Tracking with MIMO Radar: A Baseline Solution

Richard A. Coogle, John D. Glass, L. Donnie Smith, W. Dale Blair
Georgia Tech Research Institute
7220 Richardson Road, Smyrna, GA 30080
404-407-7963
{rick.coogle, john.glass, larry.smith, dale.blair}@gtri.gatech.edu

Abstract—**This paper describes a baseline target tracking system implemented using the GTRI/ONR Multiple-Input Multiple-Output (MIMO) Radar Benchmark platform. MIMO radar systems have been garnering a significant amount of attention for their potential to improve overall radar performance in comparison to existing systems. While there is much in the current literature regarding the performance and parameter design of MIMO radar target tracking systems, there is little that describes a complete target tracking solution. Such a solution would integrate measurement processing, data assignment, and track filtering into a single unit. The "MIMO tracker" described in this paper aims to provide a starting point for such tracking solutions. Although naïve in some respects, this MIMO tracker provides a comparison tool for new MIMO target tracking algorithms. The results of running the tracker with the scenarios provided in the GTRI/ONR MIMO Radar Benchmark are also presented.**

TABLE OF CONTENTS

1. INTRODUCTION

One of the primary tasks of a radar system is to track a single target or multiple targets. Such a system may be composed of several individual radars, networked together and sharing their information, whether it be their own tracks or sensor measurements, in order to generate a single track picture. However, there is more to building such a system than just sharing track states. More specifically, scheduling of radar resources, measurement processing, data assignment, and track filtering are examples of the components of a complete tracking system.

Multiple-Input-Multiple-Output (MIMO) radars offer promise in regard to improving the performance of target tracking, specifically in their ability to estimate target state by exploiting spatial diversity[1]. Comparisons have been made between MIMO radar and multistatic radars in this regard[2]. Several target detection and localization methods for MIMO radar have also been developed[2][3][4]. Additionally, there has been research done into MIMO radar's ability to mitigate the effects of clutter[5][6]. Since the primary definition of a MIMO radar system being a radar system with widely spaced or colocated antennas transmitting orthogonal waveforms[7], there has been also significant investigation into how one designs these waveforms; [8] and [9] are but two examples. Work has also been done in addressing the data assignment problem[10][11][12].

As for tracking, MIMO radar tracking performance has often been formulated in terms of the Cramér-Rao Lower Bound (CRLB) of the track covariance of a particular system[13][14][15]. The posterior CRLB has also been used in determining how to allocate resources in a MIMO radar system[16]. Track filtering has often been accomplished through the use of sequential Monté Carlo methods[17], i.e. particle filters, or the extended Kalman Filter (EKF)[14]. Particle filters have also been used when combining MIMO techniques with the method of track-before-detect[18].

What is missing in most of the literature is an integration of all of these various techniques and algorithms into a single unit. Complete solutions have been developed[12], but a baseline for comparison aside from comparing MIMO tracking solutions to those of "conventional" phased array radars is conspicuously absent. That is, there needs to be an "apples to apples" comparison. We have developed a MIMO tracking solution using the ONR/GTRI MIMO Radar Benchmark[19] that aims to act as such a baseline. It is not a sophisticated solution by any means, but provides an implementation of each of the components required for a so-called "MIMO tracker". Additionally, it provides a first answer to the challenge problem given in [19], as well as demonstrating how to develop a tracking solution in the MIMO Radar Benchmark.

This paper describes our MIMO tracking solution. It is divided into the following sections: a short overview of the system model in terms of how it considers the MIMO problem, an overview of the tracker's architecture, a description of how radar dwells are scheduled by this tracker, an illustration of sample results as generated by the MIMO Radar Benchmark, and finally, we conclude.

2. MIMO SYSTEM MODEL

There are two primary aspects of the system model to be considered for our solution: what model to use in processing sensor measurements, and the model of what we refer to as a *coordinated MIMO dwell*.

MIMO Radar as Bistatic Radar

We begin by considering a model for MIMO radar measurements. Instead of considering the detailed signal model, which in our case is left to the internals of the MIMO Radar Benchmark, we will simplify by considering MIMO radar as a series of *bistatic pairs*. Each pair will be characterized by a particular orthogonal waveform, and have an associated

---

978-1-4577-0557-1/12/$26.00 ©2012 IEEE.
[1] IEEEAC Paper #1336, Version 2, Updated 1/3/2012.
Work supported by ONR grant N00014-09-1-0054.

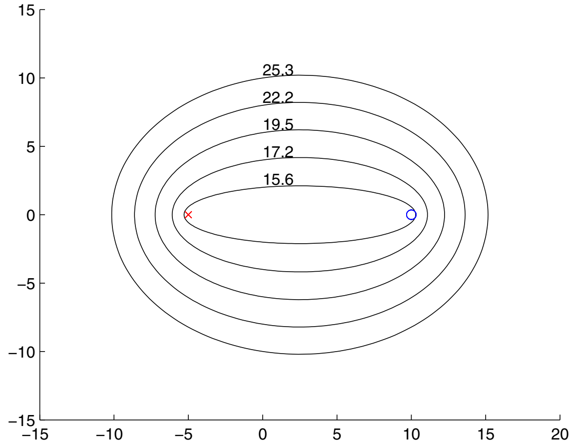

Figure 1. Several bistatic ellipses for a transmitter located at $(x, y) = (-5, 0)$ and a receiver located at $(10, 0)$. The target is moving from $(2, 10)$ to $(10, 10)$ in the positive y direction. The associated bistatic range for each ellipse is indicated on the diagram.

bistatic radar range R_b:

$$R_b = R_T + R_R$$

where R_T is the transmitter to target range and R_R is the target to receiver range, respectively. This is in contrast to considering MIMO as a *multistatic* radar configuration. Multistatic configurations do not require orthogonal waveforms and hence could violate the MIMO radar assumption. One may note that treating MIMO radar as a large bistatic radar system is atypical; however, careful examination of the signal models given in the literature will show that the radar range is usually treated in the same fashion as bistatic radar range, even if that fact is not stated explicitly[2].

