REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
6/21/95

3. REPORT TYPE AND DATES COVERED
Phase I Final Report 9/23/94-6/30/95

4. TITLE AND SUBTITLE
Offboard Sensor Correlation for the E-3 AWACS Final Report

5. FUNDING NUMBERS
CF19628-94-C-0085

6. AUTHOR(S)
Dr. C. A. Butler
Dr. J. H. Discenza
Richard S. Thompson

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Daniel H. Wagner Associates, Inc.
Station Square Two
Paoli, PA 19301
(610-644-3400)

8. PERFORMING ORGANIZATION REPORT NUMBER
6040PhI

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
ESC/PKR Capt Patricia Pirowski (ESC/AWD)
Air Force Materiel Command, USAF
104 Barksdale Street
Hanscom AFB, MA 01731-1806

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

This report describes the results of a Phase I Small Buisness Innovation Research (SBIR) program performed for USAF Electronics Systems Center. Under a previous SBIR program, Wagner Associates developed a multi-sensor integration (MSI) algorithm designed to integrate the data received from several onboard sensors into a single tactical picture for display to an AWACS operator. The overall objective of this Phase I SBIR effort was to enhance the previously existing software in order to include both offboard data and operator input. This objective was successfully met and the new version of the MSI algorithm is currently installed in MITRE's Fusion Evaluation Testbed at Hanscom AFB.

19991005 169

14. SUBJECT TERMS
MUSSLE, Off Board, AWACS, Correlation, Data Fusion, ID Fusion

15. NUMBER OF PAGES
10

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT
UNCLASSIFIED

18. SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

19. SECURITY CLASSIFICATION OF ABSTRACT
UNCLASSIFIED

20. LIMITATION OF ABSTRACT
UL

NSN 7540-01-280-5500
F19628-94-C-0085
Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std Z39-18
298-102

DTIC QUALITY INSPECTED 4

Attachment No. 1

Case: 6040

OFFBOARD SENSOR CORRELATION FOR THE E-3 AWACS

Final Report

Contract Number F19628-94-C-0085

June 1995

Prepared for

Electronic Systems Center/PKR
Air Force Material Command, USAF
104 Barksdale Street
Hanscom AFB, MA 01731-1806

In Accordance With ANSI Z39.18-1987

By

Dr. C. Allen Butler
Dr. Joseph H. Discenza
Mr. Richard S. Thompson

Daniel H. Wagner Associates, Inc.
2 Eaton Street, Suite 500
Hampton, Virginia 23669

Table of Contents

Summary

This report describes the results of a Phase I Small Business Innovation Research (SBIR) program performed for Electronics Systems Center (ESC). Daniel H. Wagner Associates, Inc. enhanced an existing software program designed to perform multiple sensor, multiple target tracking for the E-3 AWACS. The specific enhancements made to the software include: incorporation and correlation of offboard sensor data; development of a data registration algorithm; responsiveness to AWACS operator commands; and recommendation of target identification. The enhanced software has been installed in MITRE's Fusion Evaluation Testbed and is currently undergoing evaluation by MITRE personnel and AWACS operators.

1. Introduction

In the air-intensive strategic battle of the 1990's and beyond, the E-3 AWACS will be required to handle information from off board sources and from additional onboard sensors to provide complete and accurate all-source, high confidence identification. Additionally in some hostile environments, it will be advantageous to operate for long periods of time in passive mode, when emissions from radar or from both radar and IFF are restricted. These emerging criteria will place stringent requirements on the E-3's real-time information processing. In high track density environments, the present method of manual correlation of data will not be practical. Developing computer solutions to this all-source fusion will be a significant challenge and, if successful, should lead to a common, reusable Ada library of correlation and fusion software that can be applied to other Air Force platforms.

Daniel H. Wagner Associates, Inc., under a previous SBIR program, has been singularly successful in developing and implementing multi-sensor integration (MSI) algorithms that solve the problem of combining information from on-board sensors into a single tactical picture. That software has been subjected to a ground breaking test and evaluation process in ESC's Fusion Evaluation Testbed. In a test and evaluation debrief given on December 14, 1993, the results showed the Wagner MSI algorithm (hereinafter referred to as MUSSLE — Multiple Sensor Statistical Likelihood Estimator) to be very effective in automatically developing tracks from multiple sensors and to outperform the current AWACS tracker in maintaining track, especially on maneuvering targets. In the meantime, Wagner Associates has been conducting other research in the area of data fusion which has resulted in useful products that can be applied in the AWACS MSI prototype.

