Cognitive Radar Networks and Integrated Radar Modes

Nathan A. Goodman
University of Oklahoma, Advanced Radar Research Center & School of ECE
3190 Monitor Avenue
Norman, OK 73019
UNITED STATES

goodman@ou.edu

ABSTRACT

In this paper, we apply principles of Bayesian representations of radar environments to several applications including adaptive beamsteering for wide-area surveillance, cognitive radar networks, and adaptive parameter selection for track-before-detect modes. The goal of the Bayesian representations is to enable information-based metrics on which to optimize radar illumination parameters, thereby improving performance and enabling efficient use of finite radar resources. We describe how these applications can be cast in the new framework, updated as data are collected, and used to optimize search and waveform parameters.

1.0 INTRODUCTION

As described in another lecture series paper titled "Bayesian Representations and Information Metrics for Cognitive Radar," [1] traditional systems do not often make efficient use of the constrained resources at their disposal. Radar systems, and other sensors, often collect data that is irrelevant or non-informative for a particular task because sensor design and data collection are defined in a rigid way that prioritizes data collection, not necessarily information collection. However, this approach means that sensing resources are applied toward the collection of non-informative data. Because sensing resources such as transmit power, radar timeline, and data bandwidth are finite, we should instead be considering how to develop sensors capable of adjusting their measurements to maximize performance.

This behaviour is the essence of cognitive radar. In order to improvement performance, we wish to adapt transmitter properties such as transmit beam direction, waveform shape, and pulse repetition frequency in real time such that the radar is effectively used at all times. For example, if waveform spectra can be adaptively shaped to maximize target discrimination capability, then discrimination can be performed with reduced power and/or timeline, thereby freeing the radar to perform additional tasks. On the other hand, the radar system must have a metric on which to adapt, and this metric must have a dependence on the radar transmit parameters. The metric should be impacted by the propagation environment in which the radar operates, and this environment should be evolving as data are collected in order to continue informing the adaptive transmitter.

Using a metric on which to adapt, combined with a forward sensing model that predicts this metric for specific selections of transmit parameters, we can arrive at optimization schemes that begin to produce cognitive-like behaviour in radar systems. The behaviour is cognitive in the sense that knowledge of the radar sensing task(s) and propagation environment are updated as data are collected in order for an adaptive transmitter to optimize to the current state of affairs. When little prior knowledge is available, the adaptive transmitter typically emits a standard, general-purpose waveform. When strong prior knowledge is available, or when the environment

characteristics have been learned on previous transmissions, the adaptive transmitter can fine tune a waveform to achieve exploitation tasks more rapidly or with reduced power.

The development and updating of such representations is difficult and requires significant computational power. In this paper, we describe progress with applying our philosophy to different applications. First, we represent a three-dimensional target parameters space as a volume of resolution cells, each assigned a probability of target presence. We then select the beamsteering parameters such that the radar illumination falls on the region of target parameter space where there is the most to be learned. We also describe how detection and tracking functions can be integrated into a single framework, thereby automating the scheduling of search dwells and tracking dwells according to the radar system's most pressing need. Next, we expand this approach to include two radar systems observing a target area from different locations and describe how the probability space can be updated using the data from each individual radar system. Finally, we describe some of the difficulties of this gridded-probability-space approach, and describe an alternative approach using principles from the literature on track-before-detect methods. We demonstrate this approach through adaptive selection of the pulse repetition frequency (PRF) in order to avoid clutter zones and blind zones, which maximizes the information content of a particular coherent processing interval (CPI).

After describing these applications and providing some simulated performance comparisons, we make some comments regarding application of these principles to cognitive multi-platform space-time adaptive processing.

2.0 ADAPTIVE BEAMSTEERING FOR WIDE-AREA SURVEILLANCE

The goal of our adaptive beamsteering application is to use information-based metrics to steer the radar beam in an efficient way, which implies steering the beam toward an area where the most information is expected to be gained. We will divide the target parameter space (e.g., angle, range, Doppler, …) into a volume of resolution cells, with each cell containing a probability of target presence. When this probability is 0.5, then there is significant uncertainty that merits a look from the radar. When the probability is near 0.0 or 1.0, then there is little uncertainty, and a look from the radar is not merited. Of course, in general the radar beam will illuminate many such cells, such that we accumulate the expected information gained over all illuminated cells for a given beam position. The beam position that has the highest expected SNR gain is the one that will be used for the next illumination. Although in this section of the paper we ignore ground clutter, the expected information gain from a particular target parameter cell will depend on whether we also expect clutter to be present in that cell, which impacts the signal-to-interference ratio. In this section, we will describe the geometry used to demonstrate the operational principles, describe the calculations for determining the best beam position, and present some simulated results that demonstrate the improved performance of adaptive beamsteering over a conventional raster scan.

2.1 Geometry

We consider a two-dimensional array looking down at a target surface as depicted in Figure 1. The array elements are located in the x-y plane, which is oriented such that the array normal is looking down at an angle onto the target surface. A cross-section view of the geometry is available in the right panel of Figure 1. From the cross-section view, we can see that the target's k_y location on the surface produces a different angle of incidence with respect to the array's y-axis; hence, it changes the signal's spatial frequency with respect to the array's y-dimension (which is why it has been labelled k_y). The target's location in the orthogonal direction on the surface corresponds to a change in the signal's spatial frequency with respect to the array's x-dimension, so it has been labelled as k_x. For simplification, we assume a far-field, sidelooking type of geometry such that we can assume

that the k_x and k_y components are essentially orthogonal over the field of view. We can also see that if the target has velocity component in the k_x dimension, that this produces no radial velocity and no Doppler shift. Target motion in the k_y dimension, however, does produce a radial component that depends on the actual elevation angle. To continue keeping things general and simple, we equate the geometry-dependent relationship between Doppler shift and the rate of change in k_y through a constant factor, $-\beta$. The value of this factor isn't relevant for our current purposes. And finally, although the geometry clearly implies a look-down scenario onto a ground surface, we ignore clutter for the time being in order to focus on probability updates and beamsteering calculations.

