Artificial Intelligence and Radar Target Tracking

Gerard T. Capraro
Capraro Technologies, Inc., 401 Herkimer Road, Utica, NY 13502 USA

Abstract

An airborne ground looking radar sensor's performance may be enhanced by selecting and modifying algorithms adaptively as the environment changes. A brief presentation of an airborne intelligent radar system (AIRS) is provided. A description of the knowledge based tracker portion of AIRS is emphasized. Many tracking algorithms discard all information about a track once a track is dropped. Our approach maintains this information to enhance the algorithm's learning ability.

Introduction

The desire to anticipate, find, fix, track, target, engage, and assess, anything, anytime, anywhere (AF2T2EA4) by the US Air Force (USAF) will require changes to how we modify, build, and deploy radar and sensor systems. The US Air Force Research Laboratory (AFRL) is attacking these issues from a sensor and information perspective and has generated a way forward in their defining of layered sensing [1].

How can the US Air Force system of the future detect and identify threats and meet the implicit requirements of this scenario in a timely manner? We must, as a first step to full automation, implement the following ground breaking changes: place more compute intensive resources closer to sources of the information gathering – e.g. assign tasks to sensors to look for "triggers" created from intelligence surveillance and reconnaissance (ISR) sources, provide for the analysis of intelligence data automatically and without human involvement, move the human sensor operator from managing data - to managing actionable knowledge and sensor aggregation, and develop these "triggers" and rules for automatic assignment and management of heterogeneous sensors to meet dynamic and abstract requirements. This paper will address an airborne ground looking radar and how to use artificial intelligence (AI) to enhance its tracking performance.

Sensor performance may be enhanced by selecting algorithms adaptively as the environment changes. It has been shown [2-12], that if an airborne radar system uses prior knowledge concerning certain features of the earth (e.g. land-sea interfaces) intelligently, then performance in the filtering, detection and tracking stages of a radar processing chain improves dramatically. As an example the performance of an intelligent radar can be increased if the characteristics and location of electromagnetic interference, terrain features [9], mountainous terrain [10], and weather conditions are known. The Sensors Directorate of the USAF Research Laboratory conducted and sponsored research and development in the use of prior knowledge for enhancing radar performance, as did the Defense Advanced Research Project Agency (DARPA) under the Knowledge Aided Sensor Signal Processing Expert Reasoning (KASSPER) program.

One design of an intelligent radar system that processes information from the filter, detector, and tracker stages of a surveillance radar, investigated by the USAF and under the KASSPER program, was specifically designed for an Airborne Intelligent Radar System (AIRS). Futuristic advanced intelligent radar systems will cooperatively perform signal and data processing within and between sensors and communications systems while utilizing waveform diversity and performing multi-sensor processing, for reconnaissance, surveillance, imaging and communications within the same radar system. A high level description of AIRS is shown in Figure 1 and is described in detail, [8, 11], in the literature. This work has been extended to include metacognition and is illustrated in Figure 2. See [13, 14] for a more detailed description.

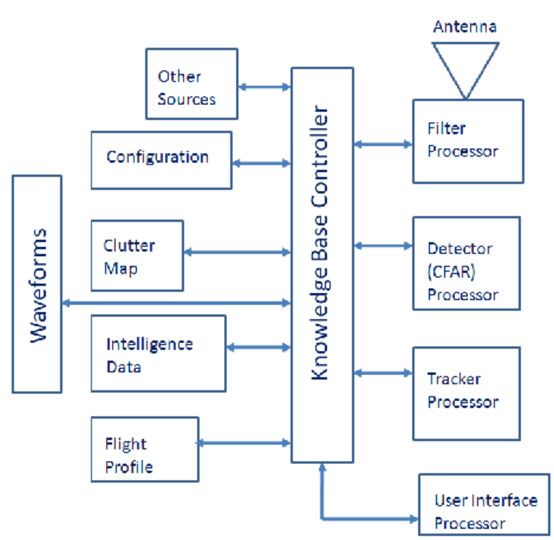

Fig.1. Airborne Intelligent Radar System (AIRS)

