An Assessment of Track Fusion Algorithms

A Major Qualifying Project Report:
Submitted to the faculty of

Worcester Polytechnic Institute

In partial fulfillment of the requirements for the

Degree in Bachelor of Science in

Mathematical Sciences

and

Physics

by

Rose Carmichael

Sponsoring Organization:
MIT Lincoln Laboratory

Supervisors:
Drs. Lisa Wei & David Malling

Project Advisors:
Professors Joseph Fehribach & Germano Iannacchione

Contents

List of Figures

Abstract

Radar is a cornerstone of modern intelligence, surveillance, and reconnaissance. While radar can determine the location of a target to within a region of space, fundamental uncertainties exist that limit the accuracy of individual radars. Data fusion is a method that is used to reduce these uncertainties and involves processing the measurements from multiple radars together. One of the main challenges of using data fusion in the field is the difficulty of being able to associate individual detections that correspond to the same target into tracks in real time. Different data fusion algorithms exist to reduce the computation time but the trade-off is lower track accuracy. The goal of this MQP was to quantify these trade-offs for different data fusion algorithms under several scenarios.

In this $1/3$ unit extension to the MQP, the two data fusion methods analyzed in the initial project will be examined. Included is a review of the background information needed to understand radar as well as an introduction to the fundamentals of radar tracking and mathematical estimators, specifically the Kalman filter. This is followed by a full explanation and breakdown of the two data fusion methods in regards to the communications and CPU cost as well as the impact on both of down-sampling, followed by some concluding remarks.

Chapter 1

Introduction

In everything from weather predictions, to air traffic control, to deep space telemetry, radar is an invaluable tool of the modern age. However, radar measurement on its own is not meaningful or useful without a way of processing the data produced. In the relatively simple example of a police officer with a speed gun, that processing requires very few resources and can be done in real time all by a small hand held device. When it comes to more involved tasks, such as target tracking, dedicated infrastructure is required to obtain and process the vast quantities of data produced. Given that more complicated applications of radar can be extremely resource intensive, it is vital when working with these applications to optimize data processing and transfer. One such application, aerial target tracking, will be the focus of this paper.

Radar is a powerful technology for locating and tracking aerial targets, but when using a single radar device, one encounters multiple limitations. Individual radars can track one or more aerial targets by themselves, but can only search a subsection of the sky at any given time with any sort of accuracy. Also, radar error compounds as range increases. In addition to the unavoidable error inherent in radar measurements, a single radar can malfunction or not be properly calibrated.

A method for mitigating the problems with using a single radar to track aerial targets is to combine the data collected by multiple radars into a ground based network. This technique, called data fusion, can increase the accuracy of target tracks. Data fusion can also expand the area that can be monitored as well as provide redundancy in case of instrument failure or other disruptions to equipment. The downside of data fusion is that it requires increased computer processing to handle all the additional data as well as a robust communications network to handle data transfer.

This paper provides the background information required to understand the basics of radar tracking and how it is impacted by implementing data fusion methods. This is done both to fulfill the additional $1/3$ unit required for double major MQPs at WPI and to act as a supplementary document for the project completed at MIT Lincoln Laboratory. The goal of the project was to perform a cost benefit analysis of two proposed data fusion methods. This paper first provides an overview of concepts necessary to understand the process of data fusion, namely radar and radar target tracking and the mathematics behind them. Next it presents the two data fusion methods examined, Central Fusion (CF) and Hybrid Fusion (HF). Following this is a breakdown of the communications and CPU cost of both methods. Lastly, there is an analysis of down-sampling's impact on these costs followed by some concluding remarks.

Chapter 2

Background

2.1 Radar

Radar, being a technology based on the behavior of electromagnetic waves, has roots that go back more than a hundred years. In 1865, James Clerk Maxwell theorized that "light is an electromagnetic disturbance propagated through the field according to electromagnetic laws"[1]. This revolutionized the scientific community's understanding of electromagnetic waves and led to further exploration of electromagnetism. In 1886, Heinrich Hertz discovered and demonstrated that radio waves can be reflected off of solid objects [2]. It was this discovery that laid the groundwork for radar. In the early 1900's, several countries began to secretly develop technology based on the discovery of Heinrich Hertz, attempting to use reflected radio waves to detect solid objects. This technology would in later years come to be called RADAR, or **RA**dio **D**etection **A**nd **R**anging [3].

The basic concept of radar has changed little from its infancy. At its core, radar is the use of radio waves to detect and range, or determine the location of objects in space. Radio waves are first sent from a transmitter into the area under study. A receiver then measures any returning waves, recognizing them as having been reflected from the surface of an object. By measuring the time between the sent and received waves, and knowing the speed of radio waves, it is then possible to determine the distance at which an object was detected.

2.1.1 Radar Basics

Radio waves, like all electromagnetic waves, travel at the speed of light c, or approximately $299{,}792{,}458$ meters per second. The radio waves are sent from the radar transmitter, reflected off the object, and detected at the radar receiver, usually in the same location as the transmitter. The distance r, measured in meters, between the radar and the object can then be calculated as half the speed of light times the amount of time t, measured in seconds, that the radio wave took to traverse the distance to and from the object. This is described in the following equation:

$$r = \frac{ct}{2} \qquad \mathbf{(2.1)}$$

This is an explanation, albeit simplified, of how radar works, and can be seen in Figure 2.1 below [4].

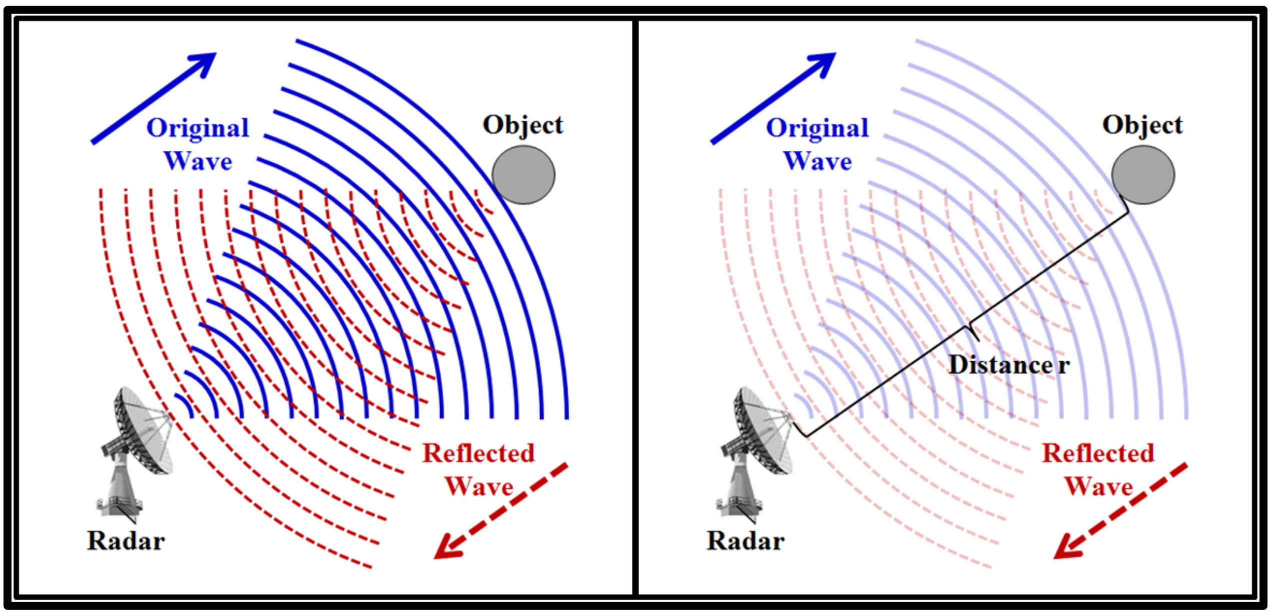

Figure 2.1 How Radar Detects an Object.

This figure illustrates the basic process of detecting a physical object in space by use of reflected radio waves. The distance r, or the distance from the radar transmitter to the object, can be calculated as seen in equation (2.1).

2.1.2 Radar Measurement Uncertainty

In reality, a radar does not emit a mathematically flawless cone of expanding electromagnetic waves that perfectly detect and reflect back information about any object in their path. The most basic radiation pattern of a radar can be seen in Figure 2.2 below [5]. The θ in Figure 2.2 refers to the radar's three decibel (dB) beam width, otherwise known as the three decibel azimuthal (side to side angular) beam width. In a three dimensional description of a radar radiation pattern, this would be written as θ_{3dB} to differentiate it from the three decibel elevational beam width ϕ_{3dB} [6]. The main lobe contains approximately 70% of the radiated energy at its peak and is used to perform the radar's detecting and ranging functions. Typically the side lobes are minimized as much as possible to reduce excess noise. The radar has blind spots in the region directly in front of the transmitter and in the space between the side lobes and the main lobe [5].

