A WIRELESS TIME-SYNCHRONIZED COTS SENSOR PLATFORM: APPLICATIONS TO BEAMFORMING[1]

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

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

ABSTRACT

In recent years, sensor network system has been proposed for various applications. In the past, most reported systems involve custom-made hardware. In this paper, we consider the use of Compaq iPAQ 3760s with their build-in StrongARM processors, ROM and RAM memories, and microphones and codecs for acoustic acquisition and processing, plus external wireless Ethernet cards for radio communication to form a distributed sensor network to perform acoustical beamforming. Time synchronization among the microphones is achieved by the Reference-Broadcast Synchronization method of Elson-Estrin. Two beamforming algorithms, based on the time difference of arrivals (TDOAs) among the microphones and least-squares estimation of the TDOAs method, and the maximum-likelihood (ML) parameter estimation method, are used to perform source detection, enhancement, localization, delay-steered beamforming, and direction-of-arrival estimation. Theoretical Cramér-Rao bound analysis of the system performance and 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). Each has a built-in microphone and codec capable of sampling rate from 8 kHz to 48 kHz for acoustical acquisition. It can also support an 11 MBit/s spread-spectrum wireless Ethernet card for wireless communication. 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. A blind beamforming method uses the maximum power criterion to obtain array weights from the dominant singular vector or eigenvector associated with the largest singular value or eigenvalue of the space-time sample data or correlation matrix. This 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. Various forms of least-squares estimation methods are applied to these TDOAs to perform source detection, enhancement, localization, and delay-steered beamforming. The second class of beamforming algorithms uses maximum-likelihood (ML) parameter estimation method to perform source localization for near-field scenarios and direction of arrival (DOA) of the sources for far-field scenarios. Several sub-arrays yielding cross bearing DOAs can also be used to perform accurate source localization.

We compare the operations of these blind beamforming

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

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. These system parameters include the time-synchronization errors in the sensor nodes, sampling rate of the iPAQ, etc. We will also compare the experimentally measured results with analytically derived Cramér-Rao Bounds (CRB) characterizing theoretical optimum performance as functions of SNR, time synchronization errors, and sensor node geometry. Our goal is to make the network self organize and dynamically configure the needed sensor nodes to perform complex beamforming operations for various applications of interest.

2. TEST BED DESCRIPTION

The test bed is a wireless sensor network that acquires acoustic signals from a target for beamforming. In this paper, we intentionally process data offline in order to study the effects of source bandwidth, ambient noise and array configuration on the performance of our beamforming algorithms; but real-time operation is of eventual interest.

2.1. Hardware Platform Description

We selecteded COMPAQ iPAQ H3760 Pocket PC as the test bed node because it has integrated senseing, 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. High bandwidth communication capability is essential to coherent signal processing such as beamforming. 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), described in more detail in [8]. 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. RBS achieves significantly better precision than traditional synchronization protocol because the nondeterministic round-trip delay is removed from the critical path.

On each node in the test bed, there is a RBS daemon acting as both a sender and a receiver. Periodically, each RBS daemon 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 daemon receives a reference packet, it reports the arrival time-stamp along with its ID back to the reference packet sender. The reference packet sender collects all reception reports and computes clock conversion parameters between each pair of nodes that heard its broadcast. These parameters are then broadcast back to reference packet receivers. The RBS daemon receives these parameters make them available to users by providing a library function that converts UNIX "timevals" from one node to another. In practice, iPAQ-to-iPAQ time synchronization via 802.11b has an error of about 1.5 μs, which is much less than a sample interval at 48 kHz sampling rate.

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. However, the low-cost consumer-grade audio codecs on iPAQ 3760s cause two difficulties. First, those codecs have large nondeterministic latency when they are requested to start recording. Simply starting recording at the same time on all sensor nodes does not get audio data starting from the same time even if all sensor nodes's CPU are perfectly synchronized. Second, those codecs sample acoustic signals at slightly different rates across different nodes although they are set to the same sampling rate. As a result, audio data time series from different sensor nodes may have different duration even if they have the same number of samples and use the same sampling rate parameter.

We avoid the first problem by using "audio server" [9]. The audio server is 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. The cluster head picks a recent local time-stamp, converts it to each sensor node's local time-stamp, and then send 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 just grabs from audio server the specified number of audio data samples starting from the specified local time, then 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.