The nature of bistatic range causes the generation of so-called *bistatic ellipses* or *contours of constant range* with their foci at the transmitter and receiver locations[20]. Examples of these ellipses are illustrated in Figure 1. This aspect of bistatic range has consequences when considering measurement uncertainty. Usually, a *linearized* form of the measurement covariance is used in order to perform data assignment. This linearized covariance has the shape of an ellipsoid; the true covariance actually has the shape of a contact lens. For bistatic radar, unlike monostatic, this contact lens *bends* along the contour of constant range. Thus, the true covariance may actually be (and often is) smaller in volume than the linearized covariance.

As a final note, instead of the "standard" formulation of bistatic range, we will instead use the alternate formulation:

$$R_b = \frac{R_T + R_R}{2} \tag{1}$$

This equation causes the bistatic range to be equal to the monostatic range if the transmitter and receiver are colocated.

MIMO Dwells and Sensor Tasking

One of the key concepts in the MIMO Radar Benchmark is that of the *coordinated MIMO dwell*. Essentially, this allows several sensors to illuminate a region of space with orthogonal waveforms and jointly obtain all of the bistatic returns. This region of space is characterized by a single point in space which is referred to as a *beam point*.

The MIMO Radar Benchmark allows different sensors to participate in different MIMO dwells, which allows for algorithms that specifically reduce radar energy and dwell time, two of the main parameters in the challenge problem described in [19].

MIMO dwells are initiated in the MIMO Benchmark by populating a *message* with the appropriate parameters, which include the beam point, the waveforms that are to be used, the set of transmitting sensors, and the set of receiving sensors. This message is then sent, and the MIMO Benchmark handles scheduling the appropriate dwell requests on the appropriate sensors and sending the resulting detection events to the MIMO tracker. Note that the MIMO tracker cannot send dwell requests directly to the sensors.

It should be noted that the sensors continue to maintain their own tracks while the MIMO tracker is running. This is to simulate real-world conditions: while sensors may send measurements to a centralized tracker, they will also track targets on their own in order to maintain situational awareness and sensor autonomy. This is especially important in the case of communications loss. Sensors will send their track data to the MIMO tracker, and vice versa. In the latter case, the sensors will attempt to correlate the MIMO tracks with their own tracks. If any correlate, they will overwrite the track states of the correlating tracks with the corresponding MIMO track states. The flow of data in the system is thus summarized by Figure 2.

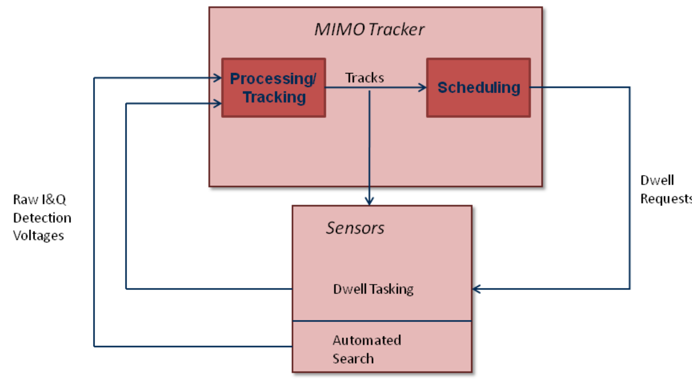

Figure 2. Illustration of the flow of data between the sensors and the MIMO tracker.

3. Track Processing

This section describes the core of the MIMO tracker: i.e. how measurements are processed, assigned to tracks, and used to update existing tracks or create new tracks. Tracks and their associated data are kept in a *track database*, which stores tracks currently being updated, as well as tracks that have been marked as dropped from consideration.

Track processing is invoked each time the scheduler receives all of the events containing the complex I and Q *detection voltages* from the sensors that have participated in a particular

[2] For example, the model given in [18] exhibits this quality.

MIMO dwell, which allows for joint processing of all of the returns. Additionally, this group of events may also contain the results of search dwells generated by individual sensors.

The tracking filter used is an Interacting Multiple Model (IMM) filter [21] with two constant velocity models, abbreviated as "IMM-CVCV". As such, some aspects of how assignment takes place are dependent on the properties of this filter. However, since the IMM-CVCV is a filter that is part of a library of filters that is packaged with the MIMO Radar Benchmark, these aspects are not necessarily of concern, as they are handled automatically. Regardless, we discuss some of them here for completeness.

Measurement Processing

Prior to data assignment and track filtering, the detection voltages must be converted into something usable by those components. Specifically, they must be transformed into position *measurements*. Most of what is required to process measurements into this form is provided as callable subroutines by the benchmark, hence the algorithm will be described here without delving too much into the details behind these functions.

The radars that are used in the MIMO Radar Benchmark are monopulse radars and generate sum-channel and difference-channel (azimuth and elevation) complex voltages; hence, monopulse direction of arrival estimation [22] is used to convert the detection voltages into sine-space *detection primitives*.

Next, the issue of detections that straddle range bins must be addressed. It may be that what appears to be two different detections in two range bins may actually be the *same* detection, if the two detections are sufficiently close together and are sufficiently close to the boundary of their respective range bins. The primitives are thus *clustered* to group similar detections together, and then *centroided* to extract a "true" measurement from the clusters. Centroiding combines the clustered detection primitives such that the variance of the measurement is less than or equal to the variance of the "best" detection primitive.

It is important to note that in some cases, the clustering/centroiding algorithm will be unable to cluster some measurements. These measurements are referred to as *limited measurements* and are given special consideration below.

The resulting measurements are then gated to make sure that dubious measurements are removed from consideration by the tracker. The measurements that remain may now be assigned to tracks.

Data Assignment

Data assigment proceeds by first correcting the measurements for refraction and sensor biases. We must then decide which measurements are to associate with existing tracks or to generate new tracks. If no tracks exist in the track database, then all of the measurements are used to start new tracks. Otherwise, we must first determine which tracks should be considered, in order to prevent spending time attempting to associate measurements with tracks that could not possibly associate simply as a result of the distance between them.

In order to determine which tracks should be used in the assignment process, first the distance of the beam point from each of the tracks is computed. If this distance exceeds a given gate value for a particular track, then that track will not be considered in the data assignment process. If the resulting set of tracks is empty, then all of the measurements will be used to create new tracks.

