Collaborative and Responsive Sensors for Low-Cost Spectrum Sensing and Geolocation

John Merritt IV*, Charles Dietlein†, Jonathan Chisum*
*University of Notre Dame, Department of Electrical Engineering
Notre Dame, IN 46556
UNITED STATES
†US Army Research Laboratory
2800 Powder Mill Road
RDRL-SER-W
Adelphi MD 20783
UNITED STATES

jmerrit1@nd.edu, jchisum@nd.edu, charles.r.dietlein.civ@mail.mil

ABSTRACT

Distributed sensor networks can serve as a robust electronic warfare support capability for sensing in contested and congested electromagnetic environments. While the response of an individual node may be unreliable due to interference or poor signal to noise ratio (SNR), the overall sensor network performance is minimally affected. In addition, a distributed sensor network enables geolocation of challenging emitters. Both sensing and geolocation benefit from increased sensor density. Therefore we present a low-cost distributed sensor network for spectrum sensing and emitter geolocation. To meet the low-cost objectives (<$70 per sensor) for this network we relax hardware performance requirements for individual sensor nodes resulting in analog impairments including poor image rejection, local oscillator drift, high phase noise, and limited dynamic range. In this work we show that cross-correlation signal processing can be used to compensate for poor node performance.

Many distributed sensor networks rely upon an asymptotic network comprised of a very large number of nodes in order to achieve acceptable network performance in spite of poor node performance. This approximation is not applicable when a limited number of nodes are present, particularly when they are clustered into local groupings. In this work we discuss a non-asymptotic, tactically-relevant sensor network clustered into squads of 5-10 nodes. To ensure overall performance of the network our approach is to connect individual nodes through a backhaul network and operate them in a collaborative and responsive mode. In this context, "collaborative" implies that nodes are able to communicate locally (in a cluster, e.g., squad) and globally (with a network controller, e.g., platoon), and "responsive" implies nodes are reactive and can change their operating state depending on input from a local node or the global controller to mitigate various analog impairments.

We develop a hierarchical signal processing method in which cross-correlation is used at the squad level for improving the performance of a cluster of sensors, and then higher-level algorithms are used at the network level. The method leverages differences in free-space path loss between sensors in a squad to achieve high linearity while maintaining low-noise levels. Performance is quantified through spur-free dynamic range (SFDR). Multiple measurements of a sensor-pair can be averaged to increase SFDR, or multiple sensors can be pair-wise averaged to increase SFDR while maintaining scan rate. We show that cross-correlation is a natural choice for processing clustered networks and that higher network level decisions can optionally adjust the performance of individual responsive nodes by adjusting front-end attenuators.

1.0 INTRODUCTION

Distributed sensor networks can serve as a robust electronic warfare support capability for sensing in contested and congested electromagnetic environments. While the response of an individual node may be unreliable due to interference or poor signal to noise ratio (SNR), the overall sensor network performance is minimally affected. In addition, a distributed sensor network enables geolocation of challenging emitters. Both sensing and geolocation benefit from increased sensor density; therefore, many distributed sensor networks rely upon an asymptotic network comprised of a very large number of simplistic sensor nodes in order to achieve acceptable network performance in spite of poor node performance. The asymptotic performance trends are not necessarily realized when a limited number of nodes are present, particularly when they are clustered into local groupings (squads). In this work we discuss a non-asymptotic, tactically-relevant sensor network distributed into several clusters, each with 5 to 10 nodes as shown in Fig.1. Sensors within a squad are assumed to "see" mostly the same emitter signals and are coarsely time synchronized (<100 microseconds). That is, an emitter of interest will be observed by all sensors within a squad (albeit with varying attenuation and delay) during a processing time window but, depending on geometry, other squads may not observe the same emitter. Therefore, sensor clustering naturally lends itself to leveraging squads to gain performance through redundancy and collections of squads to provide geographical coverage.

To ensure performance of the overall network our approach is to perform hierarchical signal processing wherein data is first processed within a squad (intrasquad) to provide increased performance (e.g., higher dynamic range, faster frequency scanning, etc.) and then it is analysed between squads (intersquad) to optionally modify the sensor network. We refer to this as a *collaborative* and *responsive* sensor network. In this context, "collaboration" implies that nodes are able to communicate locally (intrasquad) and globally (with a network controller, intersquad) and their data products can be jointly analysed to increase performance either at the squad or network level. "Responsive" implies nodes are reactive and can change their operating state depending on input from a local node or the global controller to mitigate various analog impairments. We emphasize that significant bandwidth requirements are constrained to intrasquad processing and only the higher quality derived dataset is transmitted between squads which reduces backhaul bandwidth requirements.

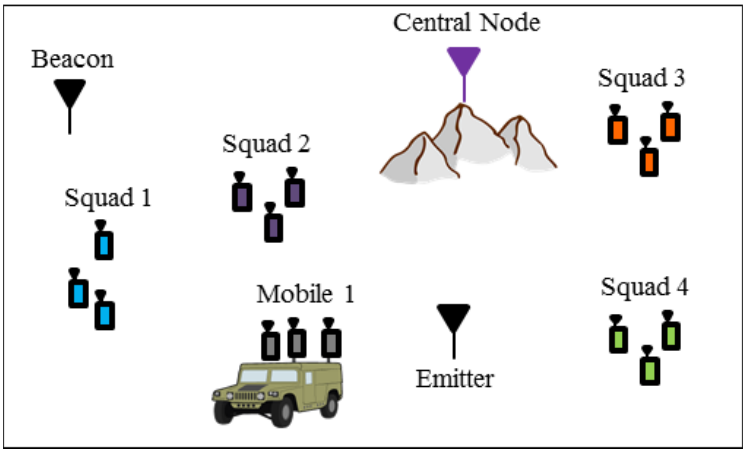

Figure 1: The tactical scenario of multiple dispersed squads of 5-10 clustered sensors is considered in this work. Emitter are assumed to not reach all squads due to geometry/topography.

