Acoustics'08
Paris
June 29-July 4, 2008
www.acoustics08-paris.org

Acoustic Source Localization via Distributed Sensor Networks using Tera-scale Optical-Core Devices

Neena Imam and Jacob Barhen

Oak Ridge National Laboratory, 1 Bethel Valley Road, Oak Ridge, TN 37831-6015, USA
imamn@ornl.gov

For real-time acoustic source localization applications, one of the primary challenges is the considerable growth in computational complexity associated with the emergence of ever larger, active or passive, distributed sensor networks. The complexity of the calculations needed to achieve accurate source localization increases dramatically with the size of sensor arrays, resulting in substantial growth of computational requirements that cannot be met with standard hardware. One option to meet this challenge builds upon the emergence of digital optical-core devices. The objective of this work was to explore the implementation of key building block algorithms used in underwater source localization on an optical-core digital processing platform recently introduced by Lenslet Inc. We investigate key concepts of threat-detection algorithms such as Time Difference Of Arrival (TDOA) estimation via sensor data correlation in the time domain with the purpose of implementation on the optical-core processor. We illustrate our results with the aid of numerical simulation and actual optical hardware runs. The major accomplishments of this research, in terms of computational speedup and numerical accuracy achieved via the deployment of optical processing technology, should be of substantial interest to the acoustic signal processing community.

1 Introduction

Acoustic source localization by means of distributed sensor networks requires very accurate time delay estimation. The use of passive sensor arrays for estimating the position of a generic acoustic source represents an old and well-investigated area. Time delay estimation techniques have been applied extensively to this area. Many of these techniques are specific to the geometrical configuration adopted for array placement thus imposing heavy restrictions on the choice of sensor configuration [1-2]. Recently however, a great deal of effort has been devoted to the extraction of spatio-temporal information from a matrix of spatially distributed sensors [3]. Notwithstanding the considerable progress reported over the years, today's leading paradigms for acoustic source localization still face substantial degradation in the presence of realistic ambient noise and clutter [4]. Figure 1 illustrates a typical distributed sensor network employed for submerged threat detection. The sensor matrix is comprised of randomly placed GPS-capable sonobuoys. The buoys are passive omnidirectional sensors that provide sound pressure measurements of the ambient conditions and of the signal emitted/reflected from the target. A self-localizing sonobuoy field provides a unique mode of underwater target detection in terms of its deployment flexibility, signal acquisition speed, focused ranging, and capability for net-centric information fusion. However, demanding calculations need to be performed to achieve source localization, and the computational complexity is known to increase dramatically with the size of the sensor array. Without the simplifying assumption of regularly placed sensors, a substantial processing power requirement is necessary that cannot readily be met with standard, off-the-shelf computing hardware. The Center for Engineering Science Advanced Research (CESAR) at the Oak Ridge National Laboratory is involved in the development and demonstration of exciting unconventional technologies for Distributed Sensor Signal (DSS) processing. The CESAR efforts in the area of DSS processing are driven by the emergence of powerful new processors such as the IBM CELL [5], and the EnLight processing platform recently introduced by Lenslet Inc. The latter, a tera-scale digital optical-core device, is optimized for array operations, which it performs in fixed-point arithmetic at 8-bit precision (per clock cycle). Its peak performance is at least two orders of magnitude faster than the fastest Digital Signal Processor (DSP) available today. This research presents a methodology for locating underwater threat sources from uncertain sensor data. A novel paradigm for implementing the Time Difference Of Arrival (TDOA)

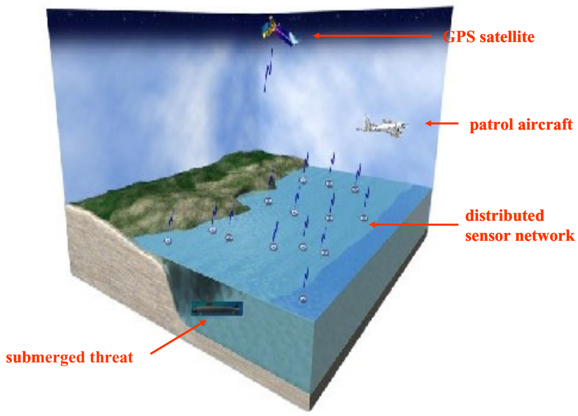

Figure 1: A distributed sensor network.