1.1. Benefits

Proper design and implementation of recently developed data fusion concepts will result in more efficient utilization of AWACS operators, sensors, computing, and communications resources. Specific advantages include: high confidence, all-source track ID; the ability to accomplish smooth, seamless cross-tell of all tracks between E-3s; reduced risk to own forces through confusion, loss of continuous tracking of hostile and friendly tracks, and data overload of a single AWACS platform; an AWACS-supplied tactical picture that is free of platform-to-platform discontinuities, missing tracks, and dual designations; and precise cueing of tactically significant targets, through multi-source tracking, resulting in more effective weapons use. The product of this Phase I research is *reusable Ada software* which, after the next round of testing and evaluation in the FET, will become a candidate for an Air Force-wide common fusion library. In Phase II, we will address the important issues of (1) real-time processing architecture, (2) hardware requirements and (3) interface with actual sensor data streams on the E-3 for in-flight demonstrations.

1.2. Background

The current automated methods of managing track data in AWACS are inadequate. They do not use all sources to provide track ID or position information on tracks. The information transmitted on the link may not accurately represent the actual target position. The track file is represented by a graphic symbol that the operator can control, including the ability to "park" the graphic symbol by moving it away from the track. The present algorithm is unable to distinguish parking from valid track updates and therefore reports the parked position on the net, resulting in misleading data. Also, because the present tracking algorithm is incapable of maintaining adequate tracking during target maneuvers, the track file can become highly inaccurate. Only the smoothed track is transmitted on the net with no indication of validity and therefore operators on other platforms (combat aircraft and other E-3s) cannot rely on the data.

2. Phase I Progress and Results

The Phase I objectives, as outlined in the proposal, were to permit the current prototype algorithm to run in the new, enhanced FET, with added sensors and off board sources, and to provide deliverable software for evaluation as a potential for an Air Force-wide common fusion library. The specific objectives were:

- Add new sensors and off board link input processing to the basic algorithm.
- Implement an interim gridlock algorithm to permit registration of position, azimuth, and time latency from off board data sources.
- Provide interfaces to accept and respond to a limited set of operator commands from the air force's prototype user system interface (USI).
- Replace the current FORTRAN sensor-to-track algorithm with a newer version, in Ada.
- Implement a new and sophisticated passive tracking algorithm to provide the ability to determine range accurately from passive sensors only.

Because of changes in the sensor models within the FET, we were required to make significant modifications to the existing software. This additional tasking prevented us from accomplishing all of our original goals. Specifically, we were able to meet the first four objectives outlined above, but were unable to implement the passive tracking algorithm. In addition to the proposed objectives described above that were accomplished, we also implemented logic to make ID recommendations based on a variety of information. In the remainder of this section, we detail the accomplishments of our Phase I research.

2.1. Sensors And Off board Link Input Processing

The modular design of the current correlation algorithm allowed us to incorporate the data from new sensors and off board sources with relative ease. Our architecture provides for separate error detection and correction modules for each individual sensor. For the remote sensor data, we relied on the track number provided and only used a gross error detection to assure that the data was not garbled or a track switch had not occurred. In addition, we used different error statistics for the position and velocity based on whether or not the data was reported by ownship. The height source also resulted in different estimates of accuracy for the reported altitude. When no height source was available, a two-dimensional rather than the usual three dimensional Kalman filter process is used for tracking.

2.2. Interim Gridlock Algorithm

In order to utilize off board data, the prototype version of MUSSLE must contain a module which compensates for biases (also called registration errors) between sensors and platforms. The reason is that, although sensor data are relatively accurate (e.g., range and bearing measurements from search and track radars, bearing measurements from ESM, and bearing and elevation data from IRST), they must first be corrected to a standard frame of reference in which the positions of the reporting platforms are known. (In Link-11, this is a Cartesian coordinate system in nautical miles with a local origin called Data Link Reference Point (DLRP); in JTIDS it is latitude and longitude (WGS-72).) Unfortunately, the inaccuracies of the reporting unit (RU) position measurements with respect to the standard coordinate system are such that they contaminate the otherwise accurate target position measurements. Recent advances in navigation (GPS) have improved the position measurement situation but have not solved the bias problem because of inadequacies in platforms' combat sensor collection, data management systems, and network configurations.

There are three important components of report bias: position, azimuth, and time. Position bias occurs because of improper operation of navigation equipment or combat direction systems on reporting platforms. Azimuth biases are generated from compass deviation. Time biases are introduced by unavoidable delays between the actual sensing of the target (e.g., radar antenna pointing at the target) and receipt of the contact report in the remote platform's correlation algorithm. All three components of bias must be estimated simultaneously, otherwise incorrect answers will be obtained. For the Phase I effort, we ignored the time component of bias, and used a simple, least-squares approach to estimating the positional components of bias.