In order to overcome the second difficulty, the real duration of audio data samples on each sensor node is measured and used as input to beamforming algorithms for adjustments. With the help of audio server, it is straightfor-

ward to measure the real duration of audio data samples by checking the timestamps of the first and last samples in the audio data time series. Fig. 1 shows the difference between measured sampling durations and requested sampling durations on a set of nodes when they are asked for 240,000 samples at 48 kHz sampling rate. We conducted 16 trial runs to obtain the figure. The difference could be up to 300 μs for a 5 s of requested sampling duration. Assuming a sound speed of 340 m/s, 300 μs time difference will lead to about 10 cm distance difference. The request and data

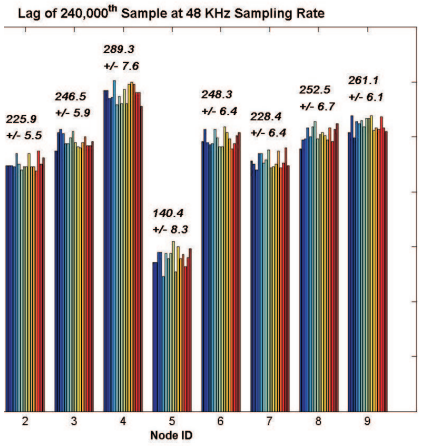

Figure 1. Measured/Requested sample duration difference

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 and returning requested audio data to the cluster head upon request. The cluster head creats one client thread for each sensor node requested for audio data. All client threads run concurrently on the cluster head, thus the reqest and data transfer between the cluster head and each sensor node proceeds concurrently and independently. Concurrency among client threads on the cluster head improves efficiency.

3. BEAMFORMING METHOD FOR SOURCE LOCALIZATION AND DOA ESTIMATION

3.1. TDOA-CLS method

The first beamforming method 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]. Without a loss of generality, we choose $p = 1$ as the reference sensor for differential time-delays. Let the reference sensor be the origin of the coordinate system for simplicity. The TDOA for P sensors satisfies:

$$t_{p1} = t_p - t_1 = \frac{\|\mathbf{r}_s - \mathbf{r}_p\| - \|\mathbf{r}_s - \mathbf{r}_p\|}{v}, \qquad (1)$$

for $p = 2, ..., P$, where $\mathbf{r}_s = [x_s, y_s]^T$ and $\mathbf{r}_p = [x_p, y_p]^T$ are the source location(s) and pth sensor location respectively in a 2-D Cartesian coordinate system, and v is the speed of propagation. This is a set of $P-1$ nonlinear equations, which can be formulated as the least squares solutions of the following $P-1$ linear equations [5]:

$$\mathbf{A}\mathbf{y} = \mathbf{b} + v^2\mathbf{c}, \qquad (2)$$

where $\mathbf{A} = \begin{bmatrix} \mathbf{r}_2^T & t_{21} \\ \mathbf{r}_3^T & t_{31} \\ \vdots & \vdots \\ \mathbf{r}_P^T & t_{P1} \end{bmatrix}$, $\mathbf{y} = \begin{bmatrix} \mathbf{r}_s \\ v\|\mathbf{r}_s\| \end{bmatrix}$,

$\mathbf{b} = \frac{1}{2}\begin{bmatrix} \|\mathbf{r}_2\|^2 \\ \|\mathbf{r}_3\|^2 \\ \vdots \\ \|\mathbf{r}_P\|^2 \end{bmatrix}$, $\mathbf{c} = \frac{1}{2}\begin{bmatrix} t_{21}^2 \\ t_{31}^2 \\ \vdots \\ t_{P1}^2 \end{bmatrix}$.

The constraint of the above equations is:

$$v^2\|\mathbf{B}\mathbf{y}\| = \|\mathbf{f}^T\mathbf{y}\|, \qquad (3)$$

where $\mathbf{B} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$, and $\mathbf{f} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$.

Define $\mathbf{g} = \mathbf{b} + v^2\mathbf{c}$, the constrained least square (CLS) solution can be obtained by Lagrangian multiplier method. The solution of the unknown vector $\mathbf{y}$ can be given by:

$$\mathbf{y} = (\mathbf{A}^T\mathbf{A} + \lambda v^2\mathbf{B} - \lambda\mathbf{f}\mathbf{f}^T)^{-1}\mathbf{A}^T\mathbf{g}. \qquad (4)$$

The Lagrangian multiplier λ can be obtained by substituting eq. (4) back to eq. (3). The resulting equation is a fourth order equation. We then select the root which gives estimate the most physical meaning.