calculation on an EnLight device is also discussed. The specific goals of this proof-of-concept effort were to demonstrate the ability to achieve required accuracy in the computations and to quantify the speed-up achieved per EnLight processor as compared to a leading-edge conventional processor (Intel-Xeon or DSP). The algorithm is designed for a single sound source localization using a distributed array of acoustic sensors. Conventional TDOA estimation procedures are used. The major focus of this paper is the time-domain implementation of TDOA estimation. The frequency domain counterpart of the analysis, complete with matched filter bank simulation for active sonar platforms detecting both target range and velocity via Doppler-sensitive waveform synthesis and generation, is presented in previous publications by the authors [6-7].

2 Technical Background

2.1 Source localization in a moving sensor field

Locating/tracking an acoustic target involves the estimation of mutual time delays between the direct-path wavefront arrivals at the sensors. Several acoustic source localization methodologies based on TDOA estimation in distributed sensor-nets are available [8-10]. In Ref. 9, an estimate for the source location is found given the TDOAs and the distributed sensor positions using Maximum Likelihood (ML) procedures. The conventional methodologies for the emitter location problem usually include iterative least squares and/or ML estimates. However, closed-form non-iterative solutions can be derived that are usually less computationally burdensome than iterative least squares or ML methods [11]. It is interesting to observe that all methodologies mentioned above require, as a necessary first step, accurate estimates of TDOAs for each combination of sensor/target to be obtained. Thus, for this proof-of-concept demonstration, our effort has focused on TDOA computations.

A signal *s(t)* emanating from a remote source is attenuated and corrupted by noise as it travels through the propagation medium. Signal *s(t)* is received as *x(t)* and *y(t)* at two spatially distributed sensors. The received signals can be mathematically modeled as

$$\begin{aligned} x(t) &= s(t) + n_1(t) \\ y(t) &= \alpha s(t+\tau) + n_2(t). \end{aligned} \qquad (1)$$

Here the signal *s(t)* and noises $n_1(t)$ and $n_2(t)$ are assumed to be uncorrelated and α is the attenuation constant. In distributed sensor networks, it is of interest to estimate the delay, τ. The arrival angle of signal *s(t)* relative to the sensor axis may be determined from the time delay τ [12]. One common method of determining the time delay τ is to compute the cross-correlation function

$$R_{x,y}(\tau) = E\left[x(t)y(t-\tau)\right]. \qquad (2)$$

Here *E* denotes expectation. The argument τ that maximizes (2) provides an estimate of time delay. Because of finite observation time however, $R_{x,y}(\tau)$ can only be estimated. For example, an estimate of the correlation for ergodic processes is given by [13]

$$\hat{R}_{x,y}(\tau) = \frac{1}{T-\tau}\int x(t)y(t-\tau)dt. \qquad (3)$$

It is also possible to extract the time domain function $R_{x,y}$ from its frequency domain counterpart, the cross power spectral density $G_{x,y}(f)$. The cross-correlation between *x(t)* and *y(t)* is related to the cross power spectral density $G_{x,y}(f)$ by the following well-known equation

$$\hat{R}_{x,y}(\tau) = \int_{-\infty}^{\infty} \hat{G}_{x,y}(f)\exp^{j2\pi f\tau}\, df \qquad (4)$$

The quantity $\hat{G}_{x,y}(f)$ is an estimate of $G_{x,y}(f)$. This is of interest because $\hat{G}_{x,y}(f)$ can be computed very fast by the optical-core processor introduced in the sequel. For the purpose of this research, a time domain analysis, calculating the correlation function $R_{x,y}$ directly from the sliding sum of the discrete-time sampled data sequences x_k and y_k, was implemented.