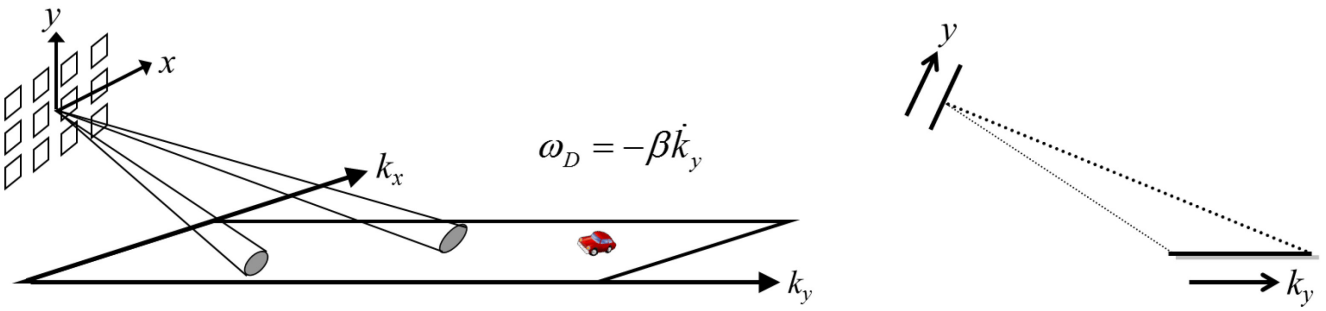

Figure 1: Geometry for developing and demonstrating adaptive beamsteering principles.

As the array steers its illumination beam around the surface of interest, we can characterize its illumination in terms of the axes k_x and k_y via the notation $S\left(k_x, k_y\right)$. We use this notation because it makes a connection with the literature on matched illumination [2-4], where the temporal spectrum of a waveform is designed to enhance SNR, discrimination ability, and/or mutual information. In general, one could consider designing an arbitrary illumination pattern $S\left(k_x, k_y\right)$ that maximizes information, similar in concept to optimizing a temporal waveform. Here, we assume a standard illumination pattern over a fixed and contiguous region of cells. Given a particular illumination pattern and a target with parameters as in the geometry above, the target will produce a signal at the array according to

$$r\left(x, y, t\right) = \alpha S\left(k_x, k_y\right) \exp\left[-j\left(k_x x + k_y y + \omega_D t\right)\right]. \tag{1}$$

where α wraps up all amplitude terms related to the radar equation and target RCS.

2.2 Scene Representation

Now that (1) describes the signal received due to a target in the scene, we must develop a method to represent potential targets in the scene. In other words, we require some way of representing what we believe to be true about the scene, such that we can optimize the illumination direction for maximum benefit. However, developing this representation poses a serious challenge. The target parameter space is continuous, and the number of targets in the scene is unknown. Therefore, there are an infinite number of possible channel realizations to represent and discriminate between, which is computationally impractical.

As a potential solution, we choose to divide the target parameter space into cells as shown in Figure 2. Each cell in the target parameter space receives a probability of target presence, and this probability is updated whenever a

measurement is taken. There is still a binary hypothesis inference problem with a maximum of 1 bit of uncertainty in each cell, but now there are many cells. This *discretized* parameter space still results in a huge number of possible channel realizations, but if the parameter cells are resolvable by the radar system, then they can be updated independently of one another. For cells that are unresolved or ambiguous (e.g., due to Doppler aliasing), then special care is needed, which is described in the next section in the context of radar networks.

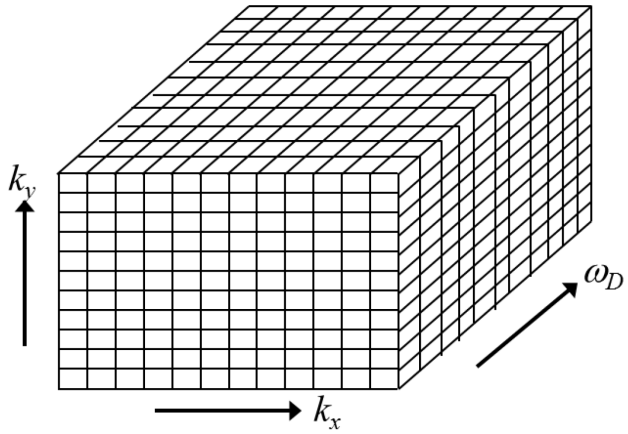

Figure 2: Discretized representation of the target parameter space. Each cell contains a probability of target presence.

In summary, each cell in the target parameter space has a probability associated with it. The radar system will illuminate a subset of cells localized in the (k_x, k_y) dimensions, but there is no ability to illuminate only a subset of Doppler bins. For those cells that are illuminated, likelihood calculations and Bayes' theorem will be used to update the cell probabilities based on an assumed model. For cells that aren't illuminated, the probabilities will be updated based on an assumed dynamic model.