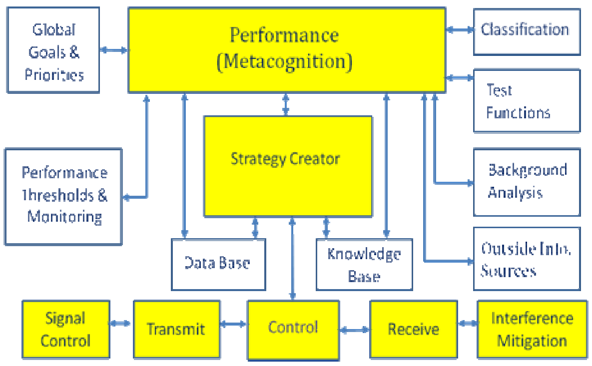

Fig. 2 Cognitive Radar Software Architecture

In this paper we wish to investigate the tracking portion of AIRS. See Figure 1. In [13] we addressed the filter and detector portions. This paper will extend our work of an AIRS architecture. We will present a AI overview of this tracking algorithm and some of its AI rules e.g. maneuver or obstacle rules and shadow rules. An AI logic structure for implementing these rules is discussed next and some additional rules for our AIRS design are provided.

The logic structure is independent of any tracking algorithm and can address aircraft or ground moving targets. It is compatible with the overall AIRS design and is modifiable. The thrust of this logic structure is to utilize as much auxiliary data (e.g. maps, other sensors, target kinematics, and radar platform characteristics) as possible to maintain individual identifiable tracks. With today's tracking algorithms if a track is dropped and another track is formed there is minimum effort expended to determine if the two tracks were formed from the same target. If a track is dropped algorithms, for the most part, do not investigate why and then use this information in enhancing the overall signal processing performance. Algorithms do not learn based upon their previous performances. They are memoryless once a track is dropped. The proposed logic structure presented herein addresses these issues and investigates the potential for building an AI based tracking algorithm.

Our current tracking algorithm has three separate instantiations. There is an uncoupled two state alpha beta filter with position and velocity component states, an uncoupled three state Kalman filter with position, velocity, and acceleration component states, and an extended four state Kalman filter with both x and y position and velocity component states. The tracker gathers reports, evaluates the reports and correlates them with known tracks, forms a correlation matrix and distance matrix, performs an association logic based upon nearest neighbor and oldest track, and performs track maintenance i.e. update extant track states, spawn new tentative tracks with unused reports and drops tracks with a state value of zero. A diagram illustrating the state logic is shown in Figure 3. A new tentative track is given a state of 1. If its projected position is detected again on the next coherent processing interval (CPI) it is given a state of 2, and so on. Once the target is in state 4 it is considered in a firm state as long as it is still detected for each subsequent CPI. Once in the firm state, if there are four consecutive CPIs in which the target is not detected (i.e. a Miss) then the track is dropped. It is our contention that once a tentative track exists then we should maintain its history even if it receives one or more misses. This is important in order to correlate false or dropped tracks with roads, or jammers, discretes, shadow regions, etc. This information is needed to feed back to the Knowledge Base Controller (KBC) shown in Figure 1.