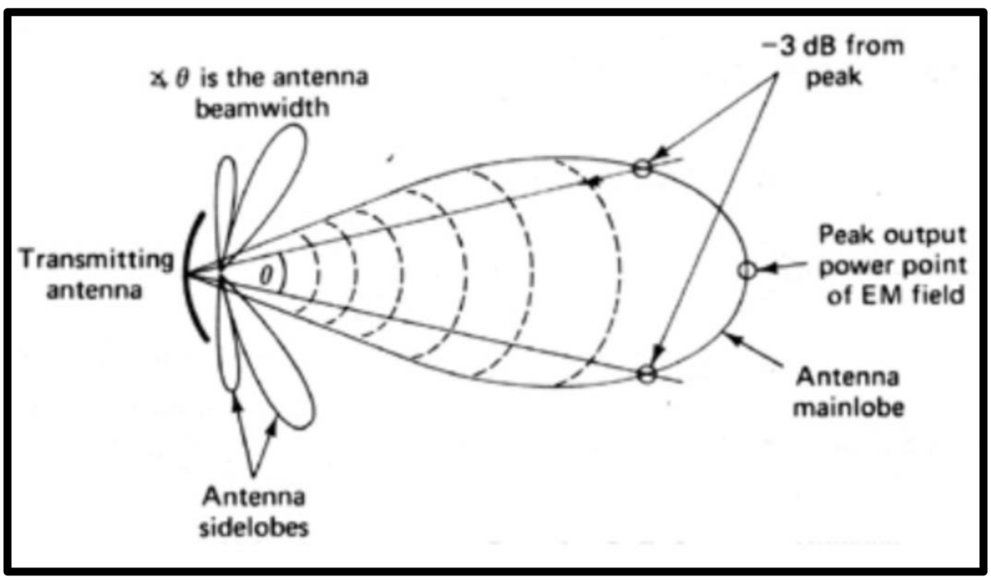

Figure 2.2 Antenna Radiation Pattern.

This figure illustrates the basic radar radiation pattern in two dimensions. The cone created by the 3 dB beam width corresponds with the cone of sight of the more simplified radar model described in 2.1.1 radar basics. Unlike the simplified model, the cone defined by the 3 dB beam width does not have uniform detection ability but instead has a decreasing gradient with the highest detections centered at the peak power output in the center and radiating outward. The 3 dB beam width is a standard cutoff point as the detections farther angularly from the center are deemed to be too scarce with too high of an associated uncertainty to be considered.

The amount of angular uncertainty is increased when the target is angularly farther away from the center of the main lobe. The range uncertainty, or the error associated with the distance to the target, is directly connected to the radar bandwidth B. The bandwidth is just the inverse of the duration of the radar pulse or the pulse width τ [7]. The three equations for the range uncertainty, azimuthal angular uncertainty, and elevational angular uncertainty are [6][8, p. 695][9]:

$$\sigma_R = \frac{c}{2B\sqrt{2SNR}} \qquad (\mathbf{2.2})$$

$$\sigma_\theta = \frac{\theta_{3dB}}{1.6\sqrt{2SNR}} \qquad (\mathbf{2.3})$$

$$\sigma_\phi = \frac{\phi_{3dB}}{1.6\sqrt{2SNR}} \qquad (\mathbf{2.4})$$

where:

c is the speed of light
B is the radar bandwidth
θ_{3dB} is the 3 decibel azimuthal beam width
ϕ_{3dB} is the 3 decibel elevational beam width
SNR is the signal to noise ratio.

These three uncertainties correspond to a pseudo cylindrical region of space surrounding the radar's detection of a target that contains the target's true location.

The last component necessary to understand equations 2.2, 2.3, and 2.4 is the radar's signal to noise ratio or SNR. The radar range equation used to calculate the SNR is [6][8, p. 66-67]

$$SNR = \frac{\sigma_s^2}{\sigma_n^2} = \frac{PG\lambda\sigma^2 n_p^{f_p}}{(4\pi)^3 R^4 k T_0 FBL}. \qquad (\mathbf{2.5})$$

where:

P is the peak power in Watts

G is the gain and equal to $\frac{4\pi A}{\lambda^2}$ (A is the effective area of the aperture)

λ is the operating wavelength of the radar in meters

σ is the radar cross-section (RCS) of the target

n_p is the number of pulses used in each measurement

f_p is equal to 1 with coherent pulse integration and 0.7 otherwise (these values are based on observation [8, p. 67])

R is the range of the target in local spherical coordinates in meters

k is the Boltzman Constant, $1.38 \times 10^{-23} JK^{-1}$

T_0 is the noise temperature and assumed to be $290K$ in most cases

F is the noise figure which describes the general ambient white noise (but not clutter)

B is the bandwidth which is the inverse of the pulse width

L is the system loss which describes attenuation of the radar signal

σ_s^2 is the signal power or mean square voltage induced by the echo

σ_n^2 is the noise power or mean square voltage induced by the background noise.

The underlying physics and mathematics of radar can be found in Jordan Kovar's paper similarly titled *An Assessment of Track Fusion Algorithms* [6]. That paper also contains a description of the functionality and the mathematical considerations that went into the radar simulator we created. Both his paper and this one serve as addendums to the project we completed at MIT Lincoln Laboratory.

2.2 Radar Target Tracking

Being able to determine the location of a stationary object is interesting, but does not have many practical applications. To make the most of radar, it is necessary to be able to determine the continuous position and projected path of a moving object. This brings us to radar tracking. Radar tracking is made possible by the fact that an object's motion can be described, or at least estimated, by a kinematics equation. Radar tracking is the process of predicting an object's future location by estimating, and later refining, the kinematics equation that describes the object's motion.

Before target tracking can be explained, it must be defined. In this context, the term "target track" refers to the kinematics equation, or the equation of motion, of the object. This kinematics equation describes the position (in space and time), the velocity, the acceleration, etc. of the target. The track itself does not contain the previous data collected on the target's location. The process of tracking can be broken down into five main steps: prediction, association, update, initiation, and maintenance. Once a track has been established, the tracking process will iterate through the five steps.

First, the tracker, or tracking algorithm, will, for each active track, make a **prediction** as to the position of the next target detection. This prediction is accompanied by an associated uncertainty, or region of error. When radar detections are received, they are compared to the predictions for each active target track. If a new detection is within the margin of error for both the temporal and physical location of a track's predicted detection, then it is **associated** with that target track.

The track is then **updated** to account for the new information now provided regarding the target's true position and motion. This update refines the values of the coefficients in the track's kinematics equation and reduces the error associated with the track. If a new detection is outside the error threshold for all current tracks, then the tracker will proceed to the **initiation** stage. The initiation stage is when any unassigned detections are used to initiate new tracks. These new tracks are accompanied by large uncertainties because the only information they contain is the current position of the target. The information regarding the target's position, velocity and acceleration will be greatly improved as these new track or tracks are looped through the tracker and gain new radar detections.

The final step is to perform **maintenance** on the various tracks stored in the database. If a track has not been associated with a new detection, then its associated uncertainty is increased. Any tracks that have not been updated for a significant number of iterations or have too high of an associated uncertainty are terminated. These terminated tracks are logged as inactive tracks and removed from the list of active tracks that will proceed again to the prediction phase. This five stage process will then begin its next iteration. A visual of this process can be found in Figure 2.3 below [10].

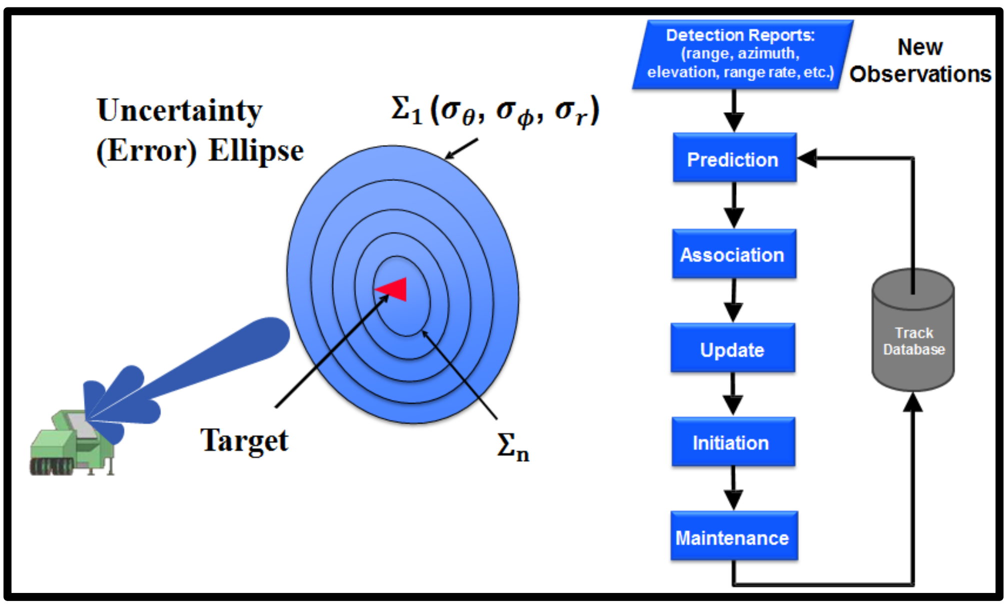

Figure 2.3 Radar Target Tracking Overview.
This figure illustrates the five main steps of implementing a radar tracking algorithm as well as highlighting the beneficial error reduction achieved with each iteration.

2.3 Mathematical Estimators

The five-step iterative tracking process described in section 2.2 is the general approach taken to implement radar tracking. Two of the steps most important to generating tracks, predicting and updating, are handled by a mathematical device called an estimator. Mathematical estimators are the product of a branch of statistics called Estimation Theory. The basic premise of Estimation Theory is to find the values of parameters affecting the distribution of a given set of measured data. These parameters are determined and refined through estimation performed on the measured data. Estimation is used because the measured data contains statistical noise and other inaccuracies. More simply, Estimation Theory is the mathematics of using an estimator to approximate unknown parameters through the use of measurements [11]. In the case of radar target tracking, the goal of an estimator is to use radar detections to improve the target track in the hope of eventually converging to an equation describing the true motion of the object being tracked.