When the source is at the far-field, the CLS solution can be derived from the following differential time delay relationship:

$$vt_{p1} = r_p(\cos\theta_s\cos\theta_p + \sin\theta_s\sin\theta_p), \qquad (5)$$

for $p = 2, ..., P$, where r_p and θ_p is the distance and angle between the pth sensor and the reference sensor respectively. The above $P-1$ equations can be formatted in the following matrix form:

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \qquad (6)$$

where $\mathbf{A} = \begin{bmatrix} r_2\cos\theta_2 & r_2\sin\theta_2 \\ r_3\cos\theta_3 & r_3\sin\theta_3 \\ \vdots & \vdots \\ r_P\cos\theta_P & r_P\sin\theta_P \end{bmatrix}$, $\mathbf{x} = \begin{bmatrix} \cos\theta_s \\ \sin\theta_s \end{bmatrix}$,

$$\mathbf{b} = \begin{bmatrix} vt_{21} \\ vt_{31} \\ \vdots \\ vt_{P1} \end{bmatrix}.$$

This is a constrained least square problem with the constraint $\|\mathbf{x}\| = 1$. The solution can be obtained by Lagrangian multiplier method. We define the function to be minimized as:

$$J(\lambda) = (\mathbf{A}\mathbf{x} - \mathbf{b})^T(\mathbf{A}\mathbf{x} - \mathbf{b}) + \lambda(\|\mathbf{x}\| - 1). \quad (7)$$

Setting the derivative of the above function to zero, we can obtain the solution of $\mathbf{x}$:

$$\mathbf{x} = (\mathbf{A}^T\mathbf{A} + \lambda\mathbf{I})^{-1}\mathbf{A}^T\mathbf{b}, \quad (8)$$

We use eq. (8) with the constraint $\|\mathbf{x}\| = 1$ to solve for λ, this yields a fourth order equation, which has at most four solutions. The solution corresponding to the more "physical" estimate that fits the data will be used.

3.2. Maximum-likelihood source localization and DOA estimation

In contrast to the TDOA-LS method, where the data is processed in the time domain, the approximated Maximum Likelihood (AML) estimator 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. In the signal model, the sensors are assumed to be omnidirectional. For simplicity, we assume both the source and sensors lie in the same plane (a 2-D scenario). For a randomly distributed array of P sensors, the data collected by the pth sensor at time n is given by:

$$x_p(n) = a_p s_o(n - t_p) + w_p(n), \quad (9)$$

for $n = 0, ..., N-1$, $p = 1, ..., P$, where a_p is the signal gain level of the source at the pth sensor, s_o is the source signal, t_p is the time-delay in samples (which is allowed to be any real-valued number), and w_p is the zero mean white Gaussian noise with variance σ^2. The time-delay is defined by $t_p = \|\mathbf{r}_s - \mathbf{r}_p\|/v$. The received wideband signal can be transformed into the frequency domain via DFT. Note the DFT creates a circular time-shift rather than the actual time-shift. Severe edge effect results from small data length N but it becomes a good approximation for large N. In the subsequent sections, we assume N is large enough and thus the noise is almost uncorrelated across the frequencies. In the subsequent sections, we denote ^{H} as the complex conjugate transpose. The array signal model in the frequency domain is given by:

$$\mathbf{X}(k) = \mathbf{d}(k)S_o(k) + \eta(k), \quad (10)$$

for $k = 0, ..., N/2$, where the array data spectrum is given by $\mathbf{X}(k) = [X_1(k), ..., X_P(k)]^T$, the steering vector is given by $\mathbf{d}(k) = [d_1(k), ..., d_P(k)]^T$, $d_p(k) = e^{-j2\pi k t_p/N}$, and $S_o(k)$ is the source spectrum. The noise spectrum vector $\eta(k)$ is zero mean complex white Gaussian distributed with variance $N\sigma^2$. The ML solution of the above signal model can be shown to maximize the following function [6]:

$$\max_{\widetilde{\mathbf{r}}_s} J(\widetilde{\mathbf{r}}_s) = \max_{\widetilde{\mathbf{r}}_s} \sum_{k=0}^{N/2} \|B(k, \widetilde{\mathbf{r}}_s)\|^2, \quad (11)$$

where $B(k, \widetilde{\mathbf{r}}_s) = \bar{\mathbf{d}}(k, \mathbf{r}_s)^H \mathbf{X}(k)$, $\bar{\mathbf{d}} = \mathbf{d}/\sqrt{\sum_{p=1}^{P} a_p^2}$ is the normalized steering vector.