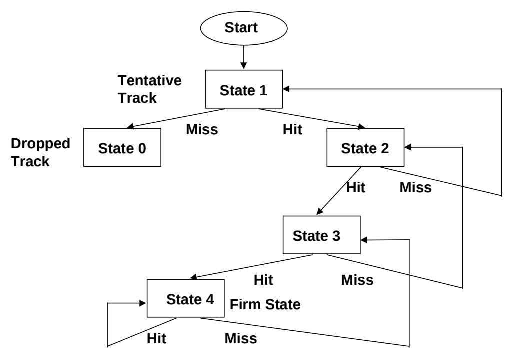

Figure 3. Integrating AI Rules

The following is a preliminary design of a logical structure to capture AI rules for the tracking portion of AIRS. It is by no means complete and does not address each of the numerous attributes for tracking any specific type of target (e.g. aircraft, Unattended Air Systems (UAS), ground vehicles, missiles) for all its possible scenarios embedded in all possible environments or clutter. It is constructed to work with a radar tracking filter such as alpha beta or Kalman. The logical structure is shown in Figure 4. It is an abstract model and will require numerous detail level designs before it can be coded and tested. The logic is described using alpha characters to indicate where in the structure we are referring. Throughout the description the use of outside data sources is illustrated and the addition or verification of data sources is presented.

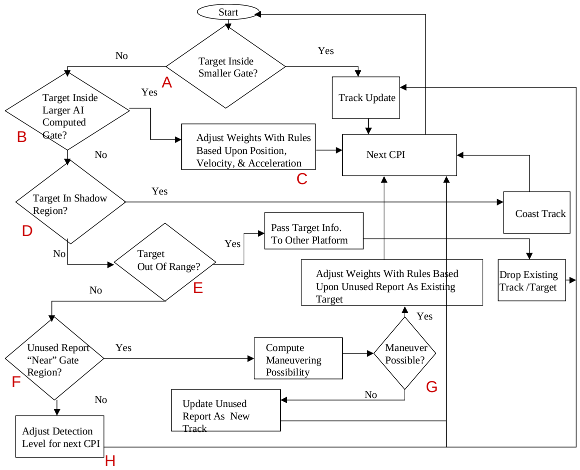

Figure 4. Logical Structure

Section A

Within this decision block (A) we are asking whether a detected target is within the gate of a known and therefore projected track. If the answer is yes then we simply update the track using the tracking filter of choice (e.g. Kalman). If however a target is detected and it is not within any projected track's gate (i.e. an unused report) then we need to determine whether it lies in a larger AI computed gate. The idea of using more than one size or variable size gate is discussed in the literature. Skolnik [15] suggests "The size of the small gate would be determined by the accuracy of the track. When a target does not appear in the small gate, a larger gate would be used whose search area is determined by the maximum acceleration expected of the target during turns." Brookner [16] states while discussing the g-h filter

"However, aircraft targets generally go in straight lines, rarely doing a maneuver. Hence, what one would like to do is use a Kalman filter when the target maneuvers, which is rarely, and to use a simple constant g-h filter when the target is not maneuvering. This can be done if a means is provided for detecting when a target is maneuvering. In the literature this has been done by noting the tracking-filter residual error, that is, the difference between the target predicted position and the measured position on the nth observation. The detection of the presence of a maneuver could be based either on the last residual error or some function of the last m residual errors. An alternative approach is to switch when a maneuver is detected from a steady-state g-h filter with modest or low g and h values to a g-h filter with high g and h values, similar for track initiation. This type of approach was employed by Lincoln Laboratory for its netted ground surveillance radar system. They used two prediction windows to detect a target maneuver. If the target was detected in the smaller window, then it was assumed that the target had not maneuvered and the values of g and h used were kept ... If the target fell outside of this smaller 3 sigma window but inside the larger window called the maneuver window, the target was assumed to have maneuvered."

Section B

These references were provided to indicate that the radar community has tried different approaches for varying the gate sizes for tracking maneuvering targets. The Kalman filter is more suited for maneuvering targets. However, a universal method for choosing a larger gate size because of a maneuver is not well established. If the larger gate is too large then multiple targets may occur within them. The maneuverability of a target is target dependent and may be human dependent and very unpredictable. What we are proposing is that the larger gate be built using AI techniques. Let the history of the target's flight and a priori knowledge about a potential target dictate how to compute the larger AI gate, e.g. a UAS versus a B-52 aircraft.