2.3.1 Kalman Filter

The most popular estimator is the Kalman filter (KF). The Kalman filter estimates the state $x \in \mathfrak{R}^{\boldsymbol{n}}$ of a discrete time process governed by a space-time model described by the linear stochastic difference equation [12]

$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1}, \qquad (\mathbf{2.6})$$

with the observations or measurements $z \in \mathfrak{R}^{\boldsymbol{m}}$ at time k of the state x represented by

$$z_k = Hx_k + v_k, \qquad (\mathbf{2.7})$$

where:

A is an $n \times n$ matrix in the difference equation that relates the state at the previous time step $k-1$ to the current step k

B is an $n \times l$ matrix in the difference equation that relates the optimal control input $u \in \Re^l$ to the state x

H is an $m \times n$ matrix in the measurement equation that relates the state to the measurement z_k

w_k and v_k are the random variables that represent the process and measurement noise (respectively). They are assumed to be mutually independent, white, and with normal probability distributions $p(w) \sim N(0, Q)$ and $p(v) \sim N(0, R)$ where Q is the process noise covariance matrix and R is the measurement noise covariance matrix.

In practice, the matrices might change with each measurement or time step, but, for this more simplified explanation, they will be assumed to be constant [12][14].

The basic process of the discrete Kalman filter is that of a predictor-corrector algorithm. The Kalman filter first will **predict** the current state estimate ahead in time with the *time update* and then to adjust or **correct** the projected estimate with an actual measurement at that time with the *measurement update*. The discrete Kalman filter time update equations 2.8 and 2.9 project the state and covariance estimates forward from time step $k-1$ to step k. In other words, the time update generates the *a priori* (denoted with a "super minus") state and covariance estimates [13][14].

$$\hat{x}_k^- = A\hat{x}_{k-1}^- + Bu_{k-1} \qquad (\mathbf{2.8})$$

$$P_k^- = AP_{k-1}A^T + Q \qquad (\mathbf{2.9})$$

where:

P_k^- is the *a priori* estimate error covariance

P_{k-1} is the *a posteriori* estimate error covariance from the previous time step.

The terms *a priori* and *a posteriori* refer to knowledge gained from theoretical deduction and from observation respectively. The Kalman filter measurement update uses the time update equations and the Kalman gain, K_f, found in equation 2.10, to generate the *a posteriori* state and error covariance estimates shown in equations 2.11 and 2.12 [14][15].

$$K_k = P_k^- H^T (H P_k^- H^T + R)^{-1} \quad \mathbf{(2.10)}$$

$$\hat{x}_k = \hat{x}_k^- + K_k (z_k - H \hat{x}_k^-) \quad \mathbf{(2.11)}$$

$$P_k = (I - K_k H) P_k^- \quad \mathbf{(2.12)}$$

The Kalman gain is an $n \times m$ matrix that minimizes the *a posteriori* error covariance. The residual $(z_k - H\hat{x}_k^-)$ is the discrepancy between the predicted measurement $H\hat{x}_k^-$ and the actual measurement z_k. A residual of zero can only occur when the predicted state and measured state are identical. The Kalman gain itself approaches zero as the *a priori* error covariance estimate approaches zero. This will weight the residual less heavily. Essentially, as the *a posteriori* estimate error covariance approaches zero, the predicted measurement $H\hat{x}_k^-$ is trusted more and more and the actual measurement z_k , which contains natural radar measurement error, is trusted less and less. A complete picture of the Kalman filter represented as a prediction-correction algorithm can be seen in Figure 2.4 below [14][16].

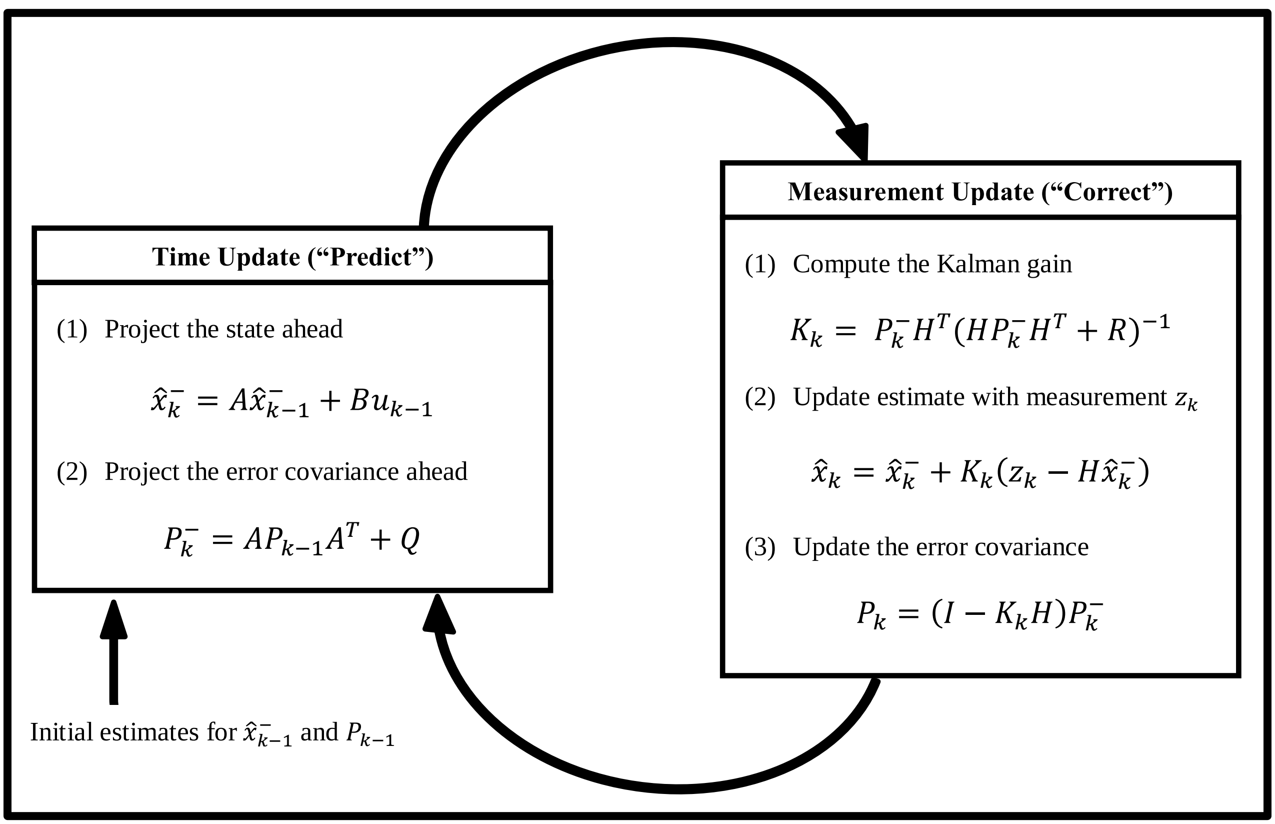

Figure 2.4 Kalman Filter Operation Overview.

This figure provides a complete picture of the operation of the Kalman filter.

2.3.2 Extended Kalman Filter

The Kalman filter is used to estimate the state of a discrete time controlled process that is governed by a ***linear*** stochastic difference equation. This can be an issue if the process that needs to be estimated is not, in fact, linear. The non-linear scenario is by far more common and more interesting. The solution is to use an extended Kalman filter (EKF) that linearizes about the current mean and covariance [14].

The project associated with this paper used the extended Kalman filter as the estimator for the simulation of both the central fuser and hybrid fuser. This was done at the suggestion of Dr. Lisa Wei for simplicity as she determined that it would be sufficient for the project's purposes as the EKF is one of the most widely used estimation algorithms for non-linear systems and is typically recommended as the first filter to try [16][17]. In some applications, the extended Kalman filter would not be appropriate as it is known to be "only reliable for systems that are almost linear on the time scale of the updates" since "difficulties arise from its use of linearization" [15]. But those concerns are less applicable due to the practically linear nature of aerial target motion, even when such targets are performing evasive maneuvers [16][17].

The mathematics behind the EKF are considerably more involved, but follow the same process as the KF. The EKF has a state vector $x \in \Re^n$, now described with a ***nonlinear*** stochastic difference equation

$$x_k = f(x_{k-1}, u_{k-1}, w_{k-1}), \qquad \mathbf{(2.13)}$$

with a measurement $z \in \Re^m$ that is

$$z_k = h(x_k, v_k), \qquad \mathbf{(2.14)}$$

where:

w_k and v_k are the random variables that represent the process and measurement noise (respectively) at each time step.

f is a nonlinear function representing the difference equation with parameters

x_{k-1} the state vector at the previous time step

u_{k-1} the driving function

w_k the zero-mean process noise

h is a nonlinear function representing the measurement equation that relates the state x_k to the measurement z_k.

In practice, the individual values of the noise w_k and v_k cannot be determined at each time step, so the state and measurement vectors have to be approximated with those values set to zero. The time update, Kalman gain, and measurement update equations for the EKF are also only slightly modified from those for the KF. The full EKF process shown in Figure 2.5 is almost identical to Figure 2.4 except that the Jacobians A, W, H, and V now all have the subscript k to reinforce that they must be recomputed at each time step and that the state estimates are modified to represent the nonlinear nature of the stochastic difference equation and measurements [14][16].