Next, the costs that are used to assign measurements to targets must be determined. The costs are computed using a log-likelihood approach. First, a Mahalanobis distance d between the track state and a measurement is calculated via the following equation:

$$d(\mathbf{z_k}, \mathbf{x_k}, \mathbf{R_k}, \mathbf{P_k}) = (\mathbf{z_k} - \mathbf{x_k})^{\mathbf{T}} (\mathbf{R_k} + \mathbf{P_k})^{-\mathbf{1}} (\mathbf{z_k} - \mathbf{x_k}) \tag{2}$$

where $\mathbf{z_k}$ and $\mathbf{R_k}$ are the measurement and measurement covariance, respectively; and $\mathbf{x_k}$ and $\mathbf{P_k}$ are the track state and track covariance. The likelihood function

$$L(d) = \frac{1}{\sqrt{|2\pi(\mathbf{P_k} + \mathbf{R_k})|}} \exp\left(-\frac{1}{2} d(\mathbf{z_k}, \mathbf{x_k}, \mathbf{R_k}, \mathbf{P_k})\right) \tag{3}$$

is then used to compute the likelihood for the measurement.[3] This calculation is repeated for each of the measurements and stored in a cost matrix; in the assignment, however, $-\log(L(d))$ is used as the cost value.

Once the costs are computed, a 2d assignment algorithm is applied to assign the measurements to the appropriate tracks. We use a modified Jonker-Volgenant assignment algorithm [23], also referred to as Jonker-Volgenant-Castañon (JVC), in order to perform the assignment. This is one of several assignment algorithms included with the benchmark; e.g. Bertsekas auction [24] and greedy nearest neighbor.

Any measurements that are unassigned are used to initiate new tracks. Note, however, if one of these measurements is a limited measurement, the track is marked as a *limited track*. Also, if there are any limited measurements at all and if any of the measurements in the complete set of measurements actually associated with a track, then we do not initiate a new track based on a limited measurement.

Track Filtering

As mentioned previously, the track filter used is an IMM filter with two constant velocity models. The measurements assigned to the appropriate track in the data assignment step are used to update the track according to the IMM algorithm. However, there are some restrictions on whether a particular track is actually updated:

- The measurement is not limited.
- The measurement is limited, and the last track update was also a limited measurement.
- The track is not a newly initiated track. Updates on initiated tracks are performed during the next resource period.

If these conditions hold, then a given track is updated, and its update count is incremented. This count is then used to determine the *track maturity*, which is used to determine the revisit rate for the track, as well as how many missed updates are required before the track is dropped. There are three levels of maturity: *Infant*, *Tentative*, and *Firm*. Maturity is determined by the number of successful updates of a track. For example, for a track to be declared *Firm*, the track must have been updated at least 20 times. In general, the track confirmation logic is an M out of N rule: i.e. if M successful

[3] $|\cdot|$ represents the matrix determinant.

Table 1. Revisit rates according to track maturity

Infant	0.25 seconds
Tentative	0.5 seconds
Firm	1.0 seconds

updates out of N trials occur, promote the track to the next maturity level.

Missed updates are determined by taking the set difference of the tracks that were updated with all of the tracks in the track database. Any tracks in the difference were the ones that were missed; their miss counts are then incremented.

The revisit rate is then calculated based on the track maturity prior to finishing the track filtering step. For each maturity, the revisit rates are set to the values in Table 1. More specifically, the more mature the track, the greater amount of time between revisits. In the case of a missed update, the revisit time is set to 0.25 seconds after the last sensor resource period. Once the track database is updated with new and updated entries, the track filtering step is completed. This portion of the tracker then becomes idle until another group of detection events is sent from the sensors.

4. Scheduling

In this section, we discuss how the MIMO tracker schedules radar resources and MIMO dwells, and how the MIMO tracker groups events together before they are passed down to the lower processing layers. Note that in all cases, scheduling is coordinated via special events that are placed on the MIMO Benchmark event queue.

Autonomous Search

Search dwells are handled entirely by the sensors. They execute fully autonomous search: the MIMO tracker has no influence on when search dwells are scheduled. The resulting detection events, as stated in Section 3, are sent to the MIMO tracker along with any detection events that were the result of track revisits.

Detection Grouping

In order to group detections, we first *tag* each MIMO dwell request with an identifier. When the individual sensor dwell requests are generated from the MIMO dwell request, the tag is passed down to these dwell requests so that the results of the MIMO dwell may be later grouped together when they are returned to the MIMO tracker.

In order to keep track of which tags have been issued and other properties regarding them, a *tag database* is maintained. Aside from keeping the identifier and whether the tag is currently being used, it stores which sensors are participating in the dwell, which track will be affected by the dwell, and the time at which the MIMO dwell should be processed.

As detection events are returned from the sensors, the MIMO tracker examines the tag that has been attached to each event. It then places the event into a list of events associated with the tag. Once the dwell processing time has passed for a particular tag, the associated set of detection events will be sent to track processing.

Track Revisits

For existing tracks, revisit dwells must be scheduled. First, out of the tracks that are valid in the track database, the revisited tracks are selected based on the revisit time described in Section 3.

Next, for each of the tracks that require a revisit, we determine which sensors should participate in the MIMO dwell. This is straightforward: if the track state lies in the field of view of a given sensor, then that sensor will participate as both a transmitter and receiver. We then extrapolate a beam point from the track state, and select a waveform and pulse width.

Waveforms are selected such that a nominal signal-to-noise ratio (SNR) is maintained for a particular track. For newly formed tracks, waveforms with the maximum number of pulses and power are used. However, the number of pulses is stepped down based on 3dB steps of SNR for each subsequent track update so long as the nominal SNR is maintained. Otherwise, the number of pulses (and waveform power, if applicable) is stepped back up in order to correct for the SNR loss by effectively doubling the output power.

A tag is then assigned to the dwell request, and, finally, the MIMO dwells requests are placed on the request queue.

5. Example Results

In this section, we provide results from running our solution against the test scenarios provided with the benchmark [19]; Table 2 summarizes the active targets in each of the scenarios. Table 3 summarizes the results across all of the scenarios; each of the scenarios was run over 20 Monte Carlo runs, with a runtime occurring during the scenarios starting at 400 seconds and ending at 700 seconds. We can see that, in general, performance of our tracker suffers as the number of targets increases, which is not unexpected. The latter scenarios provide challenges to even the best trackers, in terms of closely-spaced and possibly unresolvable targets.