2.2 EnLight optical core processor

Research efforts at Oak Ridge National Laboratory include the feasibility demonstration of high precision computations for grand challenge scientific problems using the novel, Lenslet-developed, EnLight™256 processing platform. EnLight™256 is a small factor signal-processing chip (5.5 cm²) with an optical core. The optical core performs the Matrix-Vector Multiplications (MVM), where the nominal matrix size is 256×256. The system clock is 125 MHz. At each clock cycle, 128K multiply-and-add Operations Per Second (OPS) are carried out, which yields a peak performance of 16 trillion operations per second (or TeraOPS). The architecture of such a device provides a strong rationale for using it in matrix-based applications. Due to the inherent parallelism of the architecture, the computational speed increases with the scale of the problem. The scaling penalty of the optical chip is relatively small compared to standard DSP electronics. The TDOA algorithm discussed in this paper was implemented on both the existing EnLight™64α prototype hardware, and the scaled-up EnLight™256 simulator. The EnLight™64α prototype board is a proof-of-concept demonstration hardware for the optical processor technology with a reduced size optical core. The EnLight™256 hardware is in the development process while the EnLight™256 simulator provides the opportunity to examine DSS implementation on this faster platform. EnLight™64α has an operating clock of 60 MHz. The optical core has 64 input channels, comprised from 256 vertical cavity surface emitting lasers that are configured in groups of 4 per channel. The size of the active matrix is 64×64, which is embedded in a larger

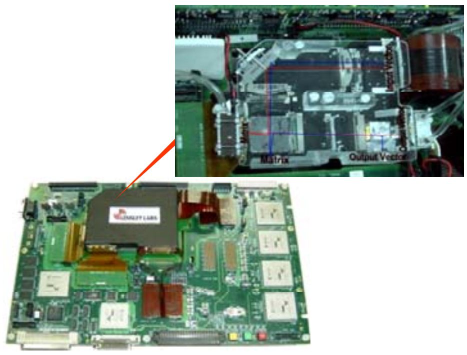

Figure 2. The EnLight optical-core processor.

Multiple Quantum Well (MQW) spatial light modulator of size 264× 288. Sixty-four light detectors, integrated with an array of analog-to-digital converters, comprise the 64 output channels. The optical core performs the MVM function at the rate of 60 $10^6 \times 64^2 \times 2$ = 492 Giga operations per second. Each of the 64 data components in the input and output channels has an 8-bit accuracy, which results in a data stream of 60 $10^6 \times 64 \times 8$ bits/s = 30.7 Giga bits per second. Figure 2 shows the EnLight™64α prototype board.

3 Numerical Simulation

In mobile target detection schemes, such as active sonar systems, the accurate estimation of TDOA by filtering through severely noisy data is crucial for tracking and target parameter (such as velocity) estimation.. To benchmark the EnLight performance, two computer codes were written, one using the EnLight™256 simulator, and the other in MATLAB. The MATLAB code readily interfaces with the software of EnLight™256 simulator, which is used to design the actual algorithm that either runs on the existing

EnLight™64α hardware platform, or is used to project the scaled performance for the EnLight™256. In that framework, a number of operational simplifications are made. In particular, the following is assumed: only a single target is present during the TDOA estimation process, the same speed of sound is experienced at each sensor location, each sonobuoy position is known exactly (via GPS) as it drifts, and the measurement errors for TDOAs are zero-mean Gaussian and independent for each sonobuoy. For the TDOA calculation, a set of synthetic data was generated. The sensor-net comprises 10 sonobuoys. It is assumed that only 7 sensors are able to detect the signal emanating from the target. It is assumed that the target emits a periodic pulsed signal with unit nominal amplitude. Pulse duration is 1 SI (Sample Interval) and inter-pulse period is 25 SIs. The size of one sampling interval is 0.08s. Noise and interference are taken as a Gaussian process with a varying power level (typically upto unity). Each sensor stores sequences of measured signal samples. Sequence lengths can range from 1K to 80K samples. This contributes to the rationale for using correlation techniques in the source localization process. For the time domain analysis, the cross-correlation R_{xy}, for two discrete-time sequences x_k and y_k (each of length M) of sensor data is calculated as