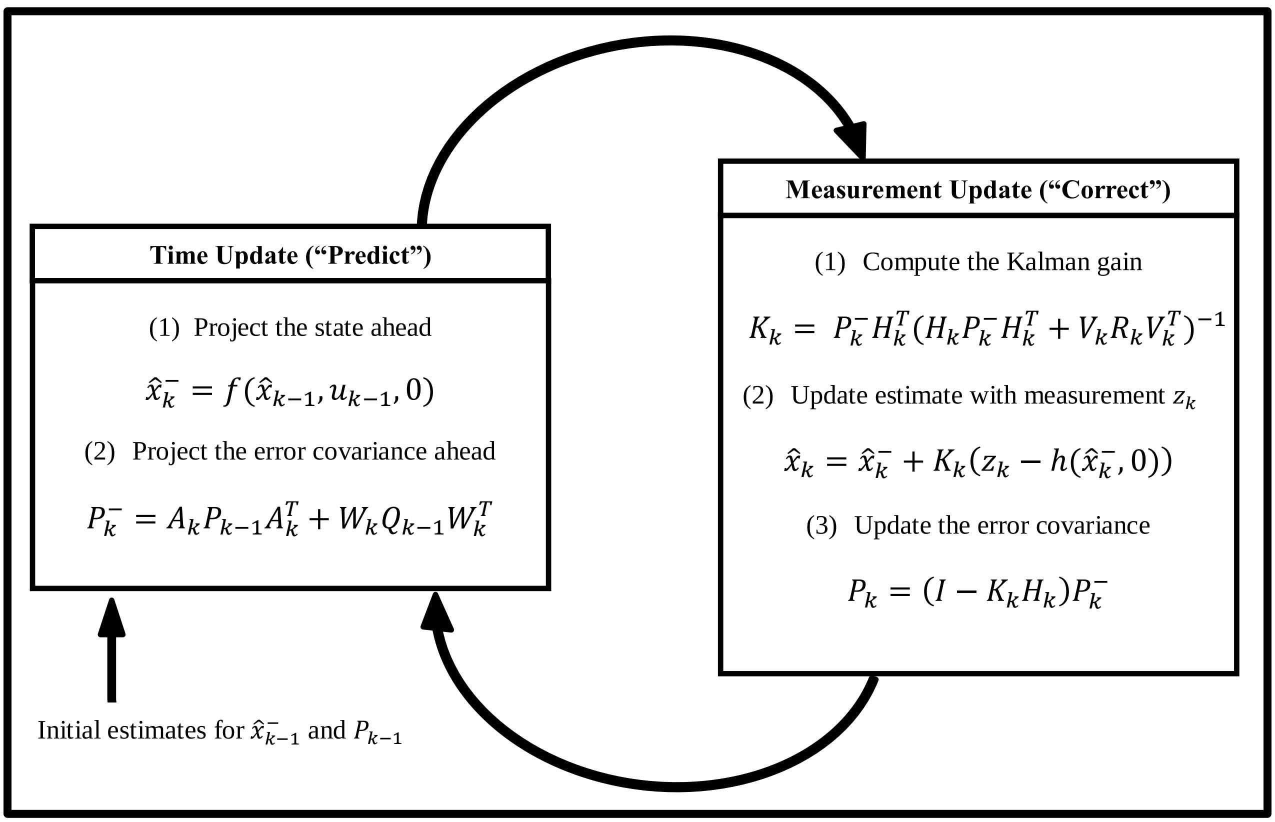

Figure 2.5 Extended Kalman Filter Operation Overview.

This figure provides a complete picture of the operation of the extended Kalman filter.

Chapter 3

Methodology

3.1 Data Fusion

Before the ‘how’ of fusion is explained, it is imperative to understand the ‘why’. In other words, why bother with fusion? There are many benefits to fusing data from multiple sources such as the ability to independently verify information or to mitigate individual calibration issues or to extend the scope of the region that can be observed. Arguably the most useful impact of data fusion is the sizable reduction of associated uncertainty. This finer resolution both greatly reduces error and also enables differentiation between targets in close proximity.

As previously mentioned in section 2.1.2, radars can locate detected objects to within a discrete disk of space with the oval shaped area determined by the two angular uncertainties and the thickness (also call the radar range resolution) constrained the radar pulse width. This region encompassing the detected target is commonly referred to as a range bin. A radar is not capable of distinguishing between one target or multiple targets within the same range bin. The accuracy and resolution of detections is markedly improved when combining information from two data sources as depicted in Figure 3.1 below [18].

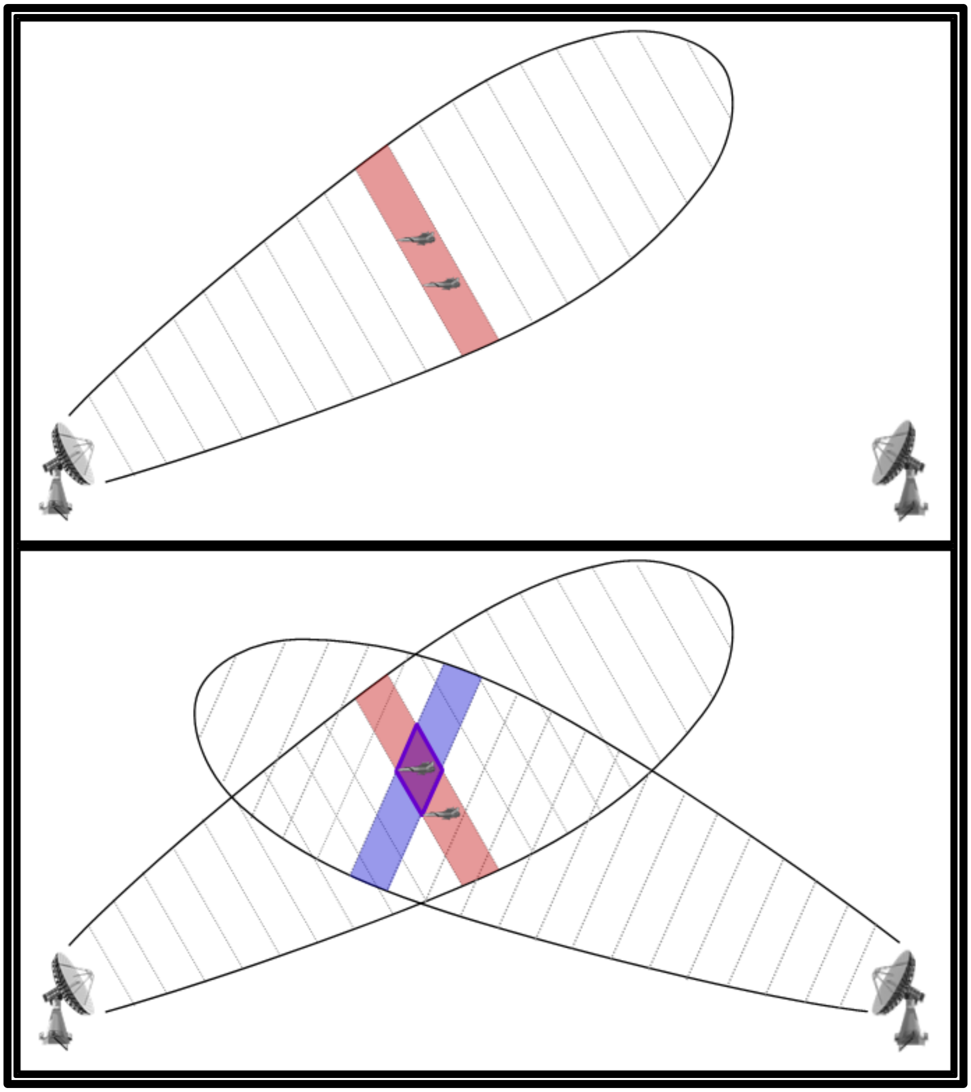

Figure 3.1 Intuition for the Benefit of Fusion.
With each additional data source, the associated uncertainty with each target is markedly reduced. This allows both for lower error when tracking and for finer resolution (thus increasing the likelihood of distinguishing between targets that are close together).

In the top image in Figure 3.1, the left-most radar is unable to distinguish between the two aerial targets as they both fall in the same radar range bin. The addition of the radar on the right in the bottom half of the figure makes it possible to distinguish both targets and reduces the uncertainty on each target from the entire range bin to a much smaller subsection. This figure only highlights the overlap and corresponding reduction in uncertainty for one of the two targets in the interest of clarity.

Now that the incentive to combine information from multiple sources is clear, it is necessary to answer the question of how to go about fusing the data. There are many different data fusion techniques and each is accompanied by its own set of advantages and disadvantages. There are two main categories of fusion methods: track to track fusion and data fusion. Four fusion methods were considered for analysis in the MQP associated with this paper: Hierarchical Fusion, Simultaneous Fusion, Central Fusion, and Hybrid Fusion. The first two were track to track fusion methods (commonly denoted as T2TF) and the last two were data fusion methods. Figure 3.2 below shows those four fusion methods [19].