The next issue concerns how to convert this volume of probabilities into a beamsteering decision. The probability volume is three-dimensional, but the beampattern is two-dimensional, so we must reduce dimensions from the parameter space to the optimization metric. Treating the decision in each cell as independent, our approach will be to accumulate the uncertainty over Doppler to obtain an entropy metric for each (k_x, k_y) cell, and then to select the beam position that covers the most entropy. Let the probability of target presence in the *i*th Doppler bin at the spatial coordinates (k_x, k_y) be denoted as $P\left(i;k_x,k_y\right)$ and the signal amplitude in the *i*th Doppler bin as α_i, our entropy calculation for the *i*th bin is

$$\left|\alpha_i\right|^2 P\left(i;k_x,k_y\right)\left[1-P\left(i;k_x,k_y\right)\right] \tag{2}$$

and the accumulated entropy over all Doppler bins at a particular (k_x, k_y) location is

$$\sigma_H^2\left(k_x,k_y\right)=\sum_{i=1}^{N_D}\left|\alpha_i\right|^2 P\left(i;k_x,k_y\right)\left[1-P\left(i;k_x,k_y\right)\right]. \tag{3}$$

The metric in (2) is based on calculating the variance of the signal strength over the two hypotheses – signal present and signal absent. We arrive at a variance that is maximized when the target probability is 0.5, and minimized when the probability is 0.0 or 1.0. This behaviour is similar to the behaviour for a binary random

variable, but the equation is different. Instead of computing entropy directly, we compute signal variance, which not only depends on the target-present probability, but also on the expected strength of the signal if it is present. For a Gaussian random variable, entropy is directly related to its variance, and the mutual information extracted from a noisy observation of the variable is related to $\log(1+SNR)$ where SNR is the ratio of the variable's variance to the noise power in the observation. If we desired to factor clutter into the calculation, we could place the noise+clutter power expected in a particular cell as the denominator in the SNR, and the variance in (2) as the numerator. Thus, the information extracted from any given cell would depend on the signal strength (through α), the target present probability (through the dependence of (2) on target probability), and the noise+clutter power. We will explicitly factor in clutter when we consider adaptive PRF selection later in the paper, but for now we assume that the interference in every cell is due to noise and has the same power. We compute the variances in (2) and accumulate over all Doppler bins at a given spatial location to achieve the 2-D variance metric in (3). This variance is called the spectral variance [4] because it depends on spatial frequencies k_x and k_y. We then substitute this variance into the mutual information equation for a Gaussian random variable, and then choose the beam position that is expected to capture the most information.

The description above describes how we select a beam position at a given instant in time. However, any scene of practical interest will be dynamic, such that targets can appear or disappear. Targets must be tracked, and areas that may have been previously "cleared" of targets must be periodically checked again. To implement these features, we institute a dynamic model for the target probabilities. First, this dynamic model allows for target probabilities to gradually converge back to a pre-defined steady state value if not updated by the radar system. This steady state value is the same target probability that is used prior to taking any measurements and should be related to the expected density of targets in the scene. Whenever a cell is not updated, the probability associated with the cell decays back to the steady-state value according to an exponential decay with a specified time constant. Areas in the search space can be assigned different time constants, such that some areas are deemed more dynamic and other areas less dynamic, reflecting prior knowledge of terrain types and high activity areas. In our simulations to be described shortly, we assumed that the spatial edges of the scene were more dynamic due to the possibility of targets entering or leaving the scene.

For targets in the scene, the radar system might determine their location by finding a cell with high probability of target presence. Sometime later, however, the target will have moved and its position will be less certain. Therefore, we allow the probability to diverge in a Kalman-like expansion of the estimated target state. We take the variance of the target parameter estimate associated with a single cell and propagate it forward in a sequential Bayesian calculation, and then take the original cell probability and re-distribute it over the cells covered by the covariance of the state vector [5]. Ideally, we would do this for every cell in the parameter space, but due to computational reasons we only perform this probability propagation for cells that exceed a defined threshold. We are, in effect, making soft detection declarations in order to propagate forward the cells that are mostly likely to have targets. Every other cell is updated using the dynamic model described in the prior paragraph.

2.3 Simulation

We generated a simulation based on the principles above, and sample results are shown in Figure 3. In the simulation, we set up a grid of cells as described, and randomly generated targets with random parameters moving through the scene. At each iteration, we compute the spectral variance at every spatial location, which is shown in the right panel of each figure pair. Based on this spectral variance, the best beam location was selected and is shown in the left panel of each figure pair.