$$R_{xy}(n) = corr([x_k][y_k]) = \sum_{k=0}^{k=M-1} x_k y_{n+k} \qquad (5)$$

where $n = 0,1,............M-1$. The correlation function was calculated for x_k and y_k sequences, both with length M = 1024 and heavily corrupted by zero-mean Gaussian noise of SNR = -28 dB. A 128 shift cross-correlation was calculated in MATLAB. These calculations were also implemented on the actual optical hardware and compared with the MATLAB simulation. Some loss of accuracy is evident due to conversion to 8 bit fixed-point representation in EnLight™64α. However, the same values of the TDOAs, as identified by the cross-correlation peaks, were obtained as the MATLAB simulations, even in the presence of significant noise signal. The hardware implementation scheme, experimental results, and simulation results from MATLAB are presented in figures 3, 5A and 5B.

3.1 Hardware Implementation

The EnLight processor is ideal for implementing large time-series correlation calculations in terms of matrix-vector multiplication operations. The processor works as a matrix-vector multiplier in which a complete MVM operation is performed for each machine cycle (8 ns). Moreover, a new vector can be presented for multiplication every machine cycle. For cases where a new vector is multiplied by the same matrix, there is no Input/Output (IO) communication latency in the processing time. Since a 30 µs IO time is currently needed to reload an entire matrix memory, there is a strong incentive to avoid algorithm constructs where this would have to be done often, and would thereby create an imbalance between IO and core computation. However, changing the entire matrix every multiply operation would be an extremely inefficient and relatively unlikely event. Therefore, the matrix is pre-buffered or loaded onto the spatial light modulator ("local memory") in order to achieve the required processing speed. The algorithms employed take this into account. The particular scheme for correlation calculation on the EnLight platform depends on the length of the two time series and the maximum correlation shift to be calculated. The loading scheme for the matrix memory and the vector register needs to be modified according to the specifics of the data sets to be manipulated. A detailed description of the hardware loading scheme for a correlation calculation of $M = 1024$ is presented in Fig. 3. As shown in Fig. 3, the initial step in the calculation is to build a 256×4 matrix M_1 from time series x_k where the sequence length M = 1024. Next a 256×1024 matrix M_2 is built from time series y_k where each row is shifted to the left by one element with respect to the previous row. The end elements are padded with zeros. This scheme is followed for the first 128 rows, as a correlation for maximum shift of 128 is performed for this example. Rows 129-256 are padded with zeros. Next, M_2 is

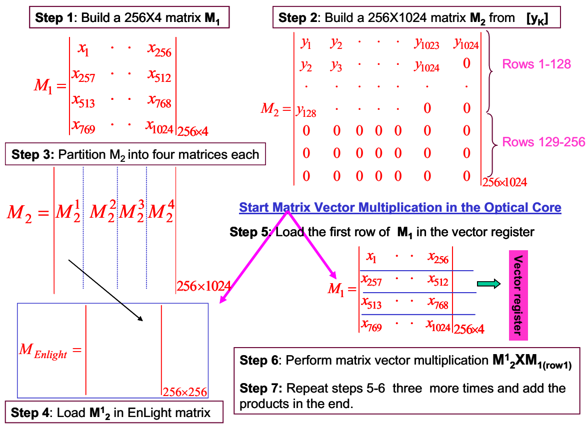

Figure 3: The data loading scheme for EnLight™256.