It is common in distributed sensor networks that sensor nodes are simplistic to achieve low-cost and/or low-power operation. The simplicity limits the quality (speed, fidelity, accuracy) of the data product from a node and ultimately limits the choice of signal processing methods that can be applied across the network. For example, at the network-level a low-cost sensor network is often not time-synchronized so that geolocation using time-of-

arrival (TOA) methods is not possible, or similarly, sensors may not share a phase reference and therefore distributed beam-forming geolocation methods are not possible. At the node-level analog impairments such as image-frequency interference and LO leakage (due to in-phase (I) and quadrature (Q) imbalance of low-cost direct-conversion receivers) or nonlinear distortion (linearity is intentionally sacrificed to reduce power consumption) limit sensitivity and accuracy.

We have recently developed the RadioHound spectrum sensing node, shown in Fig. 2 and partially described in [1], which is comprised of a 25 MHz to 6 GHz down-converter front-end which expands the operating range of the low-cost ($20) receive-only RTL-SDR software defined radio, and a Raspberry Pi to provide control/processing, data storage, and network interface. Total sensor cost is ~$70, total power dissipation is less than 3 watts, and total physical envelope is less than 12 cubic inches. Sensor image rejection is approximately 25 decibels (dB) across the band and the 1 dB input compression point is +5 dBm. This modest hardware capability derives from the low-cost requirement for the node. For this work each sensor front-end is augmented with a 0 dB to 31 dB variable attenuator to make each node responsive in dynamic range. The RadioHound sensor samples the spectrum at 2 MSPS in a 2 MHz RF band, records a time stamp, location, and either the complex baseband time-domain signal or the resulting periodogram (or power spectrum), then tunes to the next RF band and repeats. An entire scan (25 MHz to 6 GHz) in completed in ~3 seconds. The RadioHound nodes operate through a central controller over a backhaul link (currently Wi-Fi, or for field tests through a mobile ad-hoc network (MANET)) where measurement instructions are conferred and data (including metadata, spectrograms, and even raw IQ samples) are reported. Once the data are stored by the network controller (currently using cloud storage) visualization algorithms (such as spectrum-usage heat maps) can be applied to the data. This paper is organized as follows: in Section 2 we describe the basic approach to the problem of clustered sensor network processing, potential methods that may be implemented and the test setup that will be used to evaluate them. Section 3 compares simulated results considering averaging of multiple spectrum captures (across time) and from multiple sensors. Section 4 shows our experimental results.

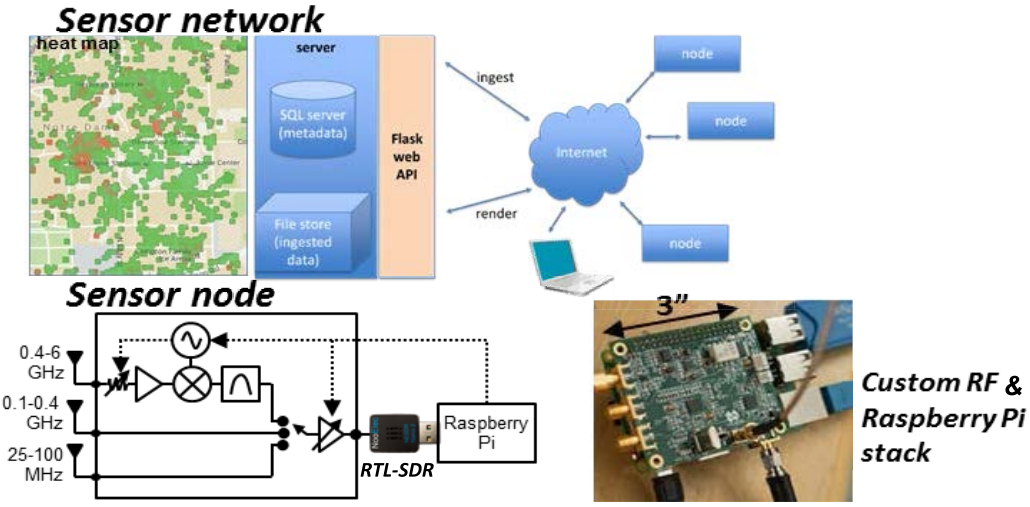

Figure 2: The RadioHound "Sensor node" includes a custom down-converting front-end to extend the RTL-SDR receive-only radio to cover 25 MHz to 6 GHz. The down-converter is compatible with the Raspberry Pi HAT expansion protocol and so directly stacks onto the Raspberry Pi (Pi). The Pi is a low cost controller providing storage and backhaul. The "Sensor network" is comprised of sensor nodes communicating with a central server through a back-haul connection (Wi-Fi or MANET radio for field tests). Measurement instructions (e.g., scan frequencies) are transmitted from the server to the nodes and data (metadata, spectrograms, time-domain samples) is stored locally or transmitted to the cloud-server. Client applications parse cloud-data and provide user visualization tools.