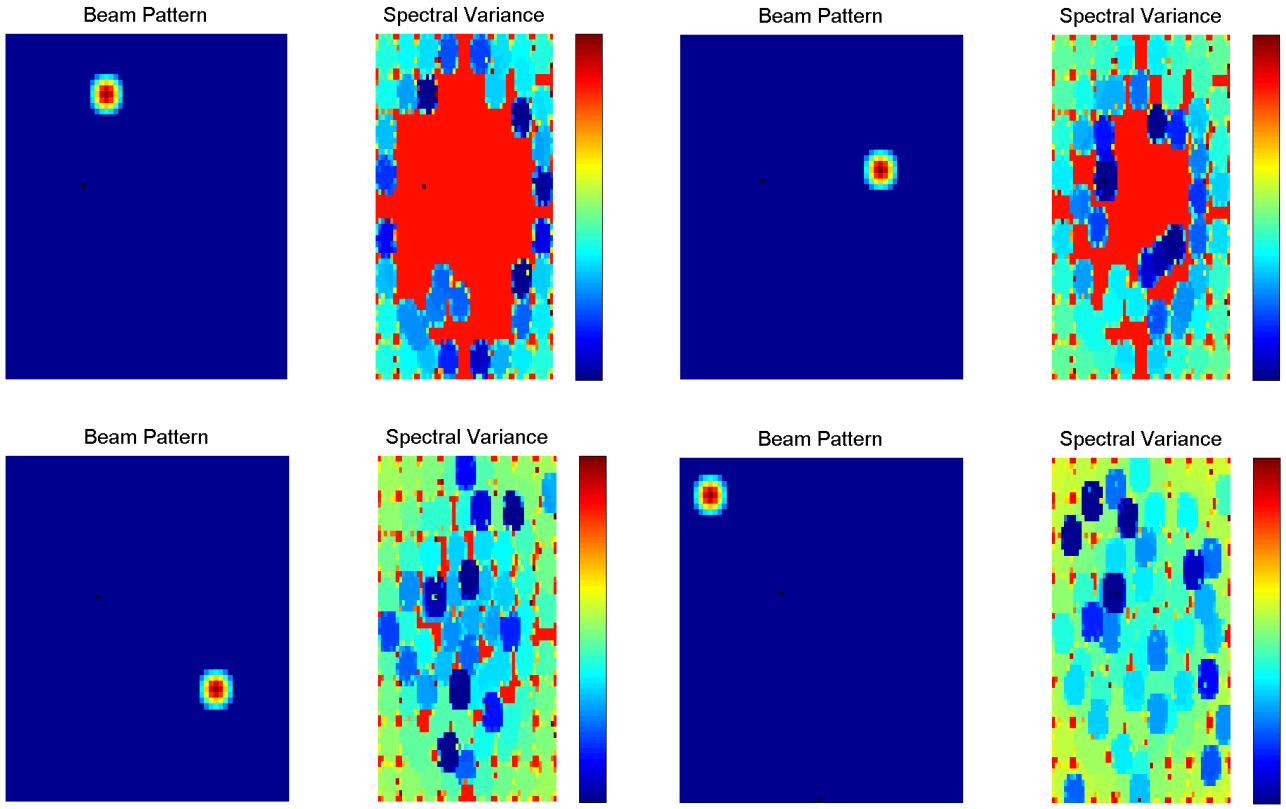

Figure 3: Spectral variance and beam locations for an adaptive search application. The results are shown after frame #40 (top left pair), frame #60 (top right); frame #90 (lower left), and frame #120 (lower right).

The scenario after the 40th illumination is depicted in the top left pair of figures. The spectral variance plot shows that the radar has been searching the edges of the target space (because these areas are the most dynamic). The highest spectral areas, color-coded as red, are in the center of the scene and haven't been searched yet. The most recently searched areas have the least uncertainty and are color-coded as dark blue. Areas that were searched several dwells ago have started to converge back to their steady-state uncertainty levels and now show up as light blue with maybe a hint of green. The true target location, seen as a black speck in the center left, has not yet been illuminated. Also note that the beam locations have been largely non-overlapping, since this would yield redundant information rather than maximum information. Because the beam shape is rounded, this has left gaps in the searched space that will need to be updated later.

The scenario after the 60th illumination is shown in the top right. The target has been discovered and updated a couple times already, yielding dark blue (i.e., no uncertainty) in the search region around the target. More of the space has been searched, and some of the more stale locations have some hints of yellow emerging. After the 90^{th} illumination (lower left), the system is now filling in some of the gaps and periodically updating the scene edges while adaptively balancing against the needs for track updates. One can see that the beam position is about to update one of the larger areas of red remaining in the variance map. Finally, after 120 iterations, the situation shown at bottom right indicates a system actively searching the outer regions of the target space again and actively choosing the best location at any given time. We note that the time between track updates is not fixed, but rather is a function of the overall state of affairs at a given time, as quantified by the spectral variance metric.

Figure 4 shows additional results for the adaptive beamsteering scenario. We performed Monte Carlo simulation of many trials of the system, allowing targets to randomly pass through the scene. The left panel of Figure 4 shows the detection probability performance for adaptive beamsteering compared to a static raster scene as a function of SNR (here, SNR is prior to processing integration gain, so the values are low). The adaptive scanning strategy clearly has better detection performance, especially at lower and mid-level SNR values. The right figure shows the accumulated energy placed on every spatial cell in the scene over a single trial of the simulation. We can see that the outer edges of the scene generally received more energy, and we can see three targets that were tracked through the scene.