partitioned into four matrices each with dimension 256×256 as shown in Fig. 3. After the matrices M_1 and M_2 are constructed, they are loaded into the optical hardware. First the sub-matrix M_2^1 is loaded into the EnLight matrix memory. Then the first row of matrix M_1 is loaded into the vector register. Matrix-vector multiplication is performed as M_2^1 M_1(row1). Steps 5-6 are repeated three more times and the products are added at the end to produce the 128 shift correlation. For the example in hand, the data sequence length is 1024, EnLight matrix size is 256×256, and the vector register size is 256×1. Four machine cycles are needed to implement the calculations (1024/256 = 4). Each matrix-vector multiplication in the optical core takes 8 ns. With one processing node, a total of 8×4 = 32 ns is required to complete the entire 128 shift correlation function. If multiple processing nodes are used then this time is further reduced. As is evidenced by the previous example, one has to be somewhat aware of hardware architecture while programming in the EnLight device. The optimization of the loading schemes for the matrix memory and vector registers, as dictated by the details of the algorithm, is also another area of intellectually stimulating research. The signal processing flow diagram of Fig. 4 outlines the hierarchical structure of software interfaces with the EnLight processing board, where the higher-level programming languages such as FORTRAN, C, or MATLAB (current implementation) generate Hardware Development Language (HDL) files and bit-streams via the use of Xilinx Sysgen blocks of the MATLAB/Simulink module to program the FPGAs that access the optical core. As shown in figures 5A and 5B, excellent results were obtained using simple data-scaling procedures, without need to invoke (at this point) available [14], more sophisticated techniques for high accuracy computation with low precision devices.

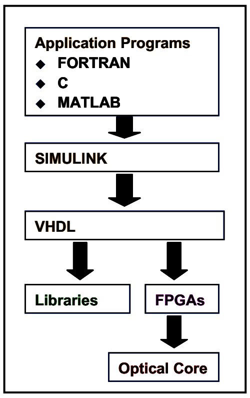

Figure 4: The software interface flow diagram.

4 Results and Discussion

The correlation functions R_{mn} were calculated for each sensor pair in the time domain and implemented on the hardware in order to demonstrate the loading scheme discussed in the previous section (also illustrated in Fig.3). As the EnLight^{TM}256 device does not exist yet, the actual hardware calculations were performed on the EnLight^{TM}64α prototype board. The extension of the loading scheme to the α board is straight-forward but more machine cycles (4 times as many) are needed to perform the same calculations. Figures 5A and 5B compare the MATLAB simulations with the EnLight^{TM}64α hardware runs. As can be seen, the numerical accuracy (with respect to the correct locations of the cross-correlation peaks) of the

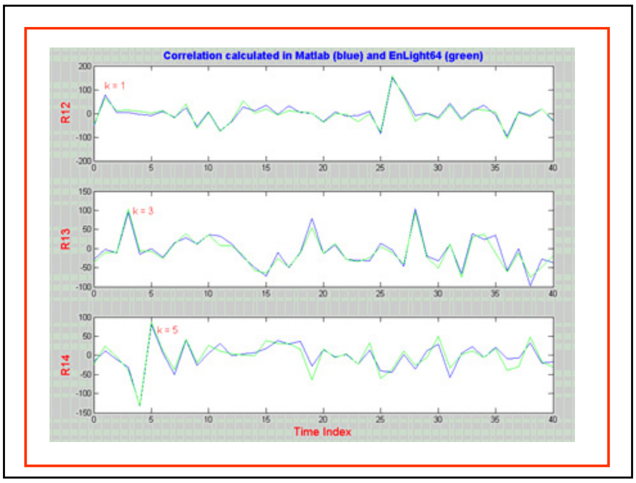

Figure 5A: Comparison of hardware run with MATLAB simulation for R_{12}, R_{13}, and R_{14}.