Similarly, in the far-field case, the unknown parameter vector contains only DOAs. The relative time delay from pth sensor to the reference sensor is given by eq. (5). Thus, the AML DOA estimation can be obtained by $arg\max_{\theta_s} \sum_{k=0}^{N/2} \|B(k, \theta_s)\|^2$.

4. CRB ANALYSIS OF THE TIME SYNCHRONIZATION ERROR FOR DOA ESTIMATION

4.1. CRB derivation for time synchronization error

In this subsection, we evaluate the theoretical performance of DOA estimation for the far-field case by CBR analysis. CBR can provide a theoretical lower bound for any unbiased estimator. However, unbiased estimator is generally difficult to obtain in practice. Some estimators, such as ML, can be shown to asymptotically approach CRB when SNR and data length are large. From the data model of the far-field DOA case, when only the time synchronization error is considered, the received waveform of the pth sensor at the k frequency bin is given by:

$$x_{pk} = s_k \exp\left[\frac{-j2\pi k(t_p - \tau_p)}{N}\right], \quad (12)$$

where s_k is the received signal spectrum at the reference sensor, t_p is the relative time delay from the pth sensor to the reference sensor. For the far-field case, $t_p = \frac{r_p \cos(\theta_s - \theta_p)}{v}$. τ_p is the time synchronization error and assumed to be IID white Gaussian with zero mean and variance σ_r^2.
Taking ln of both sides of eq. (12), and rearranging terms, we obtain:

$$z_{pk} = f_k - \frac{r_p \cos(\theta_s - \theta_p)}{v} + \tau_p, \quad (13)$$

for $k = 1, \ldots, K$ and $p = 1, \ldots, P$, where $z_{pk} = \frac{N}{2\pi k}\Im\{\ln(x_{pk})\}$, $f_k = \frac{N}{2\pi k}\Im\{\ln(s_k)\}$, and $\Im\{\}$ represents the imaginary part of a complex value. At the frequency bin k, the P equations are stacked up to form a

complete matrix. We have the following real-valued white Gaussian data model:

$$\boldsymbol{X} = \boldsymbol{G}(\boldsymbol{\Theta}) + \boldsymbol{\tau} \tag{14}$$

where Θ is the unknown parameter that we need to estimate, $\boldsymbol{\Theta} = [\theta_s, f_k]$ in our case, i.e., the source angle and the source spectrum, $\tau = [\tau_1 \dots \tau_P]^T$ and $\mathbf{G}(\boldsymbol{\Theta}) = f_k[1 \dots 1]^T - \left[\frac{r_1 \cos(\theta_s-\theta_1)}{v} \dots \frac{r_P \cos(\theta_s-\theta_P)}{v}\right]^T$.

The CRB for the white Gaussian data model can be given by the inverse of the Fisher information matrix, which is:

$$\mathbf{F} = \frac{1}{\sigma_\tau^2}[\mathbf{H}^T\mathbf{H}], \tag{15}$$

where $\mathbf{H} = \frac{\partial \mathbf{G}(\boldsymbol{\Theta})}{\partial \boldsymbol{\Theta}}$. It can be shown that:

$$\frac{\partial \mathbf{G}(\boldsymbol{\Theta})}{\partial \boldsymbol{\Theta}} = \begin{bmatrix} \frac{r_1 \sin(\theta_s-\theta_1)}{v} & 1 \\ \vdots & \vdots \\ \frac{r_P \sin(\theta_s-\theta_P)}{v} & 1 \end{bmatrix}. \tag{16}$$

Substitute eq. (16) to eq. (15), we have:

$$\mathbf{F} = \frac{1}{\sigma_\tau^2}\begin{bmatrix} \sum_{p=1}^P \frac{r_p^2 \sin^2(\theta_s-\theta_p)}{v^2} & \sum_{p=1}^P \frac{r_p \sin(\theta_s-\theta_p)}{v} \\ \sum_{p=1}^P \frac{r_p \sin(\theta_s-\theta_p)}{v} & P \end{bmatrix}. \tag{17}$$