2.0 APPROACH TO CLUSTERED SENSOR NETWORKS

Low-cost, low-power sensor nodes typically employ swept-tuned receivers with limited instantaneous bandwidth and sample rate to meet those criteria. Therefore, to achieve high scan rates and limit the collection of information to the intra-nodal communication rate, only a small number of time-domain samples are recorded in each tuned band. This, in addition to the analog limitations of such low-cost sensors, motivates the question of this work: *"Given a collection (squad) of low-cost, low-power sensor nodes with poor analog linearity, coarse time-synchronization between nodes, and observing approximately similar spectral scenes, how should the squad-level sensor data products be processed?"* In the following we compare two popular methods for deriving power spectra based on power-averaged fast Fourier transforms (FFTs) and cross-correlation. We will show that in the context of spectrum sensing of arbitrary emitters cross-correlation is the preferred approach. Further, we will show that in the context of a distributed sensor network where the path-loss between a particular emitter and the various sensors in a squad can vary over many orders of magnitude, this effect can be leveraged to increase linearity.

Within the cognitive radio community, the signal to noise ratio wall (SNR wall) theory has found credence and become a source of discussion as to the best way to approach the difficulties imposed by its implications, particularly for the use of spectrum sensing. The SNR wall [2,3] states that for a given uncertainty of the estimate of the noise, there is a minimum SNR (for common sensing applications, -6 dB SNR) at which accurate detection is impossible. It was shown in [4] that cross-correlation between two receivers can overcome the SNR wall with an efficiency related to the degree of noise correlation between the two receiver chains. If two receiver chains measure the same correlated signal but add uncorrelated noise a cross-correlation implementation can be realized. By using the cross-correlation to lower the uncorrelated noise, the uncertainty of the noise is also decreased, effectively lowering the SNR wall. In [5] a highly linear spectrum sensor was proposed wherein attenuation was added to the front-end of one of a pair of on-chip sensors and then cross-correlation was employed to recover noise performance. This was demonstrated in hardware with on-chip cross-correlation receiver pairs [6,7] where the objective is to maintain a very low channel-to-channel correlation coefficient. Here we extend the method proposed in [5] to the case of free space and we demonstrate the approach for more than two sensors. We compare the cross-correlation approach with power-averaged FFTs through simulation and then validate the cross-correlation algorithm through free-space measurements.

Here we describe several possible computations and compare their ability to improve overall linearity and noise performance quantified through the spur-free dynamic range (SFDR), defined below. To facilitate this discussion, we define N_{FFT} as the length of an FFT, and M as the number of FFTs averaged together in the following computations:

- **Coherent Averaging of Complex FFTs:** Coherent averaging of M complex N_{FFT}-point FFTs from a single sensor is the most straightforward approach for reducing noise power (-10dB/decade). This is effectively computing a large $M*N_{FFT}$–point FFT. However, for typical signals of interest (e.g., modulated data or pulsed carriers) this method is not viable because the signals are attenuated along with the noise.
- **Power Averaging of FFTs:** Power averaging of M complex N_{FFT}-point FFTs is similar to coherent averaging but uses the magnitude of the FFT values as opposed to the complex data. This has the benefit of avoiding the destructive self-interference that the coherent averaging sees, but also the inability to reduce the noise. Instead only the variance of the noise floor is reduced which does not improve SFDR with increasing M.
- **Autocorrelation:** Averaging M autocorrelated N_{FFT}-point FFTs is very similar to power averaging of FFTs. Instead of taking the magnitude of the FFT, the power spectrum is found as $s*(\text{conj}(s))=\text{abs}(s)^2$.

Thus, again, the variance of the noise floor is reduced while the mean is left the same so SFDR does not improve with increasing *M*.

- **Cross-Correlation:** The cross-correlation method uses a set of *M* N_{FFT}-point FFTs from each of a pair of sensors (total of *2M* FFTs). The *M* cross-correlations, s_1*(conj(s_2)), are averaged and the noise floor reduces by 5 dB/decade increase in *M*, as in [8]. While the other three methods do not improve SFDR over increasing *M*, cross-correlation does but requires two independent sensors.

Considering the above discussion *we proceed with cross-correlation as the method for processing multiple sensors within a squad.* Figure 3 shows a signal flow-graph an input signal, $s(t)$, and correlated noise, $n_{corr}(t)$. A pair of sensor nodes captures the signal and noise adding additional independent uncorrelated noise, $n_{uncorr1}(t)$ and $n_{uncorr2}(t)$. We model different path losses and/or controlled attenuation between sensors with an effective attenuation in the front-end of sensor 2. Both sensors sample the signal in the time-domain then window and compute complex FFTs of length N_{FFT}, which are transmitted to a controller (e.g., a processing node in a squad of sensors). The controller may aggregate *M* FFTs and perform any computation on the set of *M* complex FFTs to generate high quality power spectra.