hardware runs compares very favorably with the high precision MATLAB simulations. The green plots represent hardware runs and the blue plots represent MATLAB simulations. The x-axes are expanded for each plot for better visualization of the correlation peaks. Some loss of accuracy in the magnitudes of the correlation functions is evident due to the conversion to an 8-bit precision scheme. However the locations of the correlation peaks coincide with the MATLAB results for R_{12}, R_{13}, R_{14}, R_{15}, R_{16}, and R_{17}. The simulation and hardware data sets were further compared by calculating the percent difference in the magnitudes of the cross-correlation function as $\Delta = (R_{MATLAB} - R_{EnLight} / R_{MATLAB}) \times 100$ Δ. The Δ values were calculated for the cross-correlation peaks that identify the estimated time delay τ. Table 1 lists the various Δ values. As can be seen, the Δ values range from 17% (R_{12}) to 2% (R_{17}). For some applications these deviations in numerical values may be considered too high. However, for the benchmark source localization problem discussed in this paper, the absolute magnitudes of the cross-correlation functions are not important for the accuracy of TDOA estimation. It is the locations of the cross-correlation maxima and the relative ratios of the magnitudes of the maxima that are crucial for the determination of the quantity τ. The limited precision EnLight optical-core processor correctly identifies the TDOA values as does the high precision MATLAB simulation. In order to take advantage of the processing speed of the optical-core processor for DSS applications, one has to be aware of the device architecture and its limitations. The algorithm also needs to be adapted to circumvent the device limitations. As has been demonstrated, for properly structured algorithms, the 8 bit native accuracy of the optical chip is

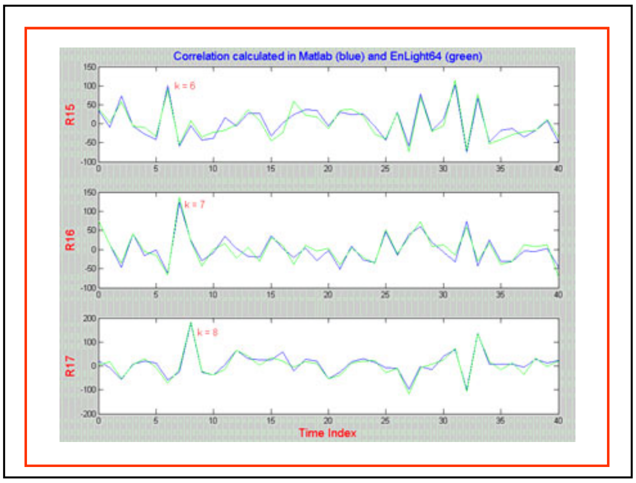

Figure 5B: Comparison of hardware run with MATLAB simulation for R_{15}, R_{16}, and R_{17}.

not an impediment to accurate underwater source localization. On the other hand, the high processing speed of the EnLight platform offers advantages for DSS applications that are unparalleled by conventional processors. In addition, it is possible to extend the accuracy of the EnLight calculations by employing advanced parallel data processing techniques as discussed in Ref. 14. Research is also underway to improve the native accuracy of optical-core platforms via improved hardware design. In terms of processing speed, benchmark calculations were carried out for Fourier transforms of long signal sequences. In particular, the execution speed of the EnLight^{TM}64α was compared to that of a computing platform using dual Intel Xeon processors running at 2 GHz and having 1 GB RAM. The benchmark involved the computation of 32 sets of 80K complex samples transforms. For each sample, both the forward and the inverse Fourier transforms were calculated. The measured times were 9,626 ms on the dual Xeon system, versus 1.42 ms on the EnLight. This corresponds to a speedup of over 13,000 on a per processor base. More details on these computations can be found in Refs. 6-7.

	R_{12}	R_{13}	R_{14}
	k=1	k=3	k=5
MATLAB	79.1978	95.5889	80.3471
EnLight64α	65.0842	103.5966	85.9405
Δ	17.82%	8.38%	6.96%
	R_{15}	R_{16}	R_{17}
	k=6	k=7	k=8
MATLAB	100.9231	123.0550	181.2274
EnLight64 α	91.9592	137.1272	185.1162
Δ	8.88%	11.44%	2.15%

Table 1: Comparison of MATLAB simulation and EnLight64 hardware run.