The CRB for the source angle can then be given by the first diagonal element of the inverse of the Fisher information matrix, which is given by:

$$\text{CRB} = \frac{\sigma_\tau^2}{\sum_{p=1}^P \frac{r_p^2 \sin^2(\theta_s-\theta_p)}{v^2} - \frac{1}{P}\left(\sum_{p=1}^P \frac{r_p \sin(\theta_s-\theta_p)}{v}\right)^2}. \tag{18}$$

Some observation can be made from the CRB formula (eq. (18)). First, the numerator of the CRB only depends on the variance of the time-synchronization error; while the denominator of the CRB depends on the array geometry and source angle. Therefore, the CRB is proportional to the time synchronization error. Furthermore, the array geometry also has effect on the CRB. Poor array geometry may lead to a smaller denominator, which results in a larger estimation variance. It is interesting to note that the geometric factor is the same as the CRB formula for additive Gaussian noise at [6], which means the array geometry produces the same effects on both kinds of errors.

Second, although the derivation is limited to one frequency bin, the resulting CRB formula is independent of that particular frequency bin. Therefore, unlike the CRB of AWGN, the CRB can not be reduced by increasing the number of frequency bins. In other words, the time synchronization error can not be reduced by increasing the data length of the received signal.

4.2. Variance lower bound for time synchronization error and AWGN

By considering time synchronization error and AWGN together, the received signal spectrum at the k frequency bin and the pth sensor, will be:

$$x_{pk} = s_k \exp\left[\frac{-j2\pi k(t_p - \tau_p)}{N}\right] + \eta_{pk}, \tag{19}$$

where the first term is the same as eq. (12), η_{pk} is the complex white Gaussian with zero mean and $N\sigma_n^2$ is the variance. Exact CRB requires the derivation of the probability density function (pdf) of the above data model, which may be a formidable task.

Here, we provide a variance bound based on the independence assumption of τ_p and η_{pk}. With this condition, the variance of the estimator will be the sum of the variance induced by these errors independently. By using $\text{var}(\theta;\tau,\eta) = \text{var}(\theta;\tau) + \text{var}(\theta;\eta)$, $\text{CRB}(\theta;\tau) \le \text{var}(\theta;\tau)$ and $\text{CRB}(\theta;\eta) \le \text{var}(\theta;\eta)$, we obtain:

$$\text{var}(\theta;\tau,\eta) \le \text{CRB}(\theta;\tau) + \text{CRB}(\theta;\eta). \tag{20}$$

The CRB induced by AWGN is given by:

$$\text{CRB} = \frac{1}{\varsigma\left[\sum_{p=1}^P \frac{r_p^2 \sin^2(\theta_s-\theta_p)}{v^2} - \frac{1}{P}\left(\sum_{p=1}^P \frac{r_p \sin(\theta_s-\theta_p)}{v}\right)^2\right]}, \tag{21}$$

where $\varsigma = \frac{2}{N\sigma^2 V^2}\sum_{k=1}^{N/2}\left(\frac{2\pi k S(k)}{N}\right)^2$ and the geometric factor is the same as the time synchronization error. Although the variance bound may not be as tight as the CRB, it can be shown to match well with the root-mean-square (RMS) error of the simulation of AML algorithm. Furthermore, it offers a much simpler and more efficient way to evaluate the variance lower bound.

5. SIMULATION EXAMPLES AND EXPERIMENTAL RESULTS

5.1. Simulation examples of time synchronization error

In this subsection, we compare the derived variance bound of the DOA estimation with the RMS error of AML in several simulations. Assume a far-field source impinges on a sensor array at $45°$. The sensor array configuration is a uniform square with four acoustic sensors, each spacing 0.345 m apart. The source is a prerecorded vehicle signal with significant spectral content of about 50 Hz bandwidth centered about a dominant frequency at 100 Hz. The sampling frequency is set to 1 kHz and the speed of propagation is 345 m/s. The data length $N \ge 2000$, which makes the edge effect of DFT insignificant. In every simulation, the RMS error of AML is computed via 100 Monte Carlo runs. In

the first simulation, we omit the AWGN and only consider the time synchronization error. The data length of the received data is set to 2000 samples. The RMS error of the AML and the CRB computed from eq. (10) are plotted as a function of the variance of the time synchronization error σ_t^2 in fig. 2. The figure shows that the CRB provides a tight