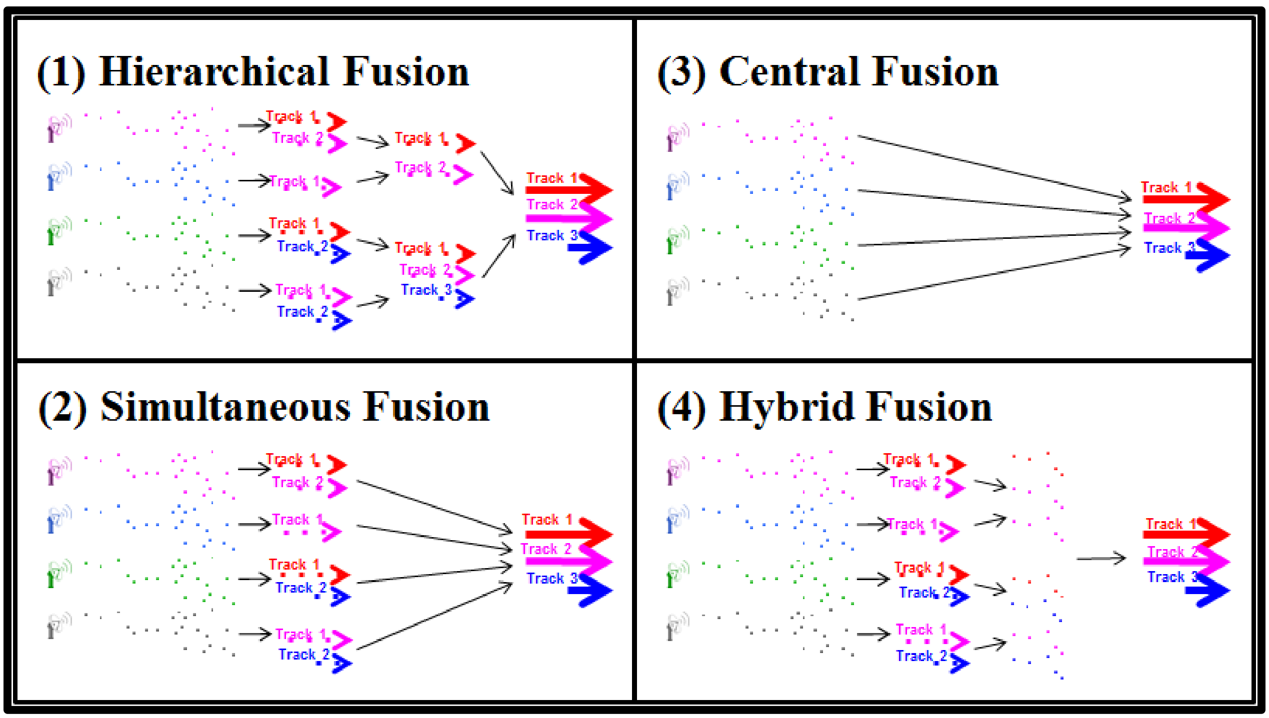

Figure 3.2 Examples of Fusion Algorithms.

This figure shows two examples of track to track fusion or T2TF (1 and 2) and two examples of data fusion (3 and 4). The dots represent raw radar data, the dotted arrows represent tracks created by radars or from track fusion, and the solid arrows represent global tracks that are the end result of the fusion method. The specifics of Hierarchical Fusion and Simultaneous Fusion can be found below. Central and Hybrid Fusion are described in the subsections 3.1.1 and 3.1.2.

The first fusion method under consideration for the project was Hierarchical Fusion as seen in Figure 3.2.1. For Hierarchical Fusion, each radar establishes and updates its own individual track. Then the tracks are fused together in pairs based on predetermined criteria. These pairs are fused until there is only one, complete, fused track. Since the end track is highly dependent on the order that the tracks are fused, the criteria for the hierarchy of radars that dictate the fusion order must be optimally selected. The criteria for fusion order could include the proximity of the detection to the radar, the associated uncertainty, the known accuracy or calibration of the radar, etc. The pros of Hierarchical Fusion are the light communications cost (the tracks are the only information that need to be sent over the network and they consist of a simple kinematics equation) and low CPU cost (the actual process of T2TF is not particularly computationally intensive). The main con is the extreme difficulty in determining the optimal fusion hierarchy. That process in and of itself is so situational that performing a general analysis of the method is not feasible, let alone for a seven week long undergraduate project.

The second possible fusion method for analysis was Simultaneous Fusion as seen in Figure 3.2.2. As in Hierarchical Fusion, each radar establishes its own tracks. Instead of fusing tracks together in successive pairs, Simultaneous Fusion calls for all the tracks to be fused simultaneously. This eliminates the dependence of fusion order and is reasonably simple to implement provided that there is exactly one target detected. When there is only one target, each track can only be associated with that one target, but, as the number of target increases, track association becomes significantly more complex (especially for targets in close proximity with each other). In fact, this track association issue is an open problem that is an area of great interest and study for experts in the field. The magnitude of the track association problem was prohibitive and eliminated Simultaneous Fusion from the list of fusion methods to analyze.

The two remaining methods, Central Fusion and Hybrid Fusion, are both data fusion methods and were therefore perfect for the purposes of the MQP. As they are both common and widely implemented fusion methods, an analysis of the relative communications and computational costs was of practical use to MIT Lincoln Laboratory. Central Fusion and Hybrid Fusion are presented in sections 3.1.1 and 3.1.2 respectively.

3.1.1 Central Fusion

Central Fusion is one of the simplest data fusion algorithms to understand conceptually. Each individual radar in the network generates radar detections and then sends all of that raw data directly to the central fuser to be processed. The central fuser then uses all of the data to make global tracks. Since all of the data is sent to the central fuser in the Central Fusion method, the extended Kalman filter, or whatever estimator is used for tracking, has all available information. Since the central fuser is fed all of the raw data directly from the individual radars, the EKF will be able to produce the *highest accuracy tracks possible* from the available data. The cons of Central Fusion stem from the same source as the pros: the sheer amount of raw data produced and transferred to the central fuser. All of that data must be sent to the central fuser and that produces an extremely heavy communication load. It also makes it so that all data processing will occur at the central fuser. This places the entirety of the CPU burden on the central fuser. Another concern with this data fusion method is that, due to its centralized nature, if anything were to cause a malfunction in the central fuser, there would be no global tracking of any targets. A vital concern would be if the clocks at the individual radars are not synchronized. If that were the case, the time stamps for data collected at the same time would not be used to update the same time step in the EKF. For the rest of this paper, let it be assumed that any radar network has synchronized clocks. Figure 3.3 provides an overview of Central Fusion [20].

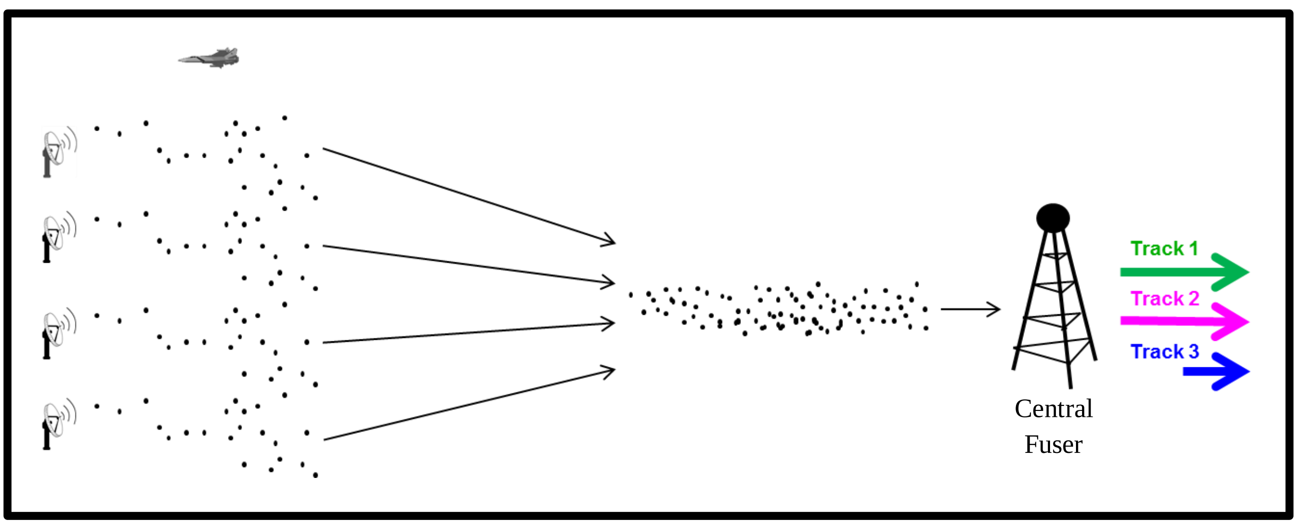

Figure 3.3 Central Fusion Overview.

Each individual radar generates detections and sends that raw data directly to the central fuser to generate global tracks.

3.1.2 Hybrid Fusion

Hybrid Fusion seeks to minimize some of the risks and costs of Central Fusion without losing all of the benefits. Hybrid Fusion is, in fact, a hybrid of Simultaneous Fusion and Central Fusion. As before, each individual radar generates its own radar detections. However, instead of simply transferring all of the raw data to the central fuser, the radars each produce their own target tracks. Then, instead of sending the tracks themselves along to the central fuser, as would be done for Simultaneous Fusion, the raw data points associated with track are sent to the central fuser. This is done because the individual radars can, in creating tracks, remove most if not all of the false alarms and other clutter from the data. The process is not perfect and will also remove some genuine target detections as *any* data point not associated with an established track are not passed along to the central fuser. The filtering of the data prior to it being sent to the central fuser serves to reduce both the communications cost and the CPU cost of the central fuser itself. Figure 3.4 provides an overview of Hybrid Fusion [21].