5 Conclusion

Distributed sensors with optical computing platforms as onboard devices present an attractive alternative to conventional dedicated sensor arrays. Future advances in DSS signal processing for improved target detection, tracking, and classification in highly noise-corrupted environments can be realized through the development of distributed systems that combine superior sensors and highly efficient computational nodes consisting of optical-core devices such as the EnLight platform. The numerical simulations and hardware implementation presented in this paper build the first stage in creating a test-bed for evaluating the performance of digital, optical-core processors in facilitating DSS signal processing. Preliminary estimates for the TDOA computation, the core of many source localization algorithms, implemented on an EnLight prototype processor indicate a speed-up factor of the order of 13,000 compared to a dual processor Xeon system. Combined with its low power requirements (approximately 50W per processor), the projected tera-scale throughput of optical-core processor technology can alleviate critical signal processing bottlenecks of relevance to many distributed sensor-net programs. This, in turn, should enable the efficient implementation of new classes of algorithms not considered heretofore because of their inherent computational complexity such as asynchronous, multi-sensor, multi-target tracking under uncertainty of noise characteristics.

Acknowledgments

Primary funding for this work was provided by the Office of Naval Research. Additional support was provided by the ORNL Laboratory Directed Research and Development (LDRD) program. Oak Ridge National Laboratory is managed by UT-Battelle, LLC for the US Department of Energy under contract number DE-AC05-00OR22725.

References

[1] A. L. Van Buren, "Near-field transmitting and receiving properties of planar near-field calibration arrays," J. Acoustical Society of America, vol. 89, pp. 1423-1427, March 1991.

[2] M. Viberg and A. L. Swindlehurst, "A Bayesian approach to auto-calibration for parametric array signal processing," IEEE Trans. Signal Processing, vol. 42, pp. 3495-3507, December 1994.

[3] V. C. Raykar, I. V. Kozintsev, and R. Lienhart, "Position calibration of microphones and loudspeakers in distributed computing platforms," IEEE Trans. on Speech and Audio Processing, vol. 13, pp. 70-83, January 2005.

[4] A. H. Nuttall and J. Wilson, "Adaptive beamforming at very low frequencies in spatially coherent, cluttered noise environments with low signal-to-noise ratio and finite-averaging times," J. Acoustical Society of America, vol. 108, pp. 2256-2265, November 2000.

[5] M. Gschwind, B. D. D'Amora, J. K.O'Brien, K. O'Brien, A. E. Eichenberger, and P.Wu, "Cell broadband engine-enabling density computing for data-rich environments," Annual Int. Symposium on Computer Architecture, January 2006.

[6] J. Barhen, N. Imam, M. Vose, A. Averbuch, and M. Wardlaw, "Underwater threat source localization: processing sensor network TDOAs with a terascale optical-core device," in E. Lefebvre *et. al.* eds, NATO ASI Book Series, pp. 56-68, IOP Press (2007).

[7] J. Barhen and N. Imam, "Sensor data processing for tracking underwater threats using terascale optical core devices, *NATO ARW Book Series,* IOS Press (in press).

[8] C. H. Knapp and G. C. Carter, "The generalized correlation method for estimation of time delay," IEEE Trans. Acoustics, Speech, and Signal Processing, vol. 24, pp. 320-327, August 197

[9] T. Ajdler, I. Kozintsev, R. Lienhart, and M. Vetterli, "Acoustic source localization in distributed sensor networks," Asilomar Conf. Signals, Systems and Computers, vol. 2, pp. 1328-1232, November 2004.

[10] S. Nardi and M. Pachter, "GPS estimation algorithm using stochastic modeling," Proc. 37th IEEE Conf. Decision and Control, pp. 4498-4502, December 1998.

[11] G. Mellen, M. Pachter, and J. Raquet, "Closed-form solution for determining emitter location using time difference of arrival measurements," IEEE Trans. Aerospace Electronic Systems, vol. 39, pp. 1056-1058, July 2003.

[12] A. H. Nuttall, G. C. Carter, and E. M. Montavon, "Estimation of the two-dimensional spectrum of the space-time noise field for a sparse line array," J. Acoustical Society of America , vol. 55, pp. 1034-1041, May 1974.

[13] A. Papoulis, *Probability, Random Variables and Stochastic Processes,* New Jersey: MacGraw-Hill, 1970.

[14] A. Fijany, J. Barhen, and N. Toomarian, "Signal processing applications of massively parallel charge domain computing devices: Part II," U.S. Patent No. 5952685 (1999).