Four bearing estimates would locate two sources' locations at the same time by AML algorithm. Source 1 played a vehicle sound, while source 2 played the same music signal as before. When only source 1 was active, the AML algorithm can locate the source with RMS error of 0.281m, which is shown at fig. 6. When two sources were played simultaneously, the AML crossing bearing method can locate the source 1 with RMS error of

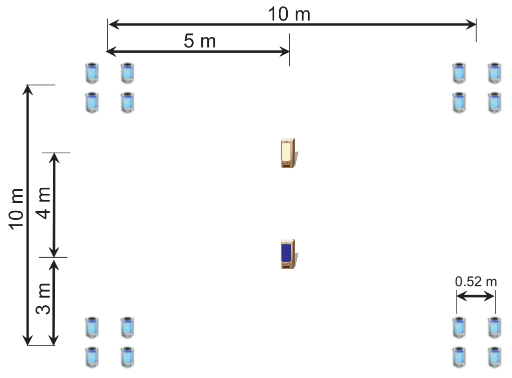

Fig. 5. Square sensor array configuration

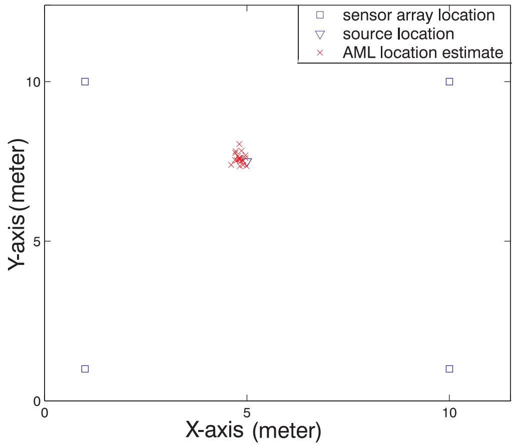

Fig. 6. Cross bearing localization of a vehicle source

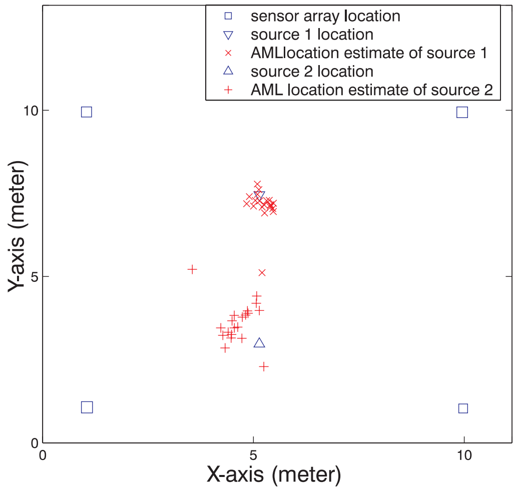

Fig. 7. Cross bearing localizations of two sources

0.431m and source 2 with RMS error of 0.837m, which are illustrated at fig. 7.

4. CONCLUSIONS

In this paper, we first describe the testbed using iPAQ 3760s to perform time synchronization and data collection. Then we demonstrate two beaforming methods for source localization and DOA estimations. These results illustrate the usefulness of this wireless testbed for performing various experimental work in signal, array, and sensor networking.

5. REFERENCES

[1] J. Agre and L. Clare, "An integrated architecture for cooperative sensing and networks", Computer, vol. 33, May 2000, pp. 106-108.

[2] G.J. Pottie and W.J. Kaiser, "Wireless integrated network sensors", Comm. ACM, vol. 43, May 2000, pp. 51-58.

[3] J.C. Chen, K. Yao, and R.E. Hudson, "Source localization and beamforming", IEEE Signal Processing Magazine, vol. 19, March 2002, pp. 30-39.

[4] K. Yao, R.E. Hudson, C.W. Reed, D. Chen, and F. Lorenzelli, "Blind beamforming on a randomly distributed sensor array system", IEEE Journal of Sel. Areas of Communication, vol. 6, Oct. 1998, pp. 1555-1567.

[5] T.L. Tung, K. Yao, C.W. Reed, R.E. Hudson, D. Chen, and J.C. Chen, "Source localization and time delay estimation using constrained least squares and best path smoothing", Proc. SPIE, vol. 3807, July 1999, pp. 220-223.

[6] J.C. Chen, R.E. Hudson, and K. Yao, "Maximum-likelihood source localization and unknown sensor location estimation for wideband signals in the near-field", IEEE Trans. on Signal Processing, vol. 50, Aug. 2002, pp. 1843-1853.

[7] The Familiar Project. `http://familiar.handhelds.org/`.

[8] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference braodcasts", Proc. of the Fifth Symposium on Operating Systems Design and Implementation, OSDI, Boston, MA, Dec. 2002.

[9] L. Girod, V. Bychkobskiy, J. Elson, and D. Estrin, "Locating tiny sensors in time and space: A case study", Proc. of ICCD 2002.