For Scenario 1, Figures 3, 4, and 5 show plots of the metrics relevant to the challenge problem, i.e. the track completeness ratio, emitted energy, and dwell time. In the case of the latter two, we focus on the performance of Sensor 1. We will go into further detail for Scenario 1, as the performance characteristics will be similar for most of the scenarios. Figures 6, 7, 8, 9, 10, and 11 emphasize this fact when compared with Figures 4 and 5. In the case of the multiple target scenarios, shown in Figures 12, 13, 14, 15, 16, and 17, for the most part reflect the results shown in the single target scenarios at the same time region. However, there is an additional peak near the beginning of the scenario where the targets move into formation, which thus requires more dwell time and energy in order to resolve the targets.

We can see that track completeness remains at 100% except for at a few key points. The spurious track ratio increases at those same points, but the redundant track ratio remains flat. This is indicative of spurious tracks (defined as tracks which do not associate with any of the scenario's truth objects), or tracks being initiated as a result of false alarms. Before we discuss the usage metrics, we emphasize the fact that only track revisit dwells affect this metric; search dwells are not taken into account[19]. As for the emitted energy, in Figure 5, the characteristic peaks and troughs in the plot are the result of the tracking waveform selection algorithm. At the peak, the highest energy waveforms are used by the radar; over time, the energy emitted decreases as the waveforms are stepped down in 3dB increments. We note that there is a sharp peak in the energy emitted and in the dwell times in the 570-600s range. If we examine the output of the benchmark's Track Plotter at that moment in time (Figure 18), we can see that this is where the three targets in the scenario begin to overlap within the sensor's field of view.

For Scenario 7, as shown in Table 3 the track completeness dips significantly compared to the other scenarios. We can attribute this to similar conditions in Scenario 1, i.e. target crossover, and the fact that the targets may be unresolvable, due to being closely spaced and maneuvering. At this crossover, Fighter 3 becomes difficult to track due to the constant revisit rates that the default solution employs, which even results in track loss. Track switches (i.e. points at where a track is associated with a different truth object) also occur between Fighters 3 and 4 before they enter formation. Again, this is due to resolvability issues between targets. Finally, due to the nature of bistatic measurements, the targets may be approaching the midpoint between two sensors, where the bistatic measurement covariance becomes distorted due to singularities in the *r*-*u*-*v* sine-space to Cartesian coordinate transformations. As a result, this causes distortion and inaccuracy in the linearized measurement covariance, which then causes issues in tracking the targets.

While these results expose flaws in this solution, especially in the case of dealing with unresolved targets, we must keep in mind that this is intended to be a baseline for future solutions by other developers and researchers in the MIMO radar community. Hence, improvements such as better resource management, data assigment tuned specifically for the MIMO case, and so on are areas of future research.

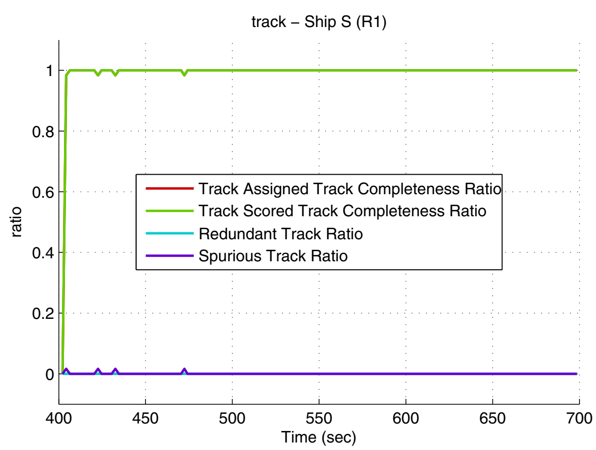

Figure 3. Plot of the track ratios; specifically the track completeness (Scenario 1).

Table 2. MIMO Radar Benchmark scenarios

Scenario	Active Targets
1	Fighter 1, Airliners A and C
2	Fighter 2, Airliners A and C
3	Fighter 3, Airliners A and C
4	Fighter 4, Airliners A and C
5	Fighters 1 and 2, Airliners A and C
6	Fighters 3 and 4, Airliners A and C
7	Fighters 1, 2, 3, and 4, Airliners A and C

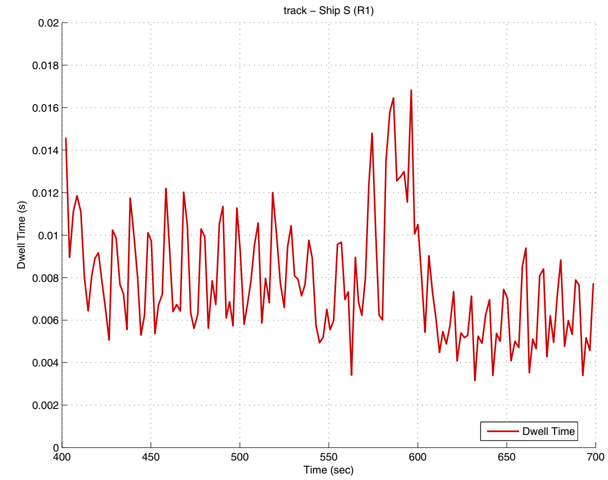

Figure 4. Plot of Sensor 1's dwell times (Scenario 1).

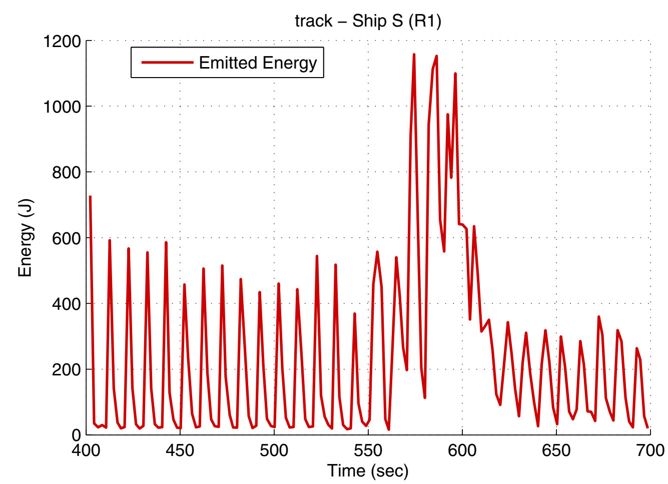

Figure 5. Plot of the energy emitted by Sensor 1 (Scenario 1).