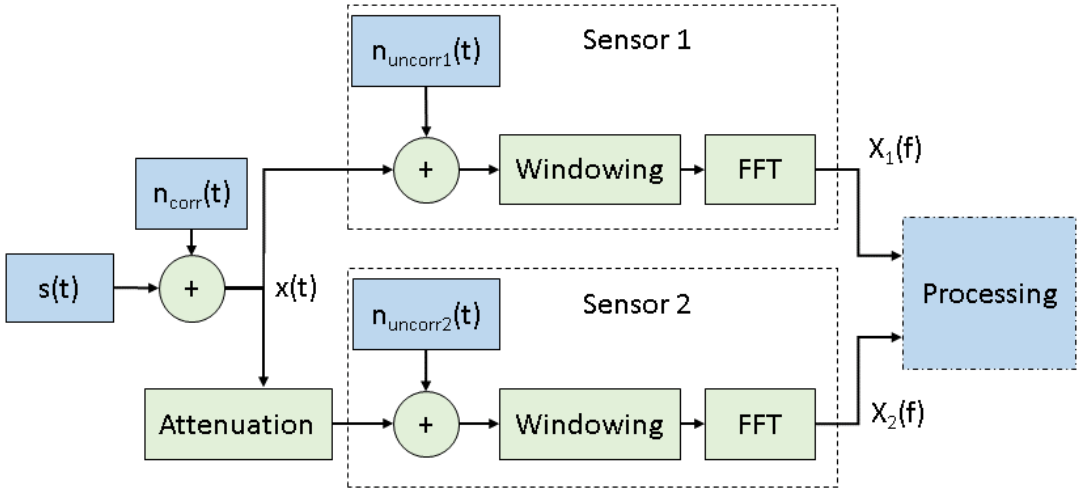

Figure 3. *S(t)* is the signal of interest. *n(t)* represents correlated noise (from wireless channel and common hardware) and uncorrelated noise (independent sensor hardware). Different path losses are represented by attenuation in the front-end of sensor 2. Samples are windowed and a Fourier transform (FFT) is calculated resulting in spectra $X_1(f)$ and $X_2(f)$. This paper discusses various "Processing" algorithms of the distributed sensor data to improve linearity and noise performance.

In addition to increasing SFDR through averaging, there are many other benefits to applying cross-correlation to distributed spectrum sensing. Provided that the signal of interest is present in measurements from both sensors (we define the signal being simultaneously present in at least 50% of the sample set as "coarse synchronization"), the cross-correlation approach is robust to time offsets. In fact, if the signal and environment are wide-sense stationary (WSS), all time delays (including zero) are equivalent. Cross-correlation is also insensitive to non-contiguous data sets. That is, a combination of *M* FFTs of length N_{FFT} may have dropped samples between the FFTs with no effect as long as the same signal exists in all FFTs. Another key benefit to cross-correlated sensors is the ability to usefully process data from a saturated sensor. By cross correlating the saturated sensor measurement with an un-saturated sensor (perhaps due to greater path loss or with more front-end attenuation), the noise floor is reduced and thus signals that would be obscured in a single attenuated sensor become observable. Let us consider the properties of cross-correlation. It is shown to have a -5 dB/decade power

response to white noise against increasing M in [8]. From Fig. 3, $s(t)$ is our signal of interest, and $n_{corr}(t)$ is the correlated noise added in the channel which both sensors see, the input to the sensors is $x(t)=s(t)+n_{corr}(t)$. Additional uncorrelated noise is added within each sensor:

$$x_1(t) = x(t) + n_{uncorr1}(t)$$
$$x_2(t) = x(t) + n_{uncorr2}(t)$$

Consider infinite length signals, $x_1(t)$ and $x_2(t)$, with zero mean uncorrelated noise. The expectation of the cross-correlation is

$$E[x_1 \bullet (x_2)^*]$$
$$E[(x + n_{uncorr1})(x + n_{uncorr2})^*]$$
$$E[|x|^2] + E[x^* n_{uncorr1}] + E[x n^*_{uncorr2}] + E[n_{uncorr1} n^*_{uncorr2}]$$
$$E[|x|^2] = E[|s|^2] + E[|n_{corr}|^2]$$

This is because the mean of the uncorrelated noise is zero, and the two separate realizations of uncorrelated noise are assumed to be independent. Using the Weiner-Khinchin theorem, we can apply this concept to a FFT. Thus, to achieve these results in practice, we would average together an infinite number of realizations of limited length samples. Under the general assumption that the uncorrelated noise is large compared to the correlated noise, this leads to significant improvement in SNR. Figure 4 shows the cross-correlation processing block. It should be understood that the input signals, X_1 and X_2, are the complex FFTs output from the setup in Fig. 3. The top processing block computes the power spectrum, C, from the complex FFT of a single sensor, X_1, and averages M sample sets. The bottom processing block computes the cross-power spectra from the FFTs originating in both sensors, X_1 and X_2. The M cross-power spectra are then averaged to obtain the improved cross-power spectrum. While it may seem pointless to compare cross-correlation to power-averaging because we have already discussed that power averaging only reduces the variance of the noise and cannot lower the noise floor itself, as will become clear through simulation in the next section, there are conditions under which power averaging has greater benefits for small M.