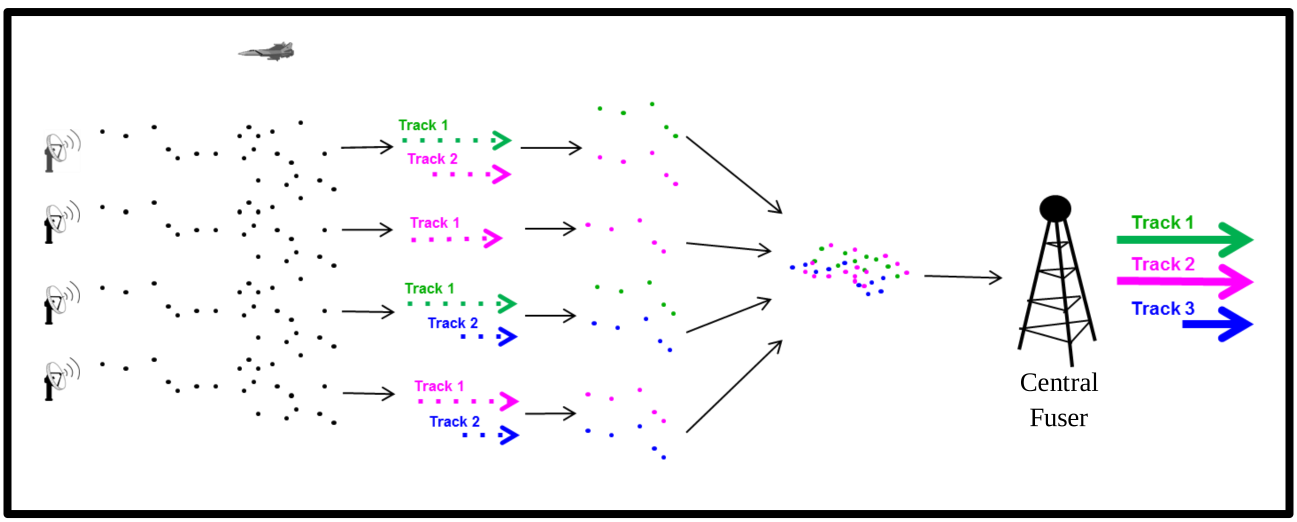

Figure 3.4 Hybrid Fusion Overview.

Each individual radar generates its own target tracks and sends only the data points associated with an established track to the central fuser for global processing.

3.2 Data Fusion Considerations

Every radar measurement consists of three spatial coordinates (r, θ, ϕ), three coordinate measurement errors $(\sigma_r, \sigma_\theta, \sigma_\phi)$, and a time stamp (t). When each of those numbers is represented by a 32 bit floating-point number, each measurement is represented by 224 bits of data. In addition, there is the possibility of the radar picking up false detections along with the detections associated with physical objects. Radars can locate detected objects to within a discrete disk of space with the oval shaped area determined by the two angular uncertainties and the thickness (also called the radar range resolution) constrained the radar pulse width. The probability of these false alarms, P_{fa}, is just the probability that the detected signal will cross the threshold voltage and result in a detection. As the noise is random, the probability of a false alarm occurring is equally likely for each of the radar range bins.

The number of range bins can be calculated by dividing the maximum range ambiguity $R_u = \frac{c}{2PRF}$ (the farthest distance that the radar can detect) by the range resolution, $\Delta R = \frac{c\tau}{2}$, where c is the speed of light, PRF is the pulse repetition frequency, and τ is the pulse width. This makes the number of range bins, N, equal to:

$$N = \frac{R_u}{\Delta R} = \frac{\frac{c}{2PRF}}{\frac{c\tau}{2}} = \frac{1}{\tau PRF} \qquad \mathbf{(3.1)}$$

Assuming a pulse width of $\tau = 1\mu s$ and a pulse repetition frequency of $PRF = 500\ Hz$ (which is on the lower end of the usual range of PRFs), then the number of range bins is:

$$N = \frac{1}{(1\mu s)(500Hz)} = 2000\ range\ bins \qquad \mathbf{(3.2)}$$

It is important to note that the probability of false alarm is most commonly associated with problems internal to the radar system itself. In real world use of radar, there are many other sources of false radar detections and actual radar detections that are not caused by the target being tracked that are a result of the environment. These sources include, but are not limited to,

- weather and other atmospheric interference,
- automotive traffic and other ground based disturbances (this is only an issue for wider elevation angles and when the radar beam is focused lower on the horizon),
- migratory birds and other objects in the air, and
- intentional jamming and other radio interference.

Due to the limited scope of the project associated with this paper, two values for the probability of false alarm, one low P_{fa}, 10^{-6}, and one high P_{fa}, 10^{-3}, will be used to represent low and high noise environments. This is only an approximation intended to roughly simulate good radar target tracking conditions with high functioning equipment versus poor radar target tracking conditions with more faulty equipment. As the goal of the project was not to create a comprehensive model of any conceivable environment with specific and real radar configurations and network layouts, the lumping together of all internal and external error sources into low and high P_{fa} values was determined to be acceptable by both Doctors Wei and Malling. For the rest of this paper, any and all external and internal noise or false detections will be referred to simply as false alarms and assumed to occur at the rate dictated by the specified P_{fa} value.

3.2.1 Communications

To determine the communications cost, or comms cost, of transmitting data for the central and hybrid fusers, we need to make a few assumptions. At any given point in time for a ten radar network, at most ten radars will generate detections. For the following calculations, we will assume that all ten radars are generating detections. We will also assume that the hybrid fuser sorts out all of the false alarms generated by each individual radar prior to transmitting the remaining raw data to the central fuser for global processing and target track production. The number of measurements per radar for the hybrid fuser would then be equal to the PRF. So, the bitrate for the hybrid fuser would be:

$$Bitrate_{HF} = (Number\ of\ radars) * \left(\frac{\frac{Number\ of measurments}{radar}}{second}\right) * \left(224 \frac{bits}{measurment}\right) =$$
$$(10\ radars) * \left(500\ \frac{\frac{measurments}{radar}}{second}\right) * \left(224 \frac{bits}{measurement}\right) =$$
$$1{,}120{,}000 \frac{bits}{second} = 1{,}120\ \frac{kbits}{second} \qquad (\mathbf{3.3})$$

The number of measurements per radar per second for the central fuser will be dependent on the probability of false alarm. Using a low P_{fa} of 10^{-6} and a high P_{fa} of 10^{-3}, the corresponding number of measurements per radar for the central fuser would be

$$\left(\frac{\frac{Number\ of measurments}{radar}}{second}\right) in\ low\ P_{fa}\ enviornment\ = \ PRF + PRF * N * low\ P_{fa} =$$
$$\left(500 \frac{\frac{measurments}{radar}}{second}\right) + (500\ Hz) * (2000) * \left(10^{-6}\ \frac{measurments}{radar}\right) =$$
$$\left(500 \frac{\frac{measurments}{radar}}{second}\right) + \left(1 \frac{\frac{measurment}{radar}}{second}\right) = \left(501 \frac{\frac{measurments}{radar}}{second}\right) \qquad (\mathbf{3.4})$$

for the number of radar measurements the CF would have to process per second in a low P_{fa} environment and

$$\left(\frac{\frac{Number\ of measurments}{radar}}{second}\right) in\ high\ P_{fa}\ enviornment\ =\ PRF + PRF * N * high\ P_{fa} =$$

$$\left(500\frac{\frac{measurments}{radar}}{second}\right) + (500\ Hz) * (2000) * \left(10^{-3}\ \frac{measurments}{radar}\right) =$$

$$\left(500\frac{\frac{measurments}{radar}}{second}\right) + \left(1500\frac{\frac{measurment}{radar}}{second}\right) = \left(1500\frac{\frac{measurments}{radar}}{second}\right) \qquad \mathbf{(3.5)}$$

for the number of radar measurements the CF would have to process per second in a high P_{fa} environment. So the corresponding bitrates for the central fuser would be

$$Bitrate_{CFlowP_{fa}} =\ (Number\ of\ radars)\left(\frac{\frac{Number\ of measurments}{radar}}{second}\right)\left(224\frac{bits}{measurment}\right) =$$

$$(10\ radars)\left(501\ \frac{\frac{measurments}{radar}}{second}\right)\left(224\frac{bits}{measurement}\right) =$$

$$1{,}122{,}240\frac{bits}{second}\ =\ 1{,}122.2\ \frac{kbits}{second} \qquad \mathbf{(3.6)}$$

in a low P_{fa} environment and

$$Bitrate_{CFhighP_{fa}} = (Number\ of\ radars)\left(\frac{\frac{Number\ of measurments}{radar}}{second}\right)\left(224\frac{bits}{measurment}\right) =$$

$$(10\ radars)\left(1500\ \frac{\frac{measurments}{radar}}{second}\right)\left(224\frac{bits}{measurement}\right) =$$

$$3{,}360{,}000\frac{bits}{second} = 3{,}360\ \frac{kbits}{second} \qquad \mathbf{(3.7)}$$

in a high P_{fa} environment.

To put these transmission rate values into perspective, the data transmission capabilities of three real-world communications technology are provided below in Figure 3.5 [22].