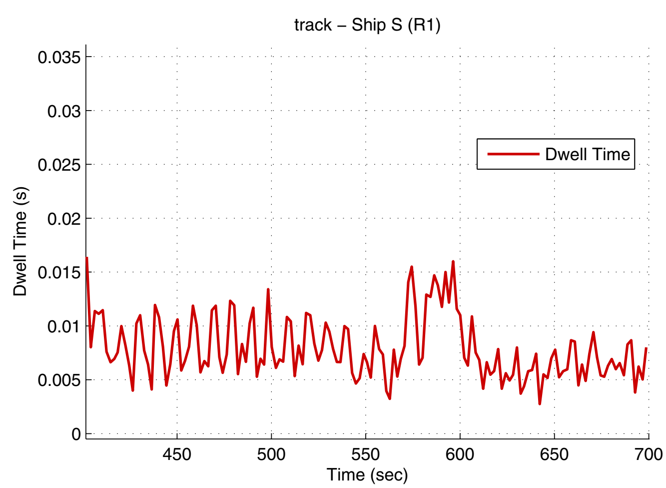

Figure 6. Plot of Sensor 1's dwell times (Scenario 2).

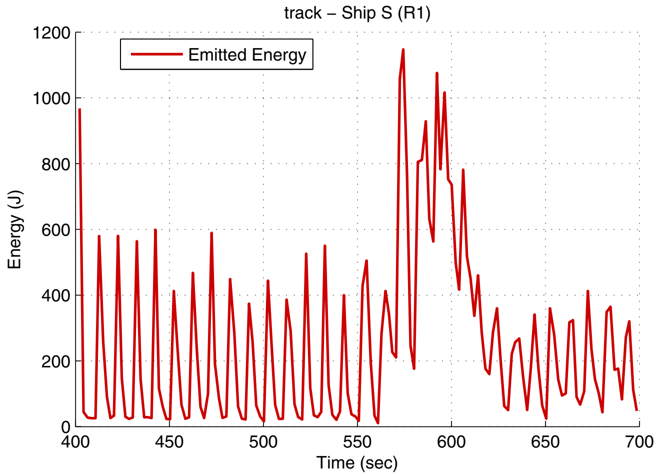

Figure 7. Plot of the energy emitted by Sensor 1 (Scenario 2).

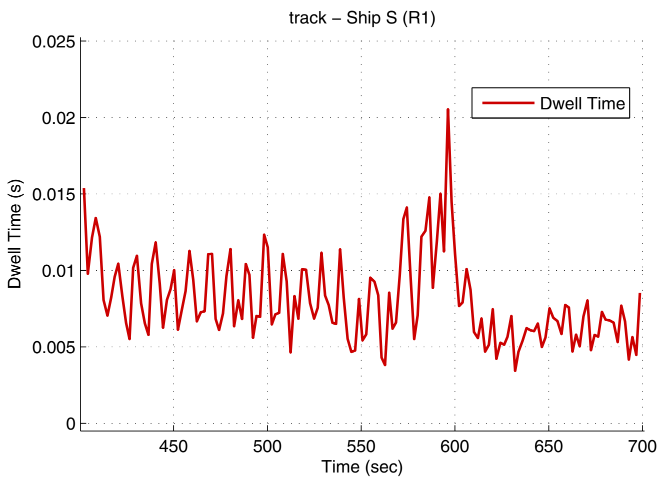

Figure 8. Plot of Sensor 1's dwell times (Scenario 3).

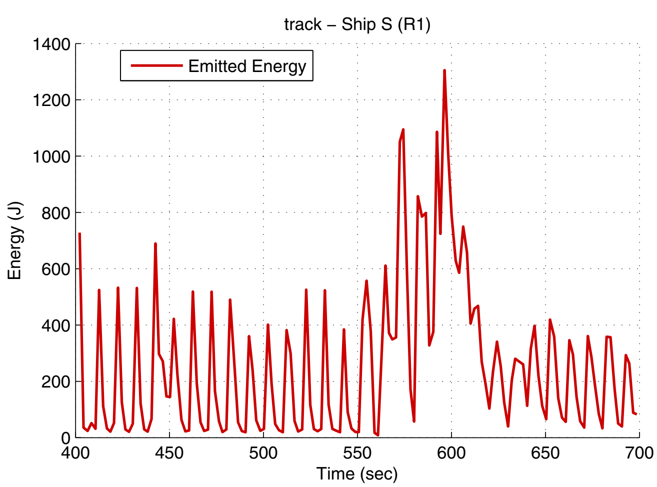

Figure 9. Plot of the energy emitted by Sensor 1 (Scenario 3).

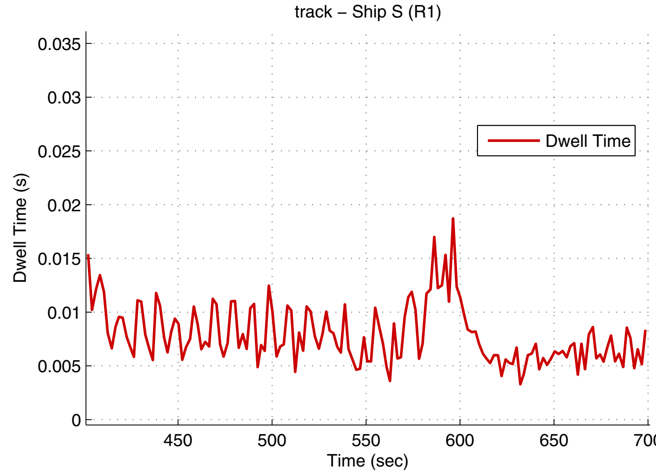

Figure 10. Plot of Sensor 1's dwell times (Scenario 4).

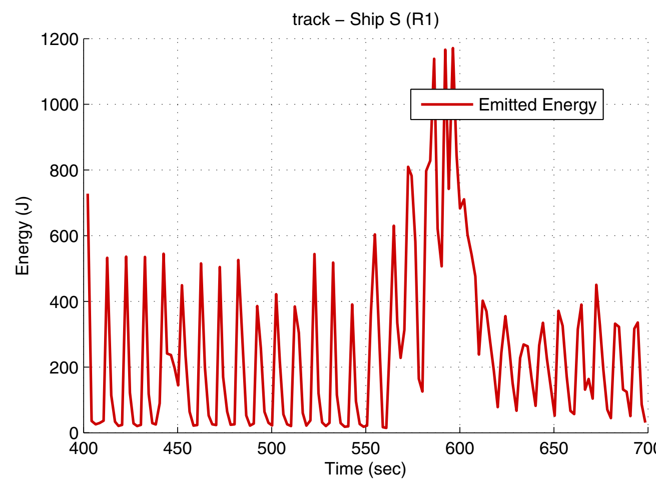

Figure 11. Plot of the energy emitted by Sensor 1 (Scenario 4).