Since we are building an intelligent surveillance system we will have data obtained from sources outside our radar system, e.g. map data, intelligence data, and other sensors. We can assume we know what type of targets we are tracking, such as helicopters, tanks, scud launchers, surveillance aircraft, fighter aircraft, and missiles. If so then we know something about their kinematics, i.e. their minimum, maximum and average velocities for different altitudes, their maximum gravitational (G) force turn they can withstand and at what radius, and their maximum acceleration. Using these data we can construct rules that will compute the larger size gate based upon a degree of belief given the type of target, e.g. helicopter or a fighter aircraft. This degree of belief can be computed using information from outside data sources, its previous kinematics data (velocity, location, etc.), radar cross section, and altitude amongst other factors such as the type of mission, its position in the scene, and sensitive locations or targets.

A simple rule is to take the maximum velocity for the target type that has the highest belief and compute the maximum distance it could have traveled from the previous position on the last CPI. This allows us to compute a semi-circle around the vector the target was heading. See Figure 5. This approach may be fine for a target like a surveillance aircraft, but not for a tank or track vehicle or scud launcher. For example, a tank which can easily turn 180 degrees, a circle may have to be drawn with radius equal to the maximum distance that can be traveled within the time between CPIs. The more we know about the targets we are tracking the more intelligent we can be in designing our rules and estimate our gate sizes.

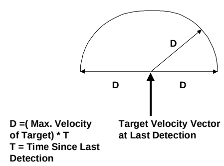

Figure 5. Example AI Computed Gate

Section C

If the target is detected in the larger gate then we need to adjust the weights of our tracking filter. Indicated in block C we can adjust the weights with rules based upon position, velocity and acceleration. These rules can be simple, e.g. if the target was detected in the larger gate then set the weights for the next CPI as if the target were detected the first time. This will eliminate any memory or smoothing that the filter had performed and start off with a larger gate size. More sophisticated rules can be employed and should be investigated further, dependent upon the tracking filter used.

Section D

If the target was not found in the smaller or the larger gate then we need to determine if it is being shadowed from our radar, possibly by terrain. Our logic is assuming that the radar system has a priori data that are available such as terrain data containing elevation attributes, roads, and bridges. With this information we can compute whether or not given the elevation of the radar and the last position of the track if there is terrain obstructing the radar's illumination of the target. If there is an obstruction then we should be able to project, given the last known velocity of the track and the changing position of the radar, how many CPIs the track will be obstructed. Based upon these computations we can then coast the track until the next CPI. For each coasted CPI we should also look for new unused reports that can occur due to our coasted track changing its projected velocity while it is being obscured. See Figure 6. If this does occur and a new track is initiated we should "flag" this track that it may be the coasted track. Once we compute when or which CPI the original track should be visible and if it isn't, even after two additional CPIs, we should then revisit the new track. During this revisit we need to compute whether or not the dynamics of the target/track were capable of maneuvering to the position that the radar detected the target. (See paragraphs F and G for more details.) If it is shown possible, then the new track should be updated as being the old track with some degree of belief. If however, the original track is detected after it has moved beyond the obstruction then we should go back to the new track that was initiated and remove the "flag" indicating the possibility that this was a firm track that was coasted.

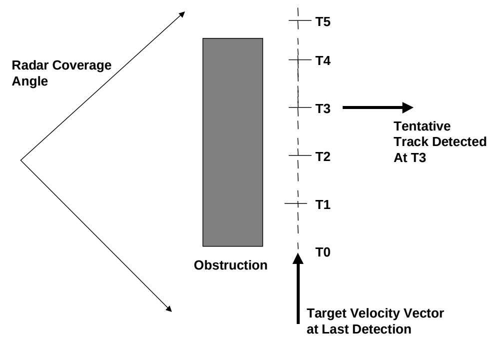

Figure 6. Track Obstruction

Section F

If the target is not in either gate and it is not shadowed then maybe the target is out of range. This is easy to compute given its last position relative to the radar. If it is out of range then we should pass this information to another sensor platform along with the track data we have acquired. The knowledge of when a target is going to reach this point can be predicted earlier than the last CPI. However, the point in space when a target will be out of range is a variable dependent upon the radar and the target's movements.