	Comms Transmission	Data Rate
Wired	DSL	16-640 kbits/s (send) 1,500-9000 kbits/s (receive)
	Fiber	10,000,00 (10 Gbits/s)
Radio	Wireless Tactical	~100 kbits/s

Figure 3.5 Communications Transmission Rates.

This figure shows the data rates that are able to be processed by three different communication transmission technologies: DSL, fiber optic cables, and wireless tactical radio transmissions. The last method, Wireless Tactical, is highlighted with a green box because the other methods require extensive preexisting infrastructure to be implemented.

Ideally, the ground based radar network would have a fiber optic communications network capable of handling far heavier data loads than either of the data fusion methods described in sections 3.1.1 and 3.1.2. Unfortunately, this is not always feasible. In the situation where there is no preexisting wired infrastructure, the capabilities of wireless tactical communications are a reasonable estimate of the maximum transmission rate for data through the network. The 100 kbits/s available are an order of magnitude short of the transmission rates required for central or Hybrid Fusion in a ten radar network, even in low noise environments. One method of resolving this communications problem is down-sampling. The impact of down-sampling on the communications cost of these data fusion methods is described in section 3.3.

3.2.2 CPU

The computational requirements or CPU requirements, unlike the communications requirements, are not independent of the implementation. This means that the CPU required to process the data is dependent on the actual coding of the tracking and fusing algorithms. To ensure that analysis is applicable to any situation (and to avoid any sort of security concerns that would arise from using specific potentially classified data), the CPU cost will not be presented directly. Instead, the number of data points that need to be processed will stand in for CPU cost as it would be proportional to the actual CPU cost. This provides a decent comparison between central and hybrid fusing methods.

The values for the bits collected per second by a single radar in low and high P_{fa} environments used in Figure 3.6 are calculated in equations 3.4 and 3.5 [23]. These calculations are made with the following assumptions:

- There are 10 radars in the network.
- There is 1 target being tracked.
- There is 1 update per second and there is a detection for each update.
- There are 500 seconds of data collected
- The Hybrid Fuser filters out all false alarms prior to sending the data to the Central Fuser.

	Single Radar		Hybrid Fuser		Central Fuser	
	(10^{-6} P_{fa}^{*})	(10^{-3} P_{fa})	(10^{-6} P_{fa})	(10^{-3} P_{fa})	(10^{-6} P_{fa})	(10^{-3} P_{fa})
Local Radar Computations	501	1,500	5,010 501/radar	15,000 1,500/radar	5,010 501/radar	15,000 1,500/radar
Central Radar Computations	N/A	N/A	5,000	5,000	5,010	15,000
Total Radar Computations	501	1,500	10,010	20,000	10,020	30,000

Figure 3.6 CPU Cost Comparisons.

This figure shows the impact on CPU cost of implementing hybrid versus just using Central Fusion in a radar network constrained by the assumptions described above in both low and high P_{fa} environments with the CPU cost for a single radar tracking on its own as a baseline. The two entries highlighted with green boxes show the drastic reduction in CPU cost at the central fuser achieved by using the Hybrid Fusion method (particularly in high P_{fa} environments).

As the values in Figure 3.6 clearly show, the use of Hybrid Fusion in a high P_{fa} environment reduces the computational processing required by the central fuser to $1/3^{rd}$ of what is required when using Central Fusion alone in otherwise the exact same scenario over the same time interval.

Depending on the actual scenario, the reduction in CPU costs from Hybrid Fusion may or may not be worth the associated reduction in target track accuracy. If the goal is to reduce the CPU costs at the central fuser due to limitations on the processing power there, Hybrid Fusion is probably the method to use. If the total CPU cost is more important, than the reduction is less significant (in this example the CPU cost is only reduced to $2/3^{rds}$ of the amount without Hybrid Fusion). These results can be used as a rough guide for the relative reduction in CPU costs achieved through Hybrid Fusion.

That being said, the cost-benefit analysis between the CPU load reduction and the track accuracy is so inexorably linked to the precise configuration of the radar network, the exact implementation of the two fusion methods, and the specific nature of the target or targets being tracked as well as the environment surrounding the entire situation. As such, this paper cannot provide any actual metrics or cutoff thresholds for what are considered universally acceptable CPU costs.

3.3 Down-sampling

Down-sampling, as the name implies, the a catch-all term used to describe methods of selectively removing data to reduce the total amount necessary to store, transmit, and process. While there are a multitude of ways to 'thin the herd', the most straight forward (and the one that is presented in this section) is to simply send data collected at predetermined intervals. In other words, if each radar sends detections every x seconds, then $1/5^{th}$ down-sampling is if each radar sends detections produced every $5x$ seconds. It is important to note that this does **not** mean that all five detections produced in that time are sent at once, but rather that only the one detection produced at that interval is sent. This reduces the total size of the data that must be sent over the communications network and then processed. A visual representation of this example can be found in Figure 3.7 below [24].

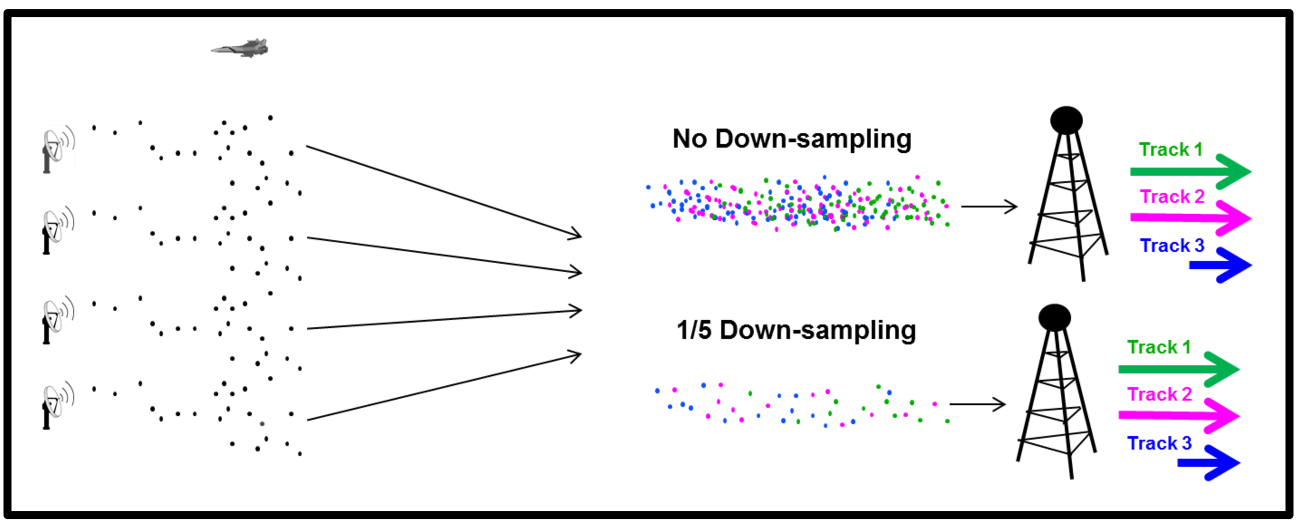

Figure 3.7 Down-sampling Overview.

The individual radars generate detections but only the data collected at every second, or fifth, or some other predetermined interval is sent to the central fuser for global processing

At a glance, there might seem to be a glaring flaw in this method of down-sampling. It **appears** to reduce the *total* communications cost by creating sections of time with no traffic on the communications system interspersed with pockets of traffic with the same prohibitive data transmission rate as before. This is not the case because, while the data that is being sent is obtained at the same instant in time by all of the separate radars, that data can now be staggered in such a way that it is transmitted over the entire time window before the next detection. This is possible because, as described in the first paragraph of section 3.2, each detection consists of three spatial coordinates, three measurement uncertainty parameters, and one time stamp t. This t ensures that the central fuser will be able to process all of the simultaneous data points in the same target track update. This does create a slight lag between the track produced by the central fuser and the actual position of the target, but any application that uses the target track in real time can easily take that into account by using the current kinematics equation to predict the location farther ahead in time or my any number of other methods.

The impact of this method of down-sampling on the CPU cost is completely straight forward. Whatever the CPU cost was at the individual radars and at the central fuser is reduced by the rate of down-sampling. Fewer data points to be processed will result in fewer necessary computations and thus a proportionally lower CPU cost. The impact of down-sampling on the communications cost is no less straight forward, but can be examined in relation to the achievable data transmission rates for different communications methods as presented in Figure 3.5. Figure 3.8 uses the values produced for the ten radar network scenario described in section 3.2.1 by equations 3.3, 3.6, and 3.7 and displays the impact of different levels of down-sampling on the communications cost for the system using Hybrid Fusion versus Central Fusion in both low and high P_{fa} environments [25]. In that particular scenario, the transmission rates would have to be down-sampled to at least $1/10^{th}$ of the initial rate (and even more in the high P_{fa} environment when using Central Fusion as the data fusion method) to be within the capabilities of wireless tactical communication.

	Hybrid Fuser (10^{-6} P_{fa})	Hybrid Fuser (10^{-3} P_{fa})	Central Fuser (10^{-6} P_{fa})	Central Fuser (10^{-3} P_{fa})
No Down-sampling	1,120	1,120	1,122.2	3,360
1/2 Down-sampling	560	560	561.1	1,680
1/5 Down-sampling	224	224	224.4	672
1/10 Down-sampling	112	112	112.2	336
1/30 Down-sampling	37.3	37.3	37.4	112
1/60 Down-sampling	18.6	18.6	18.7	56
1/120 Down-sampling	9.3	9.3	9.4	28

Figure 3.8 Impact of Down-sampling on Communications Costs.