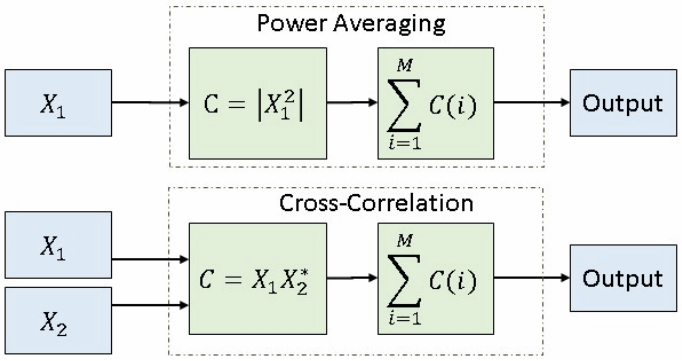

Figure 4. Two possible realizations of the processing block, showing the power averaging method, which has input from only 1 sensor, and our cross-correlation method, which takes FFTs from both sensors. Both methods determine a statistic from the inputs, and average the new information with previous results to get the resulting power spectrum.

3.0 SIMULATION AND ANALYSIS

In order to verify the above discussion and analysis we simulated the SFDR of various receiver configurations. Each receiver is comprised of a simplified third order nonlinearity and a sampling analog to digital converter. A cooperative receiver is comprised of two such sensors where sensor 1 is un-attenuated and sensor 2 includes an attenuator as indicated in Fig. 3. An input signal is created with a tone at 27.174487 MHz and a 50 percent random amplitude modulated tone at 17.146236 MHz (this represents modulated data). The modulation has a symbol rate of 40 symbols per microsecond and the resulting signal is sampled at 100 MHz for 10 milliseconds. Both tones have input power levels of -9.9 dBm. To model coarse time synchronization, one sensor was delayed by a random time up to half of the time span of a sample set. Figure 5 shows the original test signal (yellow). The orange trace shows the signal attenuated by 20 dB and passed through the nonlinearity and to sensor 2. The blue trace shows the signal un-attenuated and passed through the nonlinearity and to sensor 1. Gaussian white noise is added to the individual 'sensors' to simulate uncorrelated component noise from two different hardware sensors. The ability of all sensor configurations and processing algorithms were evaluated using the standard linearity and noise metric, spur-free dynamic range (*SFDR*). In words, *SFDR* quantified the difference between the linear signal of interest and the noise floor when the dominant (third order) nonlinear distortion terms are at the level of the noise floor. *SFDR* is defined as,

$$SFDR = \frac{2}{3}(IIP3 - P_{Noise}),$$

where P_{Noise} is the average power in the noise floor and *IIP3* is the input third order intercept point, the theoretical input power level at which third order distortion power is equal to linear receive power. The objective of each receive-configuration is to maximize linearity and reduce noise so *SRDR* is a natural metric to maximize. In particular, because we assume low-cost/low-power sensors with poor analog performance we are primarily interested in the *SFDR* of a single measurement, rather we are interested in the potential for SFDR to improve as processing time (or effort) increases. In the following we will plot *SFDR* versus the number of measurements processed, *M*.

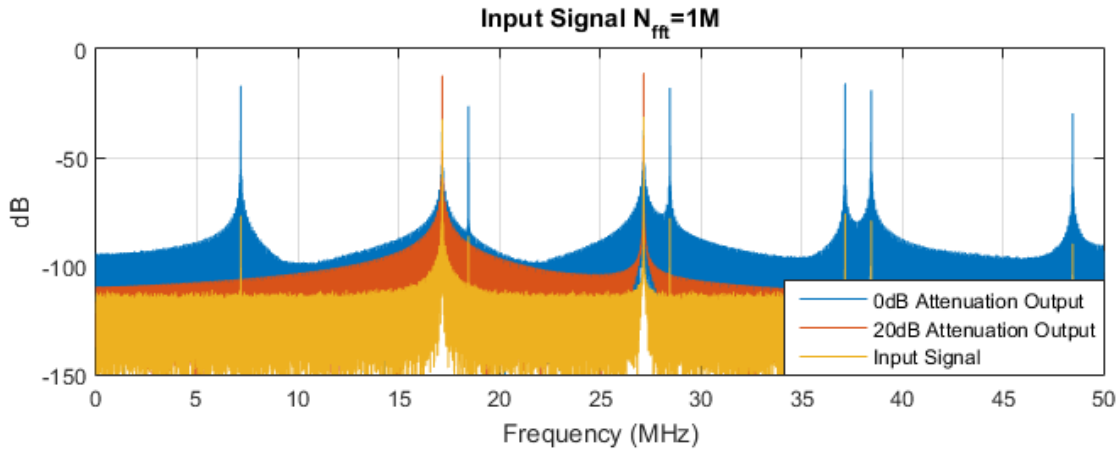

Figure 5: An example pair of input signals used in our test case. The input signal displayed is attenuated by some attenuation before being passed through a simplified third order approximation of a nonlinearity. Uncorrelated white noise is then added so that the output can be used as inputs for our simulated sensors. Two example attenuation settings, 0 and 20 dB are shown.