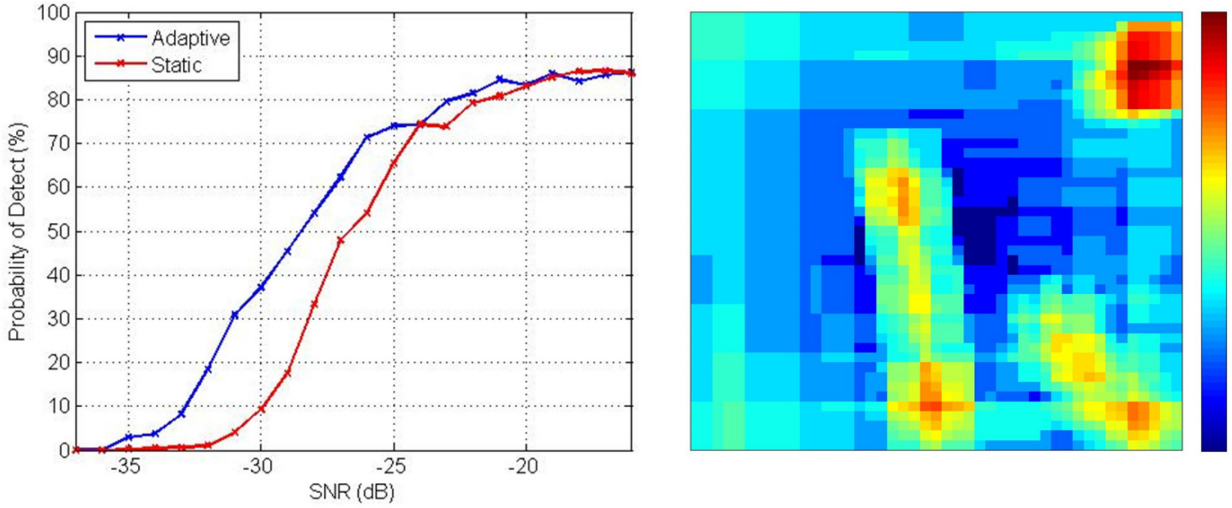

Figure 4: Detection performance (left) and sample energy distribution (right) for the adaptive beamsteering scheme.

3.0 COGNITIVE RADAR NETWORK

Based on the same principles as the adaptive beamsteering in Section 2, in this section we discuss the adaptive observation of a target space from multiple radar platforms. In such a scenario, the different platforms must jointly update the target space representation, but each in its own geometry-dependent way. For any particular radar platform, some target parameters will be observable by the platform and some will not. For example, in the scenario considered here, two radar platforms observe a scene from a relative 90-degree offset between the two, meaning that each radar can observe an orthogonal component of the target's velocity. Many of the details for this application are presented in [6], so here we only emphasize some of the most important features.

One of the major aspects of a radar network application is that different platforms observe the scene from different angles. Because range and Doppler shift are geometry dependent, this means that the target parameter space should no longer have range and/or Doppler has a parameter, but should instead be referenced to absolute positions and absolute velocities. Restricting targets to exist on a flat surface for our current application, this means that there are two position parameters and two velocity parameters, for a four-dimensional target representation space. However, each radar cannot uniquely observe the full four-dimensional space – some of the parameters in the space are unresolved, or ambiguous, to a radar platform. But the ambiguity is different for each platform. The platforms must cooperate to form a full picture of the target space, but updating the probabilities requires special attention to deal with unobserved target parameters. Our approach is to form joint

hypotheses for the different alternatives that could be true in the unobserved space, to update those joint hypotheses with data, and then to reduce back down to cell probabilities using marginal probability calculations. Details are given in [6-7].

Another salient difference in how we approached the cognitive radar network problem is in how we handled the built-in integration between search and track functions. In the previous examples, we allowed the probability within a cell to spread using sequential Bayesian updates, and then re-integrated the probability that fell within each target cell. Whenever this situation resulting in enough uncertainty, the system would automatically choose a beam position that enabled an update of that likely target. In the cognitive radar network, however, we instead chose to declare soft detections whenever a cell probability exceed a threshold. For any soft detections, we began tracking the target and directly used the entropy of the track's error covariance matrix to determine when an updated was needed. Thus, we calculated two different metrics. The variance metric based on cell probabilities was called the beam position entropy (BPE) and was used to control the search function. The metric based on track error covariance was called target track entropy (TTE) and was used to control track updates. Because these are fundamentally different types of entropy – one deals with a discrete binary decision while the other deals with estimating a continuous variable – we chose to combine the two metrics through a scale factor. Therefore, our optimization metric had the form

$$\mathrm{BPE} + \Omega \cdot \mathrm{TTE} \tag{4}$$

where Ω is the combining constant. From (4), it is easy to see that the relative emphasis placed on search versus track tasks can be varied with the constant. If $\Omega = 0$, then the target track entropy will not factor into the optimization metric and tracking will not be considered. If Ω is large, then tracking will be emphasized over finding new targets. Although we did not try this approach, it is also possible to use different constants for different targets in order to prioritize some targets over others.

Figure 5 shows energy accumulation results for the exact same target scenario run with different values of Ω. In the top panel, we have $\Omega = 0$, and the energy accumulation is consistent with a strict searching function with higher scene dynamics near the edges. In the middle panel, we have $\Omega = 1\times10^{20}$ (which shows the dramatically different scales of the two entropy types), and a hint of a track is now discernable in the lower-right corner of the figure. The target was detected as it left the center of the scene and moved to the lower right. Although some energy is allocated to tracking this target, it is not as much as in the bottom panel, where $\Omega = 5\times10^{20}$. In the bottom panel, the emphasis on the TTE measure is emphasized and a strong allocation of energy has been used to carefully track the target.