The information that can be passed to the other platform can contain the time of the first acquisition, its history path, velocity range, hypothesis of type of target, and any other kinematics or knowledge that has been gathered throughout its track. This data can be used by the message receiving platform in assigning degrees of belief about the target's maneuverability, type of target, and identification.

Section F

If the target is not in either gate, not shadowed, and not out of range then what happened to it? Maybe our knowledge about its kinematics was incorrect? Maybe our sensor and filtering model has more error variation than we thought? Maybe the target maneuvered and its radar cross section (RCS) is too low and therefore not detected. Maybe the clutter is too large and we can't detect the target?

What we can do is determine if there are any unused reports. If unused reports exist then maybe one of these are our target. First we need to perform a quick culling to determine if at maximum velocity (Vmax) our target could have traveled from where we last detected it to where the unused report was detected, a distance of D. If Vmax times T (time between the two detections) is less than D then this unused report can't possibly be due to the same track. If all unused reports result in the same finding then we conclude that there are no unused reports that may be due to our track. If however, one or more computations show that the distance to the possible reports could have been traveled by the target then we need to compute its possibility and assign a degree of belief to each report.

Section G

A simple algorithm for computing the possibility of an A/C maneuverability is illustrated in Figure 7. D is the distance between the last detection and the position of an unused report. The different radii (R1 and R2) represent the different radius that one can construct that can pass a circle or arc through the two end points of the chord of length D. If we assume that the acceleration is a maximum then we can assume that the velocity is our last estimated velocity or its maximum velocity. Each assumption has a certain amount of error. We can compute different values of R by the following:

$Rest = (Vlast)^2/Accmax$,

$Rmax = (Vmax)^2/Accmax$.

For different values of R and D we can compute the distance of the arc connecting the end points of the chord D. It can be shown from Figure 7 that:

$Theta = 2(\arcsin((D/2)/Rest))$ or
$Theta = 2(\arcsin((D/2)/Rmax))$.

The distance along the arc is $2*Pi*Rest/(Theta/360) = Darcest$. Therefore if at $(Vlast)*T$ is less than Darcest then the maneuver is not possible. Similarly if $(Vmax)*T$ is less than Darcmax then the maneuver is not possible.

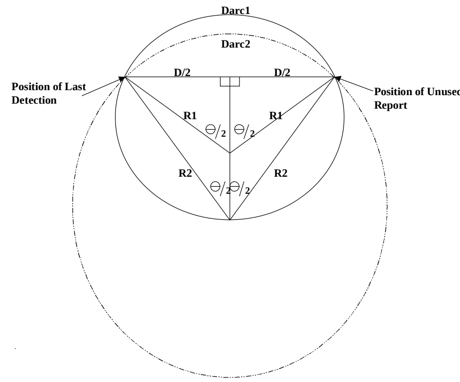

Figure 7. Maneuver Possibilities

Similar rules can be developed for different targets and their kinematics to determine the best rules for each. The developed rules can be verified and modified by consulting with experts who are aware of a target's kinematics.

Section H

If the target is not in either gate, not shadowed, not out of range, and our kinematics is verified then what happened to the target? It may have maneuvered such that its RCS decreased. If it's a ground slow moving target it may have stopped. It may be hidden by a tunnel. The level of detail for examining why a target track is undetectable needs to be perused dependent upon the target, the environment, the amount of detail a priori data available, and the scenario under investigation. For this iteration of our AI logic structure we have elected to halt our level of investigation and to coast the target. The algorithm would request the KBC to reduce the detection level for the location which we lost the target and the locations where we project the track to be for the next four CPIs. We should identify that the track is potentially dropped and treat the track as a coasted track. If after four CPIs it cannot be correlated with a detection then the tracking filter will drop the track.