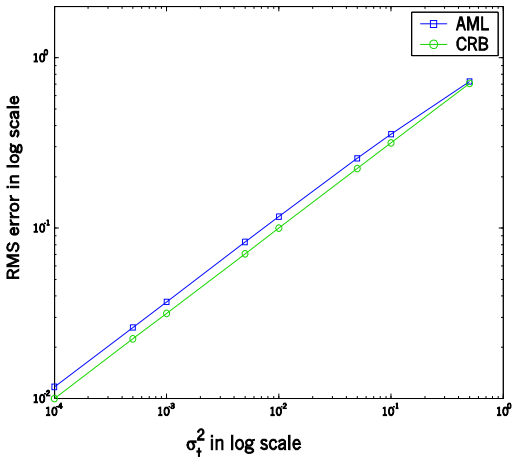

Figure 2. RMS error comparison of CRB and AML as a function of σ_t^2.

lower bound for the RMS error of the AML algorithm. Furthermore, the CRB is proportional to the σ_t^2. In the second simulation, we set $\sigma_t^2 = 0.001$ and vary N from 2000 to 4500 to perform AML DOA estimation. CRB analysis suggests that the performance of AML can not be improved by increasing the data length for fixed σ_t^2, which can be verified by the RMS error comparison between CRB and AML DOA estimation shown in fig. 3. In the third simulation, we consider the combined effect of time synchronization error and AWGN. The RMS error of AML and the derived variance bound are plotted as a function of SNR for various σ_t^2. It can be seen that the performance of AML matches well with the variance bound. Furthermore, the performance of AML is limited even at high SNR region for fix σ_t^2 at fig. 4. This is due to the time synchronization error becoming dominant at that region, which results in a error floor effect. The above theoretical as well as simulation analysis of the time synchronization error shows that it is crucial to obtain accurate time synchronization among sensors in order to yield a good performance of the coherent array signal processing.

5.2. 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. Our experimental setup for the near-field source localization scenario is shown in fig. 5. The source is placed in the middle of a square sensor array. The inter-sensor spacing **L** is 20 ft (6.1 m). The sound of a moving light wheeled vehicle is played through the speaker and collected by the microphone array embedded in the iPAQs. Fig. 6 shows that, with this configuration, the location of the