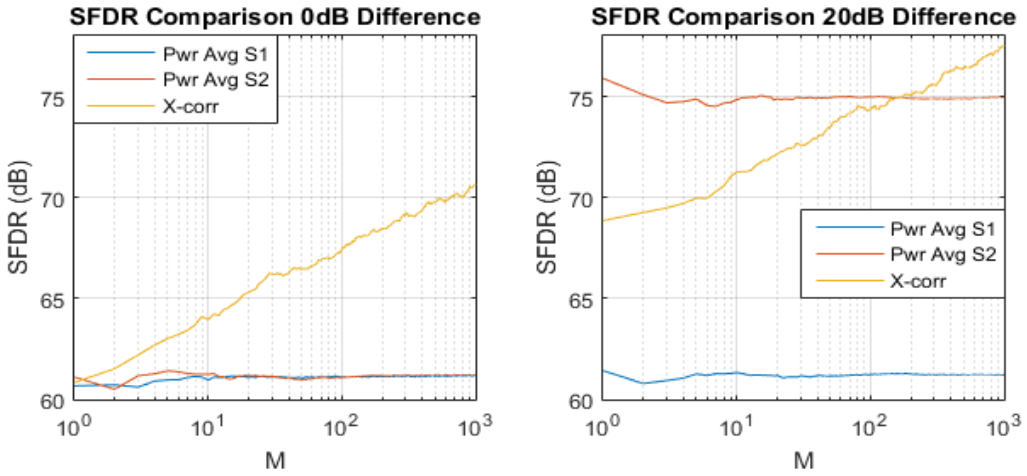

Figure 6: Two simulations were performed to compare cross-correlation against power averaging. On the left, there was no difference in attenuation between sensors while the right had S_2 attenuated by 20dB more. Results are averaged across 100 Monte Carlo simulations. As expected, the power averages do not show improvement over increasing M, while cross-correlation improves constantly.

The first simulation in Fig. 6 compares SFDR for identical sensor attenuation (left) versus one sensor attenuated 20 dB more than the other (right). As discussed in section 2, cross-correlation results in -5 dB/decade decrease in noise power versus number of measurements averaged, M. This corresponds to a *(2/3)*5* dB/decade trend in SFDR. As expected, the power averaged spectrum collected from S_1 and S_2 show no change as M increases. The simulation with a 20 dB attenuation difference shows that at higher attenuations, cross-correlation requires more sample sets to outperform power averaging of the attenuated sensor. This trend is shown in greater detail in Figure 7, where power averaging of the attenuated sensor (dashed) is compared with cross-correlation (solid) for attenuations of 0, 10, and 20 dB. For a 10 dB difference in attenuation, slightly more than 10 dB increase in M is required for the cross-correlation to achieve higher SFDR than power averaging the attenuated sensor.

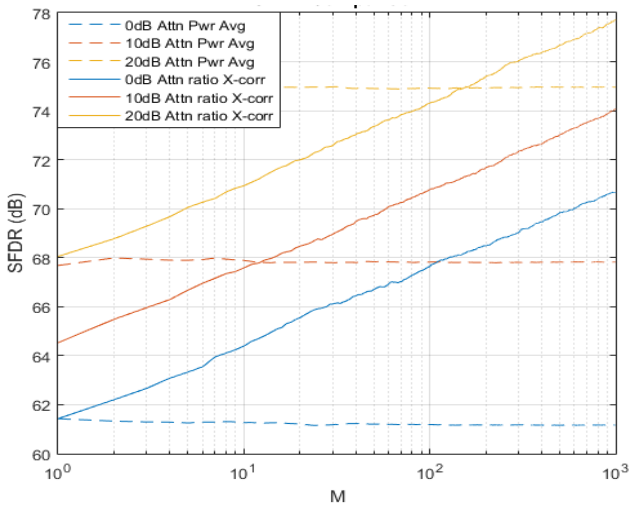

Figure 7: Two inputs with varying attenuation were analyzed (10 Monte Carlo simulations). Dashed lines represent the SFDR from power-averages of the attenuated sensor and achieves the highest single-measurement SFDR. Solid lines represent the cross-correlation between the attenuated

receiver and an un-attenuated receiver. The *M*, where solid and dashed lines intersect is the number of samples needed for cross-correlation to outperform power-averaging of the attenuated sensor.

In order to take advantage of the cluster of sensors available we extend the cross-correlation approach to provide further benefits when more than two sensors observe the same signals (note, they do not need to be calibrated). Consider simple averaging of FFTs of different pairs of sensors. While selectively choosing pairs based on prior or gathered information may lead to improved results as theorized in [9], our approach is to process each combination of two sensors and requires no prior knowledge of sensor distribution or performance. Figure 8 shows SFRD for the case of 0 dB attenuation on all sensors. By increasing the number of sensors, and thus pairs, SFDR can increases. The improvement of adding additional sensors can be considered as adding an extra M_{eff} which is equal to $M_{eff} = (1/2)s^2 - (1/2)s$ (polynomial of s-choose-2), where s is the number of sensors. Processing s sensors in this way increases computational complexity at the same rate that M_{eff} grows but provides a mechanism for trading the number of sensors for sample time (number of points in each measurement, N_{FFT}).

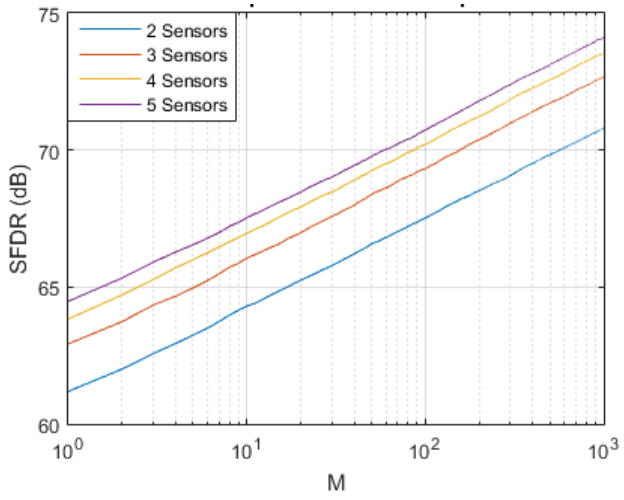

Figure 8: A comparison of SFDR versus *M*, the number of N_{FFT}-point FFTs averaged for 2 to 5 cross-correlated sensors. All sensors have 0 dB of front-end attenuation. (100 Monte-Carlo simulations)

4.0 EXPERIMENTAL RESULTS

The above simulations were validated with the experimental setup shown in Fig. 9. Two tones were used, with one modulated by noise-like data. Note, measurements are similar to simulations but at GHz frequencies. The tones were combined and transmitted wirelessly. On the receiving end, an LNA was used to add nonlinearity to the received signal. The resulting voltages were captured by an oscilloscope and processed as above.