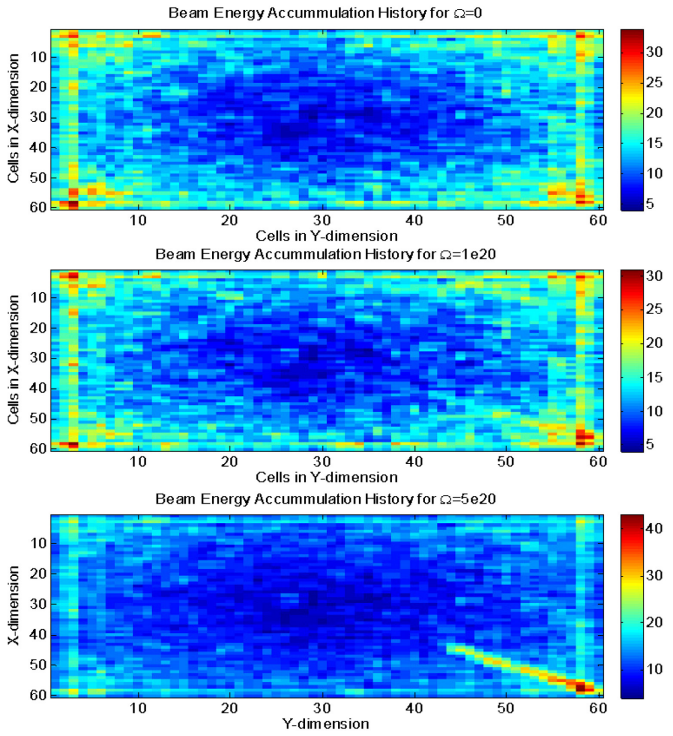

Figure 5: Accumulated energy applied to the target space as a function of the relative emphasis between searching for and tracking targets. This emphasis is varied through the variable Ω, which is used to connect the entropy metrics.

4.0 TRACK-BEFORE-DETECT WITH ADAPTIVE PARAMETER SELECTION

The two previous applications both relied on a gridded parameter space with probabilities assigned to each cell. However, through performing our tests we determined a couple drawbacks. First, when a strong target straddles two bins, the approach overestimates the uncertainty. For example, the radar might be very certain (probability near 1) that a target is present in a certain parameter cell. But a few dwells later, the target might be crossing the boundary between cells, at which point the 1 unit of probability will be split across two cells, Because each cell now has a probability of 0.5, each cell is deemed to be very uncertain and in need of update. In reality, the scenario is well determined and the inflated variance values are simply a function of the discretized parameter space.

A second difficulty is in the extension to radar networks with arbitrary observation geometries. In the cognitive network application above, we conveniently chose the two platforms to be 90 degrees apart, which allowed us to align the target parameters measured by each platform with the orientation of the gridded probability space. For platforms that do not have such convenient spacing, or that move around the scene such that their perspective changes, the gridded approach will be very inconvenient. While it is possible to arrive at ad hoc solutions for these difficulties, we desired to find a solution that was not confined to a fixed probability grid, and decided to try an approach based on track-before-detect algorithms.

Track-before-detect (TkBD) methods [8-10] are consistent with our philosophy of cognitive radar because they explicitly intend to carry over information from one illumination to another. Whereas decisions based on a single

dwell would miss many weak targets, TkBD methods track the detection metrics on a potential target over time, thereby checking for consistency over time. Many TkBD methods are based on numerical pdf representations such as particle filters; therefore, this approach enables continuous parameter values and is consistent with our desire to break away from discretized representations. Particle-based TkBD methods use clusters of particles around the estimated target state to numerical represent the state vector's pdf. From this numerical representation, the estimated state vector and error variances can be computed.

TkBD methods are available in the literature, meaning we did not have to develop the techniques from scratch. However, most of the methods are designed for a single target or for only two or three targets. Because we wish to apply the method to scenes with an unknown number of targets, we had to modify existing methods to handle a variable number of targets.

4.1 Particle-Based Representation

Here, we apply TkBD methods to the detection and tracking of targets over a wide range and Doppler space. The span of the range and Doppler space of interest precludes the absence of range and Doppler ambiguities; therefore, our observations will have blind zones (due to being blind during transmission of a pulse), mainlobe clutter zones, as well as range and Doppler aliases of these zones. An adaptive PRF selection scheme should enable optimum PRF selection such that potential target clusters do not fall into blind zones, mainlobe clutter, or onto each other.

We avoid going through all of the particle filtering steps here. Again, many of the details are covered in [11]. For now, we point out that a significant modification we made to the particle-based TkBD algorithm was to estimate the expected number of targets in the scene based on the number of clusters and their assigned probabilities of being a true target. This expected number of targets was then used to vary proportionately the number of particles used in the scene. This variability is important because it prevents strong targets from usurping all the particles from other weaker targets.

Every time data is collected and a range-Doppler map is formed, the strongest cells in the RD map are identified. Each strong cell is either associated with an existing cluster of particles (which may require checking for association through range and Doppler ambiguities) or used to generate a new cluster of particles. Each cluster has a number of particles with locations and weights that can be used to compute the potential target's parameter state. Each cluster also has a binary existence variable (similar to the detection source variable described in [1]), and a "probability of existence" (the probability that the cluster represents a true target). There are several steps to perform in the process of associating range-Doppler peaks with clusters, updating the clusters and probability of existence, resampling the scene, and so on. Although these take much effort to develop and set up, we do not cover the details here as they are part of the existing TkBD literature.

4.2 Adaptive PRF Selection Procedure

For the adaptive parameter selection, we wish to choose the waveform PRF that minimizes the expected entropy after data collection and processing. Thus, we must predict the expected posterior target probability in both cases – target present and target absent. If the target is present, we desire the probability to converge toward 1. If the target is absent, we desire the probability to converge toward 0. If the estimated target state falls into a blind zone, clutter region, or on top of another cluster, these situations will negatively impact the system's ability to drive these probabilities to a decision.

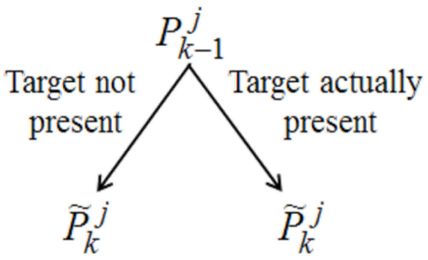

Figure 6: Predicted probability transition diagram.