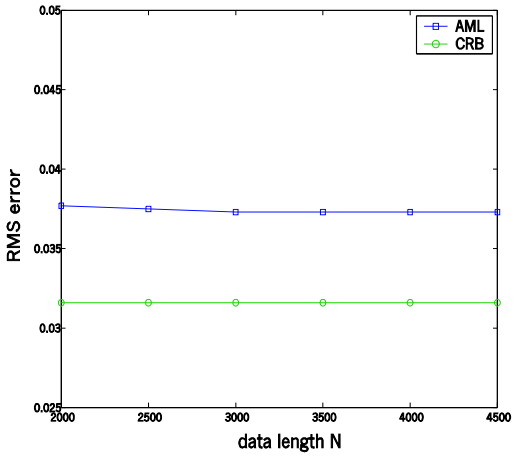

Figure 3. RMS error comparison of CRB and AML as a function of data length

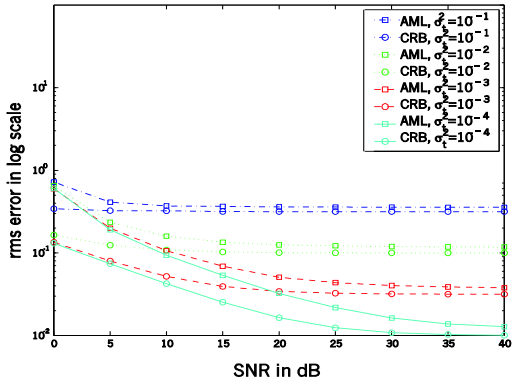

Figure 4. RMS error comparison of variance bound and AML

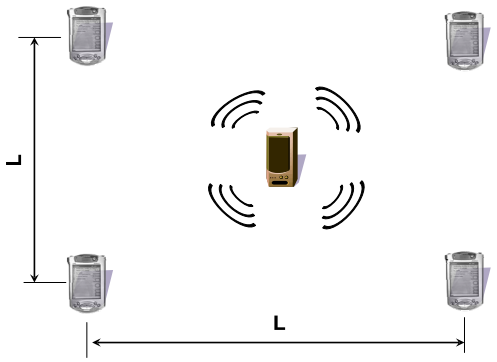

Figure 5. Square Configuration

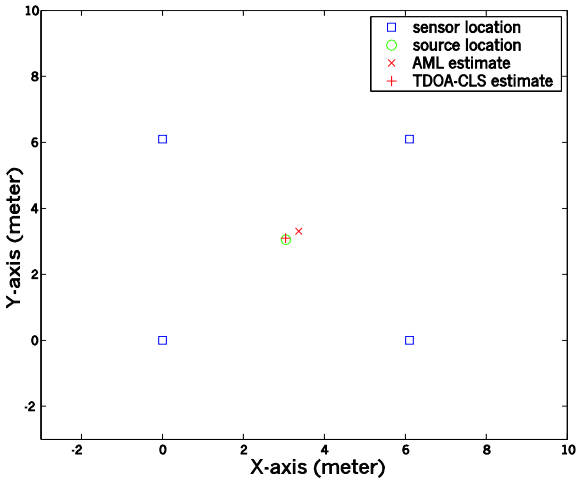

Figure 6. Source localization result of a driving vehicle signal

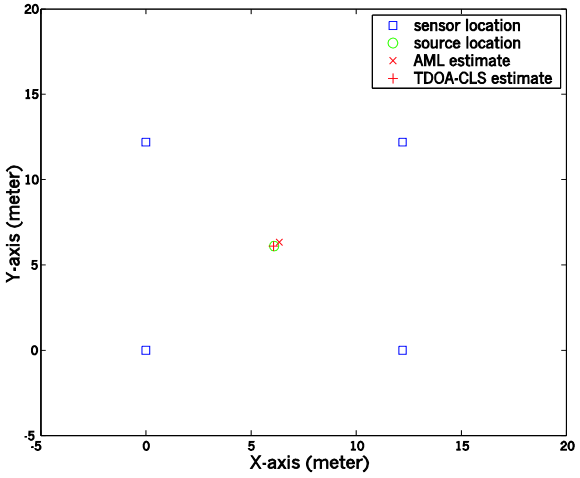

Figure 7. Source localization result of a music signal

speaker can be estimated with a RMS error of 0.407 m using the near-field AML source localization algorithm. A RMS error of 0.0473 m was observed for the same data using the two-step CLS method. We used similar configuration for the second experiment, except the inter-sensor spacing **L** is set to be 40 ft (12.2 m). The loud speaker played a prerecorded organ music, whose spectrum has a 2 kHz bandwidth and 1.75 kHz central frequency. For this wideband source, the AML algorithm can estimate the source with 0.355 m RMS error, while the RMS error by using TDOA-CLS is 0.0349 m (see fig. 7). Thus, both beamforming methods are capable of locating the source, but TDOA-CLS gives a better performance.