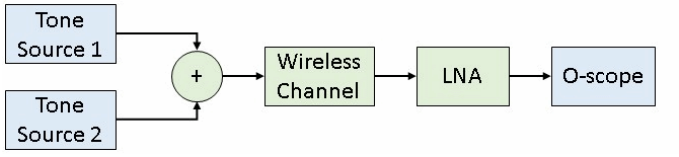

Figure 9: Experimental setup-Tone 1 is 1.471998324 GHz. Tone 2 is 1.143266815 GHz and was amplitude modulated up to 50% by a noise-like sequence. Attenuation was achieved by changing the distance between the transmitter and sensor or adding an attenuator before the nonlinear LNA.

Figure 10 compares SFDR and spectra for a pair of sensors where one sensor is un-attenuated and the other is attenuated by 5 dB. SFRD is compared for power-averaging of the un-attenuated (blue) and attenuated (orange) sensors with cross-correlation (yellow) of both sensors. SFDR (left) shows similar trends to simulations and reveals *(2/3)*5* dB/decade improvement in SFDR versus number of measurements averaged, *M*. The spectra used to compute SFDR are shown in the middle column (for power-averaging FFTs) and the right column (cross-correlation). Power-averaged FFTs show a reduction in variance of the noise floor but the mean noise floor remains constant versus increasing *M*. Cross-correlation, however, shows improvement in the mean over increasing *M*. This is the key advantage of cross-correlation.

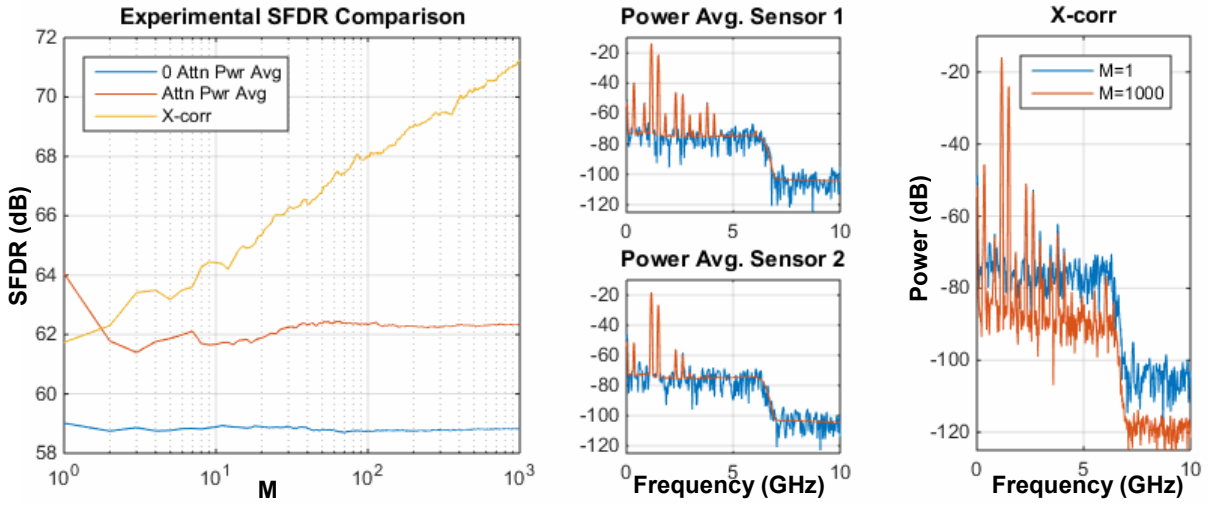

Figure 10: Measurements of 2 sensors where one sensor is attenuated by 5 dB. SFDR trends agree with simulations. The three spectral plots on the right have the same axis and compare a single N_{FFT}-point FFT (blue) and an average of *M=1000* such FFTs (orange). The two power average plots use observations from one sensor at a time. As expected, power averaging only lowers the variance of the noise, while cross-correlation (far right) lowers the noise while preserving signal components.

Next, five different sensors were processed in the same manner as Fig. 8 (that is, cross-correlated spectra were computed for each pair (5-choose-2)) and then averaged versus the number of measurements, *M*. Figure 11 shows the resulting SFDR for sensors 1 through 5 having 0 dB, 5 dB, 10 dB, 15 dB and 15dB of attenuation, respectively. These attenuations were realized through free-space path loss (no fixed attenuators were used). Measurements were collected one at a time (~90 seconds between measurements) verifying that cross-correlation is insensitive to time-delay as long as the conditions they are coarsely time synchronized (defined above as the condition that the signal of interest is present in at least 50% of each time window). Again, the measurement trend is very similar to the simulations and verifying that, in addition to averaging multiple measurements, multiple sensors be used to increase performance. We emphasize that multiple sensors is a means of performing faster scans for a fixed performance level.