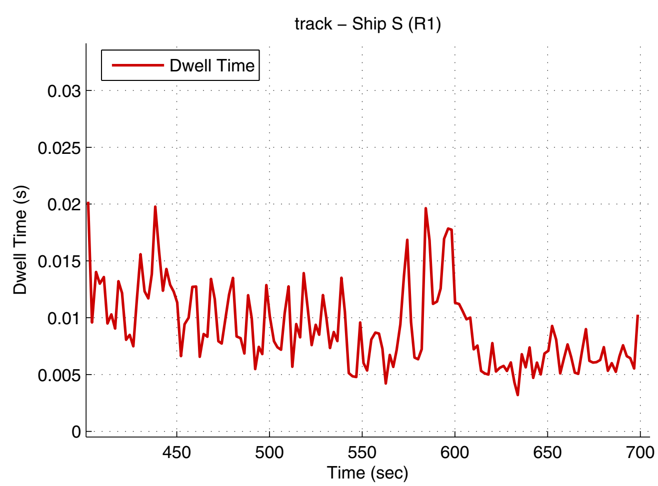

Figure 12. Plot of Sensor 1's dwell times (Scenario 5).

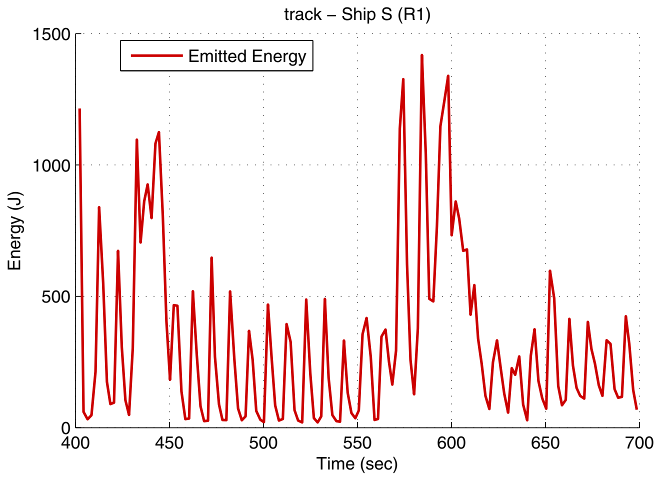

Figure 13. Plot of the energy emitted by Sensor 1 (Scenario 5).

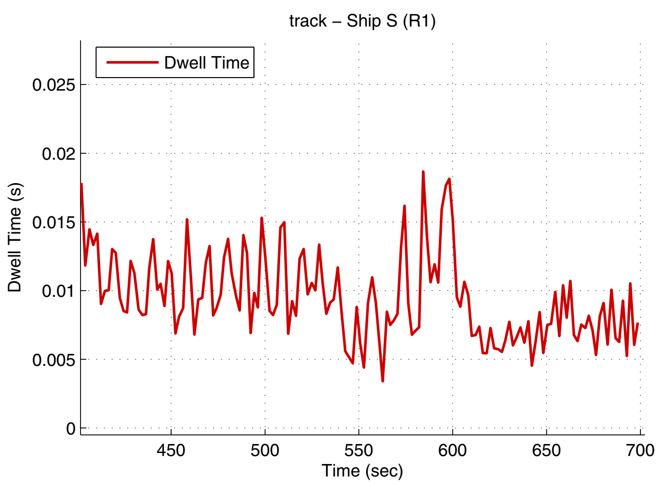

Figure 14. Plot of Sensor 1's dwell times (Scenario 6).

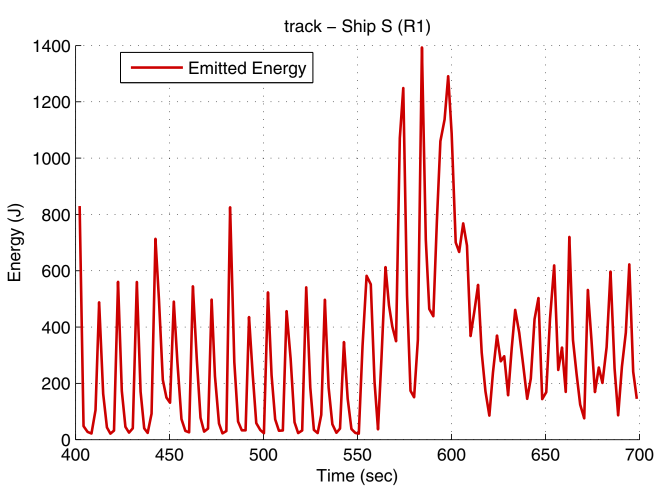

Figure 15. Plot of the energy emitted by Sensor 1 (Scenario 6).

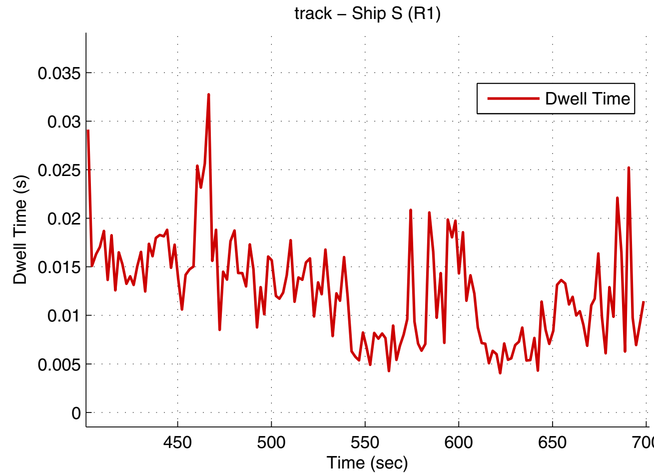

Figure 16. Plot of Sensor 1's dwell times (Scenario 7).