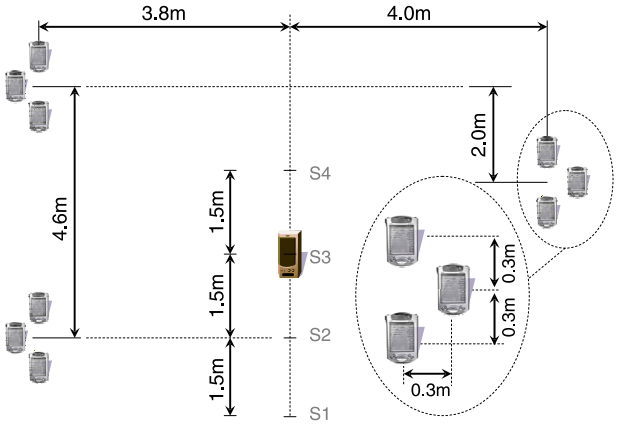

Figure 8. Array configuration 1

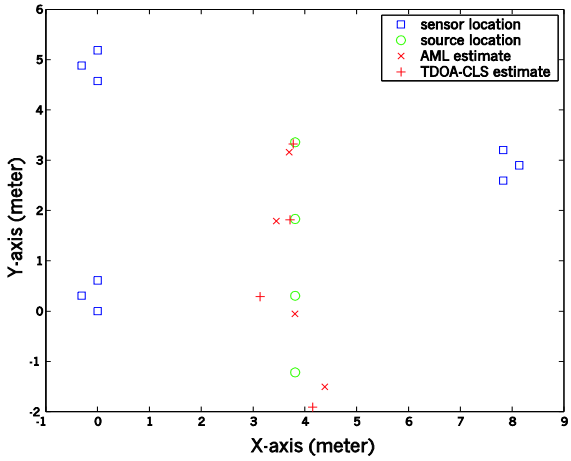

Figure 9. cross bearing result of a vehicle signal

We consider two different subarray configurations for the far-field cross bearing source localization. In one configuration, we used a triangular sensor array as shown at fig. 8. The cross-bearings from three widely separated subarrays yield the estimated source locations, S1,..,S4, in fig. 8. The same vehicle sound was played at four different loca-

tion once at a time. The estimated results of AML as well as TDOA-CLS are shown in fig. 9. It can be seen that more accurate estimation can be obtained when the source is in the middle of the subarrays. This maybe due to more coherent waveforms are received at all three subarrays when the source is inside the convex hull of the sensors. Comparison of AML and TDOA-CLS can also be made. Unlike the near-field case, the performances of these two methods are similar. The RMS errors of 0.638 m at location S4 and 0.225 m at location S1 are reported for AML method. With TDOA-CLS, the RMS error is 0.769 m at S4 and 0.051 m at S1.

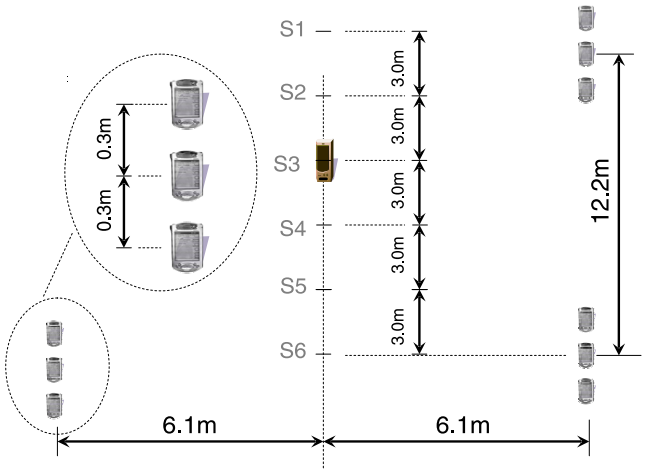

Figure 10. Array configuration 2

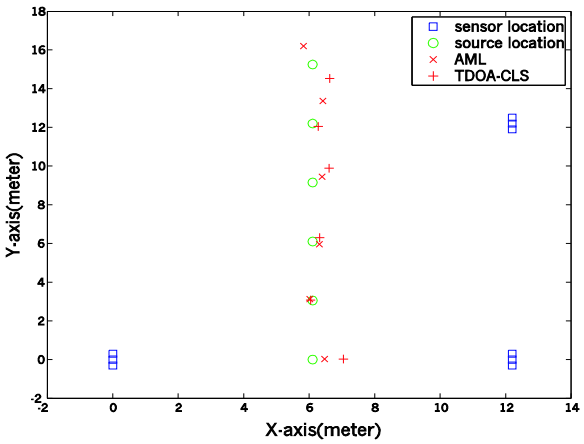

Figure 11. cross bearing result of a music signal

In the configuration of fig. 10, we use a linear subarray with three sensors. Three bearing estimates locate six different locations of the source, which played the music sound as before. The results are shown at fig. 11. Similar performance of the bearforming methods can be observed: with AML, an RMS error of 0.998 m is observed for the source location S1 and an RMS error of 0.115 m is observed at location S5. With TDOA-CLS, the RMS errors of 0.893 m and 0.064 m are reported for locations S1 and S5 respectively.

6. CONCLUSIONS

In this paper, we first describe the testbed using iPAQ 3760s to perform time synchronization and data collection. Then we describe two beaforming methods for source localization and DOA estimations and associated CRB analysis. Some simulation and experimental results, illustrating the usefulness of this sensor network system, are reported.

7. 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, to appear Aug. 2002.

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

[8] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference braodcasts", Technical Report UCLA-CS-020008, University of California, Los Angeles, May 2002. `http://lecs.cs.ucla.edu/publications`.

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