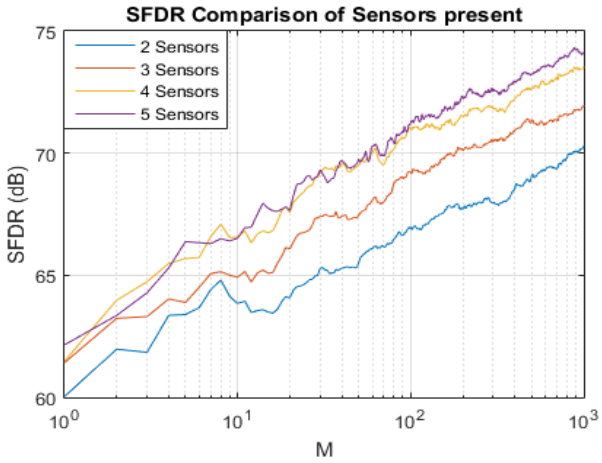

Figure 11: A comparison of SFDR for different numbers of sensors. The experimental setup was used with different antenna positions determining the attenuations. 5 measurement sets were used to imitate 5 separate sensors, and cross-correlation was used to process the data.

5.0 CONCLUSION

We have developed a cross-correlation method for processing multiple distributed sensors in a spectrum sensing sensor network which takes advantage of the unique clustering and low-density of tactically-relevant networks. In particular, we have shown that cross-correlation with a small squad of sensors improves linearity and noise performance by leveraging redundancy. If desired, only the final high-fidelity power spectrum is transmitted over the network backhaul to a global controller which results in a significant reduction in required backhaul capacity. Averaging cross-power spectra improves SFDR by *(2/3)*5* dB/decade of the number of spectra, M. Additionally, multiple sensors can be processed to achieve an $M_{eff}=(1/2)s^2-(1/2)s$, providing a means of improving measurement quality while maintaining high scan rates. Finally, it was shown that attenuation levels of one or more sensors can be used to tune SFDR versus M, and that in the case of distributed sensors, varying free-space path loss automatically realizes such attenuation. Squad-level or network level processing allows for the possibility of switching in additional controlled levels of attenuation to optimize SFDR and scan rate for the particular application or to mitigate a saturated sensor in the presence of a high power emitter.

ACKNOWLEDGEMENTS

Research was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-16-2-0140. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

REFERENCES

[1] N. Kleber, J. Chisum, A. Striegel, B. Hochwald, A. Termos, J. N. Laneman, Z. Fu, J. Merritt, "RadioHound: A Pervasive Sensing Network for Sub-6 GHz Dynamic Spectrum Monitoring," presented at the 2017 IEEE International Symposium on Dynamic Spectrum Access Networks (DySpan), Feb. 2017, Baltimore MD, USA. [url: https://arxiv.org/abs/1610.06212]

[2] R. Tandra and A. Sahai, "SNR Walls for Feature Detectors," in 2007 2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, Apr. 2007, pp. 559–570

[3] R. Tandra and A. Sahai, "SNR Walls for Signal Detection," IEEE Journal of Selected Topics in Signal Processing, vol. 2, no. 1, pp. 4–17, Feb. 2008.

[4] M. S. O. Alink, A. B. J. Kokkeler, E. A. M. Klumperink, G. J. M. Smit, and B. Nauta, "Lowering the SNR Wall for Energy Detection Using Cross-Correlation," IEEE Transactions on Vehicular Technology, vol. 60, no. 8, pp. 3748–3757, Oct. 2011

[5] M. S. O. Alink, E. A. M. Klumperink, M. C. M. Soer, A. B. J. Kokkeler, and B. Nauta, "A 50mhz-To-1.5ghz Cross-Correlation CMOS Spectrum Analyzer for Cognitive Radio with 89db SFDR in 1mhz RBW," in 2010 IEEE Symposium on New Frontiers in Dynamic Spectrum, Apr. 2010, pp. 1–6.

[6] M. S. O. Alink, E. A. M. Klumperink, A. B. J. Kokkeler, W. Cheng, Z. Ru, A. Ghaffari, G. J. M. Wienk, and B. Nauta, "A CMOS spectrum analyzer frontend for cognitive radio achieving +25dbm IIP3 and -169 dBm/Hz DANL," in 2012 IEEE Radio Frequency Integrated Circuits Symposium, Jun. 2012, pp. 35–38.

[7] M. S. O. Alink, E. A. M. Klumperink, A. B. J. Kokkeler, M. C. M. Soer, G. J. M. Smit, and B. Nauta, "A CMOS-Compatible Spectrum Analyzer for Cognitive Radio Exploiting Crosscorrelation to Improve Linearity and Noise Performance," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 59, no. 3, pp. 479–492, Mar. 2012.

[8] J. Briaire and L. K. J. Vandamme, "Uncertainty in Gaussian noise generalized for cross-correlation spectra," Journal of Applied Physics, vol. 84, no. 8, pp. 4370–4374, Oct. 1998.

[9] Y. Zeng and Y. C. Liang, "Eigenvalue-based spectrum sensing algorithms for cognitive radio," IEEE Transactions on Communications, vol. 57, no. 6, pp. 1784–1793, Jun. 2009.