This figure shows the impact of different levels of down-sampling on the data transmission rates, in kbits/second, of the previously described ground based radar network tracking a single aerial target in both low and high P_{fa} environments when using Hybrid Fusion versus Central Fusion. The green box highlights the levels down-sampling that produce bitrates within the data rate capabilities of wireless tactical communication: a bitrate of approximately 100 kbits/second.

From this analysis, it is readily apparent that down-sampling can reduce the computational burden and lighten the communications load enough to potentially allow for use of a wireless tactical communications network. Unfortunately, these reductions do not come without the price of reduced track accuracy. Much of the benefit of using an estimator such as the EKF to track a target stems from the iterative nature of the error reduction. As depicted in both Figures 2.4 and 2.5, when each new radar measurement is provided to the EKF and used to correct the prediction, the error covariance is also corrected and reduced. Down-sampling reduces the frequency of those updates and therefore increases the amount of total time before the track can achieve the same amount of accuracy as the track produced with the non-down-sampled data set. Since there is more time between each update, the target will also have had more time to move. This means that the tracker will have to make predictions about the target's location at larger spatial intervals. The larger intervals also increase the associated uncertainty with the predictions, further reducing the accuracy of the target track.

Chapter 4
Conclusion

This paper provided the background required to gain a basic understanding of radar and radar target tracking and how that tracking is affected by the use of data fusion. Section 2.1 introduced the fundamentals of radar (bouncing radio waves off of solid objects to determine their position in space), as well as the inherent nature and cause of uncertainty or error in radar measurements. Section 2.2 provided a basic intuition for how radar is used to track targets and section 2.3 introduced how those tracks are produced through the use of estimators and explained the mathematics behind two of those estimators; the Kalman filter and the extended Kalman filter.

Section 3.1 described the concept behind and benefits of fusing the information from multiple radars to create more accurate target tracks with more coverage and reliability than tracks produced by one radar alone. It also discussed the main categories of fusion, track to track fusion and data fusion, and the benefits and draw backs of four fusion algorithms with particular emphasis on the two data fusion methods of Central Fusion and Hybrid Fusion. Section 3.2 analyzed the communications and computational requirements of Central and Hybrid Fusion and provided real world values for feasible transmission bitrates of standard communication methods as well as a metric to compare the relative CPU costs in different scenarios.

The final section, section 3.3, proposed, analyzed, and discussed down-sampling as a method for mitigating the communications and CPU costs of data fusion. The purpose of this paper was to provide enough background and context to intuitively understand the methods and results of the MQP completed at MIT Lincoln Laboratory for Group 101 in the fall of 2015.

This paper was also intended to be able to be used as a framework to understand considerations that go into the selection of a data fusion methods for a given radar network and aerial target tracking scenario. In a noisy or cluttered environment radars produce larger quantities of detections that have high rates of false detections and high associated uncertainties. In that scenario, a more distributed fusion method such as Hybrid Fusion would be a more intuitive choice than a very centralized method like Central Fusion, especially if the radar network is using a wireless tactical communications system. In a situation where high accuracy is a priority and there is no communications constraint, Central Fusion might be preferred.

There are many different applications for radar target tracking that all place differing amounts of importance on communications and CPU cost reduction versus track accuracy. This paper analyzed only two of the multitude of potential data fusion methods as it is difficult to conduct useful universal analysis of such subjective costs and benefits. One area for further study could be to investigate the considerations of a handful of specific scenarios to produce more definitive recommendations for when to use a specific fusion algorithm. It would also be interesting to analyze other data fusion methods such as combining some ideas from Hierarchical Fusion with Hybrid and Central Fusion. Such a fusion algorithm could combine raw data or a subsection of the radar network, and then send the data associated with the tracks produced to a global central fuser. This proposed system has the potential to drastically reduce communications cost, but would have to be thoroughly analyzed to know what impact it could have on the quality of various metrics.

One final area for further study is the possibility of using down-sampling as a way of tracking multiple aerial targets flying in close proximity to each other. The basic idea of this "Swarm Tracking" arises from one of the perceived draw backs of down-sampling; the larger prediction regions caused by the delay between updates. In a scenario where there are multiple targets traveling in a "swarm", the tracker has great difficulty in distinguishing detections from the individual targets. Since not every target is guaranteed to be detected on each iteration of the tracking algorithm, detections from different targets are likely to be confused and thus produce tracks that are not of a single target but in fact an amalgamation of multiple targets. The tracker would run the risk of not getting enough sequential updates (due to the difficulty in making an association between the track and incoming radar detections) to avoid having the maintenance phase of the target tracking algorithm remove the track.

This would be increasingly more likely for smaller targets or targets with lower radar cross sections as they are less likely to be detected at each time step. When the radar detections are down-sampled there are larger gaps between detections. This would force the uncertainty, or region around the prediction that would cause detected measurements to be associated with the track, to be larger. Now, instead of trying to track all of the targets individually, there could be one track for the entire swarm. The motion of the swarm could then be predicted as a whole by treating the individual aerial targets as one large object. Further study, analysis, and refinement of the "Swarm Tracking" potential of down-sampling could serve to augment the tracking capabilities of ground based radar networks presented in this paper.

Bibliography

[1] Maxwell, J.C., *A Dynamical Theory of the Electromagnetic Field*, pages 459-512. Philosophical Transactions of the Royal Society of London, 1865. [Online; accessed September 2016].

[2] Kostenko, A. A., Nosich A. I., and Tishchenko I. A., *Radar Prehistory, Soviet Side*, Proc. of IEEE APS International Symposium 2001, vol.4. p. 44, 2003

[3] McGraw-Hill dictionary of scientific and technical terms / Daniel N. Lapedes, editor in chief. Lapedes, Daniel N. New York; Montreal : McGraw-Hill, 1976. [xv], 1634, A26 p [Online; accessed September 2016]

[4] Carmichael, R.T., How Radar Detects an Object. Worcester Polytechnic Institute. Personal Communication. 2016.

[5] Zarinamdamin, *Chapter 5 Microwave Antenna*, 13 Nov. 2013. [Online; accessed October 2016]

[6] Kovar, Jordan, *An Assessment of Track Fusion Algorithms*, Worcester Polytechnic Institute, 2016

[7] Wolff, Christian. *Radar Tutorial*, GNU Free Documentation License [Online; accessed November 2016].

[8] Richards, Mark A.; Scheer, Jim; Holm, William A. *Principles of Modern Radar: Basic Principles*. SciTech Pub., 2010.

[9] Ryuhei Takahashi, Kazufumi Hirata, Teruyuki Hara, and Atsushi Okamura. Derivation of monopulse angle accuracy for phased array radar to achieve cramer-rao lower bound. In *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference* on, pages 2569-2572. IEEE, 2012.

[10] MIT Lincoln Laboratory. Radar Target Tracking Overview. Personal communication.

[11] Walter, E.; Pronzato, L. *Identification of Parametric Models from Experimental Data.* London, UK: Springer-Verlag, 1997 [Online; accessed November 2016].

[12] Castanedo, Federico. "A Review of Data Fusion Techniques," *The Scientific World Journal*, vol. 2013, Article ID 704504

[13] Siciliano, B; Khatib, O. (Eds). *Springer Handbook of Robotics: Chaper 25 Multisensor Data Fusion.* 2008. [Online; accessed July 2015].

[14] Welch, Greg;Bishop, Gary. *An Introduction to the Kalman filter*. University of North Carolina at Chapel Hill, July 24, 2006, [Online; accessed August 2015].

[15] Julier, S.J.; Uhlmann, J.K., *Unscented filtering and nonlinear estimation*. Proceedings of the IEEE: 401–422. 2004 [Online; accessed October 2016].

[16] Wei, Lisa. MIT Lincoln Laboratory. Personal Communication. 2015.

[17] Chang,C. Bing; Dunn, Keh-Ping. *Theory and Application of Estimation and Association*. Lecture 4: Nonlinear State Estimation, MIT Lincoln Laboratory, Lexington MA. June 22 2015.

[18] Carmichael, R.T. MIT Lincoln Laboratory. Intuition for Benefit of Fusion. Personal Communication. 2015.

[19] Wei, L.; Carmichael, R.T. MIT Lincoln Laboratory. Examples of Fusion Algorithms. Personal Communication. 2015.

[20] Carmichael, R.T.; Kovar, J.; Wei, L. MIT Lincoln Laboratory. Central Fusion Diagram. Personal Communication. 2015.

[21] Carmichael, R.T.; Kovar, J.; Wei, L. MIT Lincoln Laboratory. Hybrid Fusion Diagram. Personal Communication. 2015.

[22] Kovar, J.; Wei, L. MIT Lincoln Laboratory. Communications Transmission Rates. Personal Communication. 2015.

[23] Kovar, J.; Carmichael, R.T., MIT Lincoln Laboratory. CPU Cost Comparisons. Personal Communication. 2015.

[24] Carmichael, R.T.; Kovar, J. MIT Lincoln Laboratory. Down-sampling Diagram. Personal Communication. 2015.

[25] Carmichael, R.T.; Kovar, J. MIT Lincoln Laboratory. Impact of Down-sampling on Communications Costs. Personal Communication. 2015.