Summary

This paper has provided a brief overview of a hypothesized integrated end-to-end radar signal and data processing chain. The majority of the paper described a tracking algorithm and proposed an AI logic structure for incorporating rules for different targets, environments, and scenarios. The driving force of this logic structure is to use AI to learn about each track and to analyze each track completely before it is dropped. The logic structure is independent of any tracking algorithm, environment, target type, or scenario.

The AIRS architecture is new and revolutionary. Its potential is great. It is one element in a bigger program dealing with waveform diversity and sensors as robots.

References

[1] Bryant, M., Johnson, P., Kent, B. M., Nowak, M, and Rogers, S., "Layered Sensing Its Definition, Attributes, and Guiding Principles for AFRL Strategic Technology Development"

[2] W. Baldygo, M. Wicks, R. Brown, P. Antonik, G. Capraro, and L. Hennington, "Artificial intelligence applications to constant false alarm rate (CFAR) processing", Proceedings of the IEEE 1993 National Radar Conference, Boston, MA, April 1993.

[3] R. Senn, "Knowledge Base Applications To Adaptive Space-Time Processing", AFRL-SN-TR-146, Final Technical Report, July 2001.

[4] P. Antonik, H. Shuman, P. Li, W. Melvin, and M. Wicks, "Knowledge-Based Space-Time Adaptive Processing", Proceedings of the IEEE 1997 National Radar Conference, Syracuse, NY, May 1997.

[5] M. C. Wicks, W. J. Baldygo, JR., and R. D. Brown (1996), Expert System Constant False Alarm Rate (CFAR) Processor, U. S. Pat. 5,499,030.

[6] Multi-Channel Airborne Radar Measurement (MCARM) Final Report, Volume 1 of 4, MCARM Flight Test, Contract F30602-92-C-0161, for Rome Laboratory/USAF, by Westinghouse Electronic Systems.

[7] C. T. Capraro, G. T. Capraro, D. D. Weiner, and M. Wicks, "Knowledge Based Map Space Time Adaptive Processing (KBMapSTAP)," Proceedings of the 2001 International Conference on Imaging Science, Systems, and Technology, June 2001, Las Vegas, Nevada.

[8] A. Farina, H. Griffiths, G. Capraro, and M. Wicks, "Knowledge-Based Radar Signal & Data Processing", NATO RTO Lecture Series 233, November 2003

[9] Capraro, C. T., Capraro, G. T., Bradaric, I., Weiner, D. D., Wicks, M. C., Baldygo, W. J., "Implementing Digital Terrain Data in Knowledge-Aided Space-Time Adaptive Processing", IEEE Trans, on Aerospace and Electronic Systems, Vol. 42, No. 3, July 2006.

[10] C. T. Capraro, G. T. Capraro, and M. C. Wicks, "Knowledge Aided Detection and Tracking", Proceedings of the IEEE 2007 National Radar Conference, Boston, MA, April 2007

[11] G. Capraro and M. Wicks, "An Airborne Intelligent Radar System", Radar 2004, International Conference on Radar Systems, Toulouse France, 2004.

[12] W. L. Melvin, M. C. Wicks, and P. Chen (1998), Nonhomgeneity Detection Method and Apparatus For Improved Adaptive Signal Processing, U. S. Pat. 5,706,013.

[13] G. Capraro and M. Wicks "Metacognition for Waveform Diverse Radar", *International Waveform Diversity and Design Conference*, Kauai, Hawaii, January 2012

[14] G. Capraro, "Cognitive RF Systems and EM Fratricide – Part II", Proceedings of the 2013 International Conference on Artificial Intelligence, Volume I, pp 172-176, Las Vegas, Nevada USA, July 22-25, 2013

[15] M., L., Skolnik, "Introduction to Radar Systems", McGraw Hill, New York, 1980.

[16] E., Brookner, "Tracking and Kalman Filtering Made Easy", Wiley, New York, 1998.r\