We developed predicted probability transitions for several scenarios. First, if the target falls into a blind zone (or range ambiguity of a blind zone), then the target is not observed and the target's probability of existence is left unchanged. There is no change in entropy, and no information is gained. Second, if the target does not fall into a blind zone and does not alias with another cluster, then we can predict the expected SNR and, hence, the predicted posterior probability. This posterior probability depends on both SNR, and on the probability of target presence prior to the current measurement. To accommodate these situations, we built two tables based on numerical simulation. The first table lists the predicted posterior probability versus SNR and the prior probability if the target is truly present. We simulated many detection statistics for the target-present case at each SNR, and numerically estimated the average resulting posterior probability, which goes into the table. We also did the same for the target-absent case by simulating many detection statistics with the target absent. Let the probability of existence for the *j*th cluster prior to the current measurement be denoted as P_{k-1}^j, the expected posterior probability if the target is present be $\tilde{P}_k^j$, and the expected posterior probability if the target is absent be $\hat{P}_k^j$. Then the expected posterior entropies are

$$\hat{h}_0 = -\hat{P}_k^j \log_2 \hat{P}_k^j - \left(1-\hat{P}_k^j\right)\log_2\left(1-\hat{P}_k^j\right)$$

if the target is absent, and

$$\tilde{h}_1 = -\tilde{P}_k^j \log_2 \tilde{P}_k^j - \left(1-\tilde{P}_k^j\right)\log_2\left(1-\tilde{P}_k^j\right)$$

if the target is present. Finally, the predicted entropy metric is the weighted average of these two entropy calculations, where the weights are the prior probabilities that they are true, such that

$$\tilde{h} = \left(1-P_{k-1}^j\right)\hat{h}_0 + P_{k-1}^j\tilde{h}_1. \tag{5}$$

For every PRF within the radar system's capability, the metric in (5) is computed for every cluster of particles. The PRF that produces the lowest expected posterior entropy is selected for the next data collection.

Finally, we note that if multiple clusters will overlap for a particular PRF, then this ambiguity must be accounted for. We do so in the same way that we handled unobserved or ambiguous parameters in the cognitive radar network example, whereby we expand the set of joint hypotheses, and predict *their* expected posterior probabilities and expected posterior entropy. For example, suppose that two clusters will overlap for a particular PRF. There are four possibilities:

1. Neither cluster is a target
2. Only the 1st cluster is a target
3. Only the 2nd cluster is a target
4. Both clusters are targets.

The prior probabilities for each of these joint cases can be obtained from the prior existence probabilities of each cluster. As we did for the non-ambiguous case, we build a lookup table prior to simulation that can be used to lookup the expected probability transitions, but in this case the table is for the joint hypotheses. And then we use the prior probabilities for each joint hypothesis, and the expected probability transitions for each joint hypothesis, to compute the weighted average entropy.

4.3 Simulations

We developed a simulation to adaptive select between seven different PRFs. The range-Doppler space had 500 Doppler bins and 5000 range bins, and no PRF allowed for unambiguous observation of the entire space. Many trials of the simulated system were performed, with several targets randomly generated in the scene for each trials. We evaluated performance in two different ways. In the first method, we generated four random targets per trial, and let the simulation iterate until the probability of existence for all four true target clusters reached 0.9 and all other clusters had a probability below 0.9. In this case, if we were to apply 0.9 as the detection threshold, all targets would be detected and no false alarms would occur. The number if iterations required to reach this state is random, and therefore differed on each trial. We averaged the number if iterations required to converge as a function of target SNR (after processing gain). These results are shown in Figure 7, where we see that the adaptive PRF method requires significantly fewer iterations to converge to accurate decisions than an approach that simply cycles through the available PRFs. Figure 8 shows performance for simulations fixed at 60 iterations. After the final iteration, any cluster above a detection threshold was declared a target, and the number of true detections and the number of false alarms were counted. On the left panel, we show ROC comparisons between adaptive and cyclical PRF selection. Obviously, the ROC improves for stronger targets, but the adaptive PRF selection performs better than cyclical PRF selection at all SNRs. The improvement is most significant at the lower and mid-level SNR values. At the highest SNR value of 8.5 dB, the TkBD approach has performed well enough to detection most targets using either PRF selection scheme. For comparison, the right panel shows ROC performance for *single-shot* detection. That is, detection with no carryover of information from CPI to CPI. To produce the right panel, we applied a threshold to each individual CPI, and if the target was detected on any of the CPIs, it was deemed a detection for that simulation. Comparison of the two figures clearly shows the major benefit of TkBD methods, and then the additional benefit of optimizing the measurements to push the TkBD method to faster decisions.

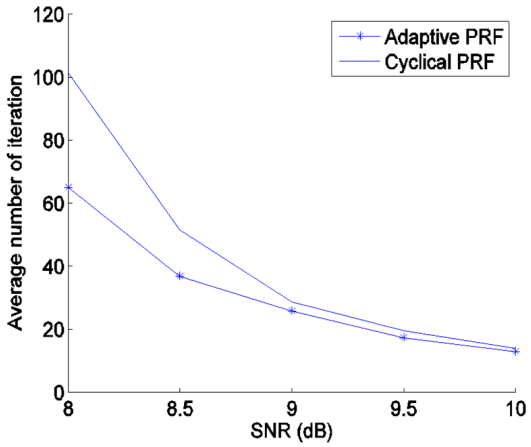

Figure 7: Average number of CPIs necessary to converge to full target detection with no false alarms. Comparison of adaptive PRF selection to cyclical PRF selection.