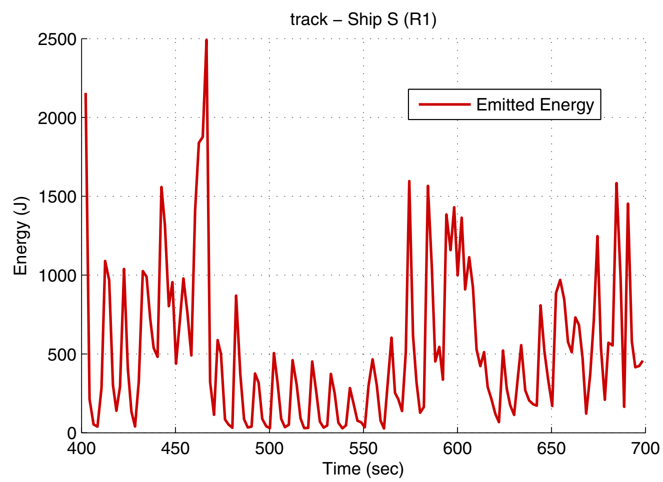

Figure 17. Plot of the energy emitted by Sensor 1 (Scenario 7).

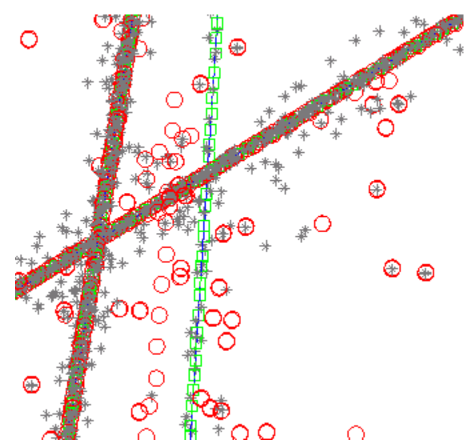

Figure 18. Screen capture of the Track Plotter showing the point of overlap of the three targets with respect to Sensor 1. The circles are the beam points, the stars are measurements, and the squares represent the track. Sensor 1 is offscreen, in the lower right.

Table 3. MIMO Tracker results

Scenario	Average Completeness Ratio	Total Energy (kJ)	Total Dwell Time (s)
1	0.992	3004	83.382
2	0.996	3214	85.268
3	0.991	3245	86.277
4	0.991	3156	84.785
5	0.987	4143	99.391
6	0.912	4118	103.145
7	0.865	6526	138.478

6. CONCLUSION

We have presented a baseline MIMO tracking solution implemented using the ONR/GTRI MIMO Radar Benchmark. This solution demonstrates some of the challenges involved in creating a complete MIMO radar tracking solution, while providing a point of comparison for other solutions.

Future work includes examining the data association problem more closely; i.e. which approaches would be more effective in the MIMO environment, as well as any additional considerations for tracking using bistatic measurements. The latter is brought into consideration due to the fact that as targets approach certain regions of the bistatic ellipse, specifically the areas between the sensors in the bistatic pair, the measurement model covariance will become distorted with respect to the sensors. Effectively dealing with this distortion is still an open issue.

REFERENCES

[1] Q. He, R. Blum, and A. Haimovich, "Non-coherent MIMO radar for target estimation: More antennas means better performance," in *Proceedings of the 43rd Annual Conference on Information Sciences and Systems. CISS 2009*, Mar 2009, pp. 108–113.

[2] A. A. Gorji, R. Tharmarasa, and T. Kirubarajan, "MIMO vs. multistatic radars for target localization," in *Proceedings of Signal and Data Processing of Small Targets*, vol. 7445, no. 1. SPIE, 2009, p. 744502.

[3] M. Jiang, J. Huang, and Y. Hou, "MIMO radar joint estimation of target location and velocity with multiple subcarrier signals," in *Proceedings of the IEEE International Symposium on Phased Array Systems and Technology (ARRAY)*, Oct 2010, pp. 398–402.

[4] A. Tajer, G. Jajamovich, X. Wang, and G. Moustakides, "Optimal joint target detection and parameter estimation by MIMO radar," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 1, pp. 127–145, Feb 2010.

[5] V. Mecca, D. Ramakrishnan, and J. Krolik, "MIMO radar space-time adaptive processing for multipath clutter mitigation," in *Proceedings of the Fourth IEEE Workshop on Sensor Array and Multichannel Processing*, July 2006, pp. 249–253.

[6] A. Sheikhi and A. Zamani, "Adaptive target detection in clutter using MIMO radar," in *Proceedings of the International Radar Symposium. IRS 2006.*, May 2006, pp. 1–4.

[7] J. Li and P. Stoica, *MIMO Radar Signal Processing*. IEEE-Wiley Press, 2008.

[8] J. Zhang, B. Manjunath, G. Maalouli, A. Papandreou-Suppappola, and D. Morrell, "Dynamic waveform design for target tracking using MIMO radar," in *Conference Record of the Forty-Second Asilomar Conference on Signals, Systems and Computers*, Oct 2008, pp. 31–35.

[9] B. Manjunath, J. Zhang, A. Papandreou-Suppappola, and D. Morrell, "Sensor scheduling with waveform design for dynamic target tracking using MIMO radar," in *Conference Record of the Forty-Third Asilomar Conference on Signals, Systems and Computers*, Nov 2009, pp. 141–145.

[10] C. Kabakchiev, I. Garvanov, L. Doukovska, V. Kyovtorov, and H. Rohling, "Data association algorithm in multiradar system," in *Proceedings of the IEEE Radar Conference. RADAR'08*. IEEE, 2008, pp. 1–4.

[11] A. A. Gorji, R. Tharmarasa, and T. Kirubarajan, "An assignment based algorithm for multiple target localization problems using widely-separated MIMO radars," in *Proceedings of Signal Processing, Sensor Fusion, and Target Recognition XIX*, vol. 7697, no. 1. SPIE, 2010, p. 769702.

[12] A. Gorji, R. Tharmarasa, and T. Kirubarajan, "Tracking multiple unresolved targets using MIMO radars," in *Proceedings of the 2010 IEEE Aerospace Conference*, Mar 2010, pp. 1–14.

[13] Q. He and R. Blum, "Cramer-Rao bound for MIMO radar target localization with phase errors," *Signal Processing Letters*, vol. 17, no. 1, pp. 83–86, Jan 2010.

[14] H. Godrich, V. Chiriac, A. Haimovich, and R. Blum, "Target tracking in MIMO radar systems: Techniques and performance analysis," in *Proceedings of the 2010 IEEE Radar Conference*, May 2010, pp. 1111–1116.