2.2.1. Position and Azimuth Biases

We begin by describing a model with constant biases in position and azimuth (no time bias). The coordinate system we use is x,u positive east; y,v positive north; and angles positive clockwise from north. Figure 1 shows a single observation by reporting unit i (RU_i) and the bias error components. In this diagram u_i and v_i represent the target's position relative to the observer and "true" is taken to mean accurate relative to the reference platform.

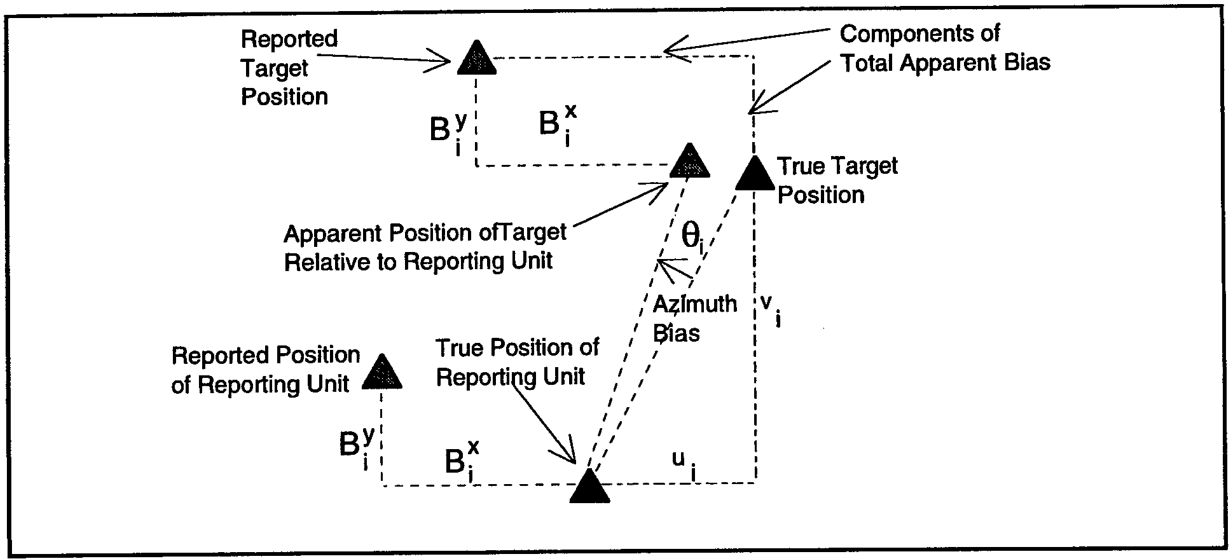

Figure 1. Position and Azimuth Components of Observation Bias in the Gridlock Problem.

If $[x^*\ y^*]^T$ denotes the true position of the target, the biased report, without random observation noise, is

$$\begin{bmatrix} x^* \\ y^* \end{bmatrix} + \begin{bmatrix} \cos(\theta_i) & \sin(\theta_i) \\ -\sin(\theta_i) & \cos(\theta_i) \end{bmatrix} \begin{bmatrix} u_i \\ v_i \end{bmatrix} + \begin{bmatrix} B^x_i \\ B^y_i \end{bmatrix} - \begin{bmatrix} u_i \\ v_i \end{bmatrix} \tag{1}$$

where $\vec{B}_i = \left[B^x_i,\ B^y_i\right]^T$ is the position bias and θ_i is the azimuth bias.

2.2.2. Data Management for the Interim Algorithm

In order to develop the matrices for the solutions of the estimation equations generated by (1), MUSSLE will need to select high confidence mutual observations for which to collect data. Of course, if there are significant unestimated biases, it may be difficult for the algorithm to develop a correlation solution that will identify those mutual observations. In order to permit the interim implementation to obtain a starting gridlock solution, we arbitrarily increase the observation error parameters for each off board sensor until the first estimation process has been completed. Once the bias estimates are applied, the error parameters are restored to their normal values for further processing of the reports. We chose to only look at correlations between remote tracks and locally held radar tracks. (The other sensors were used in determining if a given correlation was valid, but the estimate of bias was based on the radar track position only.)

The biases need to be re-estimated at regular intervals, to account for drift and occasional jumps in the bias values. In our interim solution, we use a Kalman filter approach which automatically accounts for the "newness" of the more recent observations.