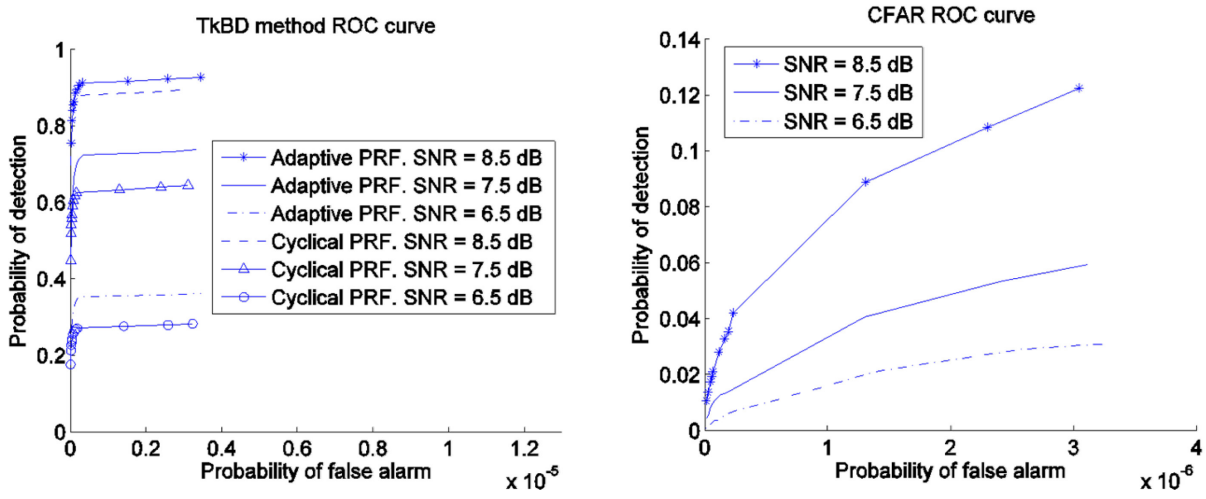

Figure 8: ROC performance curves for adaptive versus cyclical PRF selection.

5.0 CONCLUSIONS

We plan to pursue these methods further in future work. In particular, several of the elements can be combined to produce a ground-moving target indicator (GMTI) scenario for radar systems performing space-time adaptive processing (STAP). Multiple platforms can perform their STAP processing and use the results to update particle-filter-based TkBD algorithms. Thus, we plan to incorporate clutter, continuous parameter spaces, and realistic air-to-ground geometries into a cognitive GMTI network. The GMTI network application will be interesting because targets have the potential to fall into the mainlobe clutter ridge for some platforms, but not others, depending on geometry. Therefore, adaptively selecting the right platform with the right waveform parameters to update and search for targets will be an interesting optimization problem.

For now, we believe that the basic examples shown here demonstrate the potential benefit of cognitive radar systems operating on probabilistic representations of the desired target space. The representation of continuous target parameter spaces, and the infinite number of possible permutations of target presence, pose a real difficulty that must be handled in some way. To date, we have handled this through a probability grid approach and through a cluster-based TkBD approach. Both approaches have yielded nice results, though more efficient and better methods can surely be developed in the future. Continued development of efficient representations that can be updated as new data arrive will be key enablers for the future of cognitive radar.

REFERENCES

[1] N.A. Goodman, "Bayesian representations and information metrics for cognitive radar," *NATO SET-216 Lecture Series on Cognition and Radar Sensing*, 2015.

[2] M.R. Bell, "Information theory and radar waveform design," *IEEE Trans. Inf. Theory*, vol. 39, no. 5, pp. 1578–1597, 1993.

[3] D.A. Garren, M.K. Osborn, A.C. Odom, J.S. Goldstein, S.U. Pillai, and J.R. Guerci, "Enhanced target detection and identification via optimised radar transmission pulse shape," *Proc. IEEE*, vol. 148, no. 3, pp. 130-138, Jun. 2001.

[4] R.A. Romero, J.H. Bae, and N.A. Goodman, "Theory and application of SNR- and MI-based matched illumination waveforms," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 47, no. 2, pp. 912-927, April 2011.

[5] P. Nielsen and N.A. Goodman, "Integrated detection and tracking via closed-loop radar with spatial-domain matched illumination," in *Proc. 2008 International Conference on Radar*, Adelaide, Australia, pp. 546-551, Sept. 2008.

[6] R.A. Romero and N.A. Goodman, "Cognitive radar network: cooperative adaptive beamsteering for integrated search-and-track application," *IEEE Trans. Aerospace and Electronic Systems*, vol. 49, no. 2, pp. 915-931, April 2013.

[7] R. Romero, C.M. Kenyon, and N.A. Goodman, "Channel probability ensemble update for multiplatform radar systems, in *Proc. 2010 International Waveform Diversity and Design Conference*, pp. 182-187, Niagara Falls, August 2010.

[8] M.S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, No. 2, pp. 174-188, Feb. 2002.

[9] M.G. Rutten, N.J. Gordon, and S. Maskell, "Efficient particle-based track-before-detect in Rayleigh noise," *in Fusion 2004: Proc. 7th Int. Conf. on Information Fusion*, Stockhelm, Sweden, June 2004.

[10] D. J. Salmond and H. Birch, "A particle filter for track-before-detect," in *Proc. of the 2001 American Control Conference*, pp. 3755-3760, 2001.

[11] J. Bae and N.A. Goodman, "Adaptive PRF selection technique for multiple targets in track-before-detect," in *Proc. 5th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing*, St. Martin, pp. 448-451, Dec. 2013.