[15] H. Godrich, A. Haimovich, and R. Blum, "A MIMO radar system approach to target tracking," in *Conference Record of the Forty-Third Asilomar Conference on Signals, Systems and Computers*. IEEE, 2009, pp. 1186–1190.

[16] J. D. Glass and L. D. Smith, "MIMO radar resource allocation using posterior Cramér-Rao lower bounds," in *Proceedings of the 2011 IEEE Aerospace Conference*, Mar 2011, pp. 1–9.

[17] S. Sen and A. Nehorai, "OFDM MIMO radar design for low-angle tracking using mutual information," in *Proceedings of 3rd IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, Dec 2009, pp. 173–176.

[18] B. Habtemariam, R. Tharmarasa, and T. Kirubarajan, "Multitarget track before detect with MIMO radars," in *Proceedings of the 2010 IEEE Aerospace Conference*, Mar 2010, pp. 1–9.

[19] R. A. Coogle, J. D. Glass, L. D. Smith, P. Miceli, A. Register, P. West, and W. D. Blair, "A MIMO radar benchmarking environment," in *Proceedings of the 2011 IEEE Aerospace Conference*, Mar 2011, pp. 1–10.

[20] N. Willis, *Bistatic Radar*. SciTech Publishing, 2005.

[21] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. New York: John Wiley & Sons, 2001.

[22] W. D. Blair and M. Brandt-Pearce, "Monopulse DOA

estimation for two unresolved rayleigh targets," *IEEE Transactions on Aerospace Electronic Systems*, vol. AES-37, no. 2, pp. 452–469, April 2001.

[23] R. Jonker and A. Volgenant, "A shortest augmenting path algorithm for dense and sparse linear assignment problems," *Computing*, vol. 38, no. 4, pp. 325–340, 1987.

[24] D. Bertsekas, *Linear Network Optimization*. Cambridge, MA: MIT Press, 1991.

BIOGRAPHY

Richard A. Coogle is a PhD student in the field of electrical and computer engineering at the Georgia Institute of Technology, as well as a Graduate Research Assistant with the Georgia Tech Research Institute. He received his BS in computer engineering (2003) and MS in electrical and computer engineering (2009) from the Georgia Institute of Technology. Prior to returning to Georgia Tech to pursue a PhD and joining GTRI in 2010, he worked for Lockheed Martin Aeronautics, developing simulation and embedded systems software, and for Northrop Grumman Corporation, designing and developing a simulation software platform called the Dynamic Test Stand (DTS). While at Northrop Grumman, Mr. Coogle also led the development of an automated test infrastructure to be integrated with DTS. At GTRI, he is one of the primary developers of the ONR/GTRI MIMO Benchmark. His research interests include target tracking, field robotics, behavior-based robotics, task assignment algorithms, and determining optimal paths for moving a virtual character across a computer screen.

John D. Glass is a PhD student at Georgia Tech. He received the B.S.E.E from the University of Tennessee in Knoxville in May 2009 (Summa Cum Laude) and the M.S.E.C.E. from Georgia Tech in December 2010. He has worked several internships with Honeywell Intl. in Clearwater, FL, working on embedded GPS/INS systems, as well as the Y-12 National Security Complex in Oak Ridge, TN, working with amorphous wires. Currently he is a graduate research assistant at GTRI, as a developer for the GTRI/ONR MIMO Radar Benchmark. His current interests lie in the general field of digital signal processing and target tracking.

L. Donnie Smith is a Research Engineer with the Georgia Tech Research Institute. He received the B.S. degree in Computer Engineering (Summa Cum Laude) from Georgia Tech in 2002, and from June 2003 to August 2005 worked for Northrop Grumman Corporation, designing simulation and test stand architectures. After going back to graduate school in 2005, he accepted a fulltime position at GTRI in February 2007, shortly before earning his M.S. degree in Electrical and Computer Engineering, also from Georgia Tech. As a member of the Georgia Tech research faculty, Mr. Smith has contributed to projects in the areas of target tracking, signal processing, and radar simulation, and these efforts were recognized by promotion to Research Engineer II in July of 2009. He is currently working towards a PhD at Georgia Tech in Electrical and Computer Engineering.

W. Dale Blair is a Principal Research Engineer at the Georgia Tech Research Institute in Atlanta, GA. He received the BS and MS degrees in electrical engineering from Tennessee Technological University in 1985 and 1987, and the Ph.D. degree in electrical engineering from the University of Virginia in 1998. From 1987 to 1990, he was with the Naval System Division of FMC Corporation in Dahlgren, Virginia. From 1990 to 1997, Dr Blair was with the Naval Surface Warfare Center, Dahlgren Division (NSWCDD) in Dahlgren, Virginia. Dr Blair is internationally recognized for conceptualizing and developing benchmarks for comparison and evaluation of target tracking algorithms. Dr Blair developed NSWC Tracking Benchmarks I and II and originated ONR/NSWC Tracking Benchmarks III and IV NSWC Tracking Benchmark II has been used in the United Kingdom, France, Italy, and throughout the United States, and the results of the benchmark have been presented in numerous conference and journal articles. He joined the Georgia Institute of Technology as a Senior Research Engineer in 1997 and was promoted to Principal Research Engineer in 2000. Dr Blair is co-editor of the Multitarget-Multisensor Tracking: Applications and Advances III. He has coauthored 22 refereed journal articles, 16 refereed conference papers, 67 papers and reports, and two book chapters. Dr Blair's research interest include radar signal processing and control, resource allocation for multifunction radars, multisensor resource allocation, tracking maneuvering targets, and multisensor integration and data fusion. His research at the University of Virginia involved monopulse tracking of unresolved targets. Dr Blair is the developer and coordinator of the short course Target Tracking in Sensor Systems for the Distance Learning and Professional Education Department at the Georgia Institute of Technology. Recognition of Dr Blair as a technical expert has lead to his election to Fellow of the IEEE, his selection as the 2001 IEEE Young Radar Engineer of the Year, appointments of Editor for Radar Systems, Editor-In-Chief of the IEEE Transactions on Aerospace and Electronic Systems (AES), and Editor-in- Chief of the Journal for Advances in Information Fusion, and election to the Board of Governors of the IEEE AES Society,1998-2003, 2005-2007, and Board of Directors of the International Society of Information Fusion.