2.3. Operator Commands from USI

There are rudimentary operations in the user system interface (USI) that require notification to and action by MUSSLE. ESC has implemented these in the FET and created additional records in the FORTRAN common blocks that provide this data to MUSSLE. When the FET is operating in interactive mode with an operator in the loop, MUSSLE *automatically* initiates tracks only in the ATI regions.

The basic operator functions that we implemented were Initiate Track, Drop Track, and Reinitiate Track. These functions are described below:

2.3.1. Initiate Track

The FET can be run in either Batch or Interactive mode. MUSSLE continues to perform its usual tracking and correlation functions independent of the FET's mode. However, when the FET is run in interactive mode, MUSSLE does not communicate all of its tracks to the FET. Each track in the database is marked with a flag that indicates whether or not the track is currently being displayed to the operator. The process of changing a track from the status of "Not Displayed" to "Displayed" is then equivalent to initiating a track. MUSSLE only initiates tracks that first appear within the Automatic Track Initiation (ATI) regions or that are designated by the operator. When MUSSLE receives a command from the operator to initiate a track, an estimate of the track's current position and velocity is provided by the operator. MUSSLE finds the track in its database that is (a) not already displayed to the operator and (b) closest statistically to the given position and velocity. This track is then marked as Displayed and the usual information (tracking and ID data) is provided each scan to the FET.

It is possible that the "closest" track in the database is so far from the given position that it is unreasonable to make the association between the "initiate track" command and the "closest" track. In this case, a "dummy" track is created in MUSSLE's database and dead-reckoned using the velocity specified by the operator. Because MUSSLE was unable to associate a "real" track, this dummy track is maintained forever until the operator either reinitiates or drops the track. The philosophy here is that the operator is always right. If the operator wants to see a track at a given position, it shall be displayed even if MUSSLE is unable to associate any sensor data with this track.

2.3.2. Drop Track

In the situation where an operator can see a harmless track moving out of the surveillance area or where a track does not appear to be associated with any sensor data, the operator may wish to command MUSSLE to drop a track. MUSSLE will continue to process reports and correlate tracks even after they have been dropped by the operator. The only difference will be that a flag will be set in the track record to prevent reporting or display of the dropped track. If the operator subsequently commands MUSSLE to initiate a track that has been previously dropped, all the state and tracking data will still be in MUSSLE's database and reporting can resume instantly.

2.3.4. Reinitiate Track

If an operator determined that a track that he previously initiated or that was automatically initiated by MUSSLE is no longer associated with the desired sensor data, the operator may wish to command MUSSLE to reinitiate a given track. In this instance, MUSSLE will set the flag on the current track to "Not Displayed" and will follow the same procedure as described above when a track is initiated to find the "closest" track in the database to the newly designated position and velocity.

2.4. MATCH Ada Version

The previous FET implementation of MUSSLE contained an older version of the MATCH algorithm, coded in FORTRAN. Over the past three years, work has been ongoing to rewrite the MATCH algorithm in Ada. This effort is now complete and we have recently implemented the Ada version in a Navy correlator called the Global Correlation Engine (GCE). It has been tested and run in a technology demonstration sponsored by the Naval Sea Systems Command at the Wallops Island Naval Warfare Systems Center Dahlgren Division (NSWC-DD) prototype laboratory. The project was called Integrated Interior Command and Control, or $(IC)^2$. In this project a multiple-network simulation of a shipboard information system was set up in a ship-like laboratory, with multiple feeder networks connected to a high-speed fiber optic backbone. The GCE was established in one of these feeder networks and all internal and external sources of target contact data were then collected in one processing node and fed into the GCE. As many as twelve separate sources of data were successfully correlated in real-time, including radar, IFF, ESM, and simulated external links. The external links were simulated in order to operate the demonstration in unclassified mode. As a result, no gridlock was necessary for the off board data.

The FORTRAN version of the MATCH code was successfully replaced by the Ada version. The result is that MUSSLE is entirely written in Ada.

2.5. ID Recommendation

MUSSLE attempts to make recommendations of Target ID to the operator. These recommendations are based on a variety of sensor and kinematic input. For example, if a correlated track is using data from an IFF sensor with confirmed mode 4 responses, an ID of FRIEND is recommended to the operator. Similar recommendations are made based on ESM contacts and knowledge of the ESM emitter database and on the hypothetical sensors. Additionally, ID recommendations are made based on heuristics associated with ID By Origin (IDBO) regions and ID Gates. The combination of various ID information is performed using a simple rule-based logic.