

# Cost-Aware Sequential Bayesian Tasking and Decision-Making for Search and Classification

Y. Wang\*, I. I. Hussein\*, D. R. Brown III<sup>†</sup>, and R. S. Erwin<sup>‡</sup>

**Abstract**—This paper focuses on the development of a cost-aware sequential Bayesian decision-making strategy for the search and classification of multiple unknown objects within a task domain. Search and classification of multiple objects of unknown numbers are competing tasks under limited vehicle and sensory resources. This is because sensor-equipped vehicles in the system can perform either the search or classification task but not both at the same time. The decision of one task over the other may result in missing other, more important objects not yet found or missing the opportunity to classify a found critical object. In this paper we develop a cost-aware sequential Bayesian decision-making strategy for search and classification, which results in the detection and satisfactory classification of all the unknown objects in the task domain.

## I. INTRODUCTION

In a search and classification mission, an autonomous sensor-equipped vehicle searches for and classifies multiple objects distributed over a domain. The objective in a search task is to find each object and fix its position in space. The objective in a classification task is to observe each found object and collect enough information to classify it. Therefore, search and classification are two competing demands. A sensor vehicle has to decide on whether to continue searching or stop and characterize once it finds an object. This decision may be very critical in some applications as in search and rescue, where, for example, finding and analyzing a nonhuman object may come at the cost of delaying or altogether missing a live human victim. Conversely, a vehicle may come across a human victim and, at the cost of missing it, decides to continue the search task.

When considering the cost of taking each new observation, one may make a decision using a limited number of observations if the uncertainty is high. This is because, at the outset of the mission, it is costly to keep taking observations at one specific spot while ignoring a large amount of unsurveyed regions in the domain. Under these scenarios, the vehicle will make temporary decisions based on the currently available observations, which may lead to relatively large error probabilities of the decisions made, but can come back later when the observation cost is lower because more regions of the domain would

have been investigated by that time, or equivalently, the uncertainty level is lower.

We first review some of the related literature. Inspired by work on particle filtering, in [1] the authors develop a sensor team configuration control strategy under a probabilistic framework to get optimal estimates for target tracking. In [2], the authors use the Beta distribution to model the confidence level of target existence for an unmanned aerial vehicle (UAV) search task in an uncertain environment. In [3], the above uncertainty measurements are extended by using the Modified Bayes Factor, and prediction of future measurement is also taken into account to calculate the possible uncertainty reduction in UAV search operations. An alternate approach for searching/tracking in an uncertain environment is simultaneous localization and mapping (SLAM) [4].

Coordinated search and tracking in a probabilistic framework has been studied mainly for optimal path planning in the literature. In [5], the authors investigate search-and-tracking using recursive Bayesian filtering with fore-known targets' positions with noise. The target may be lost and needs to be found again due to measurement noise. The results are extended in [6] for dynamic search spaces based on forward reachable set analysis. In [7], the author proposes a Bayesian-based multisensor-multitarget sensor management scheme. The approximation strategy, based on probability hypothesis densities, maximizes the square of the expected number of targets. With the same objective, in [8] the authors seek to maximize the probability of finding a target with some foreknown location information in the presence of uncertainty. It is worth noting that in the above literature there is no explicit decision-making strategy for search and tracking.

Sequential detection [9] allows the number of observation samples to vary in order to achieve an optimal decision. In our problem, with a relatively high observation cost at the beginning of the mission, it is wise to make a crude decision with fewer observations first, and return to update the decision later when the cost is low. The Bayesian sequential detection method used in this paper is such that the Bayes risk (to be formally defined in Section V) is minimized in each time step. Another sequential detection method is the Sequential Probability Ratio Test (SPRT) [10], [9] based on Neyman-Pearson formulation where no prior probability information is needed.

The paper is organized as follows. We first introduce the sensor model in Section II. Combined with a real time observation, the posterior probabilities are updated according to the Bayes rule in Section III-A. In Section III-B,

\*Mechanical Engineering Department, Worcester Polytechnic Institute, 100 Institute Road, Worcester, MA 01609. E-mail: {yuewang, ihussein}@wpi.edu.

<sup>†</sup>Electrical and Computer Engineering Department, Worcester Polytechnic Institute, 100 Institute Road, Worcester, MA 01609. E-mail: drb@wpi.edu.

<sup>‡</sup>Air Force Research Laboratory, Space Vehicles Directorate AFRL/RV, 3550 Aberdeen Ave. SE, Kirtland AFB, NM 87117.

an uncertainty map for the search process is built based on the probability of object presence over the domain. A classification uncertainty function is also defined for each found object. Based on the uncertainty function, we define metrics for search and classification in Section IV. In Section V, we develop the cost-aware sequential Bayesian decision-making strategy. To illustrate the performance of our approach, in Section VI we provide a simulation for a single cell in a search task. The algorithms for motion control of the autonomous vehicle is developed in Section VII and a full-scale simulation over the entire mission domain for search versus classification is presented in Section VIII. We conclude the paper with a summary of current and future work in Section IX.

## II. SETUP AND SENSOR MODEL

Let  $\mathcal{D} \subset \mathbb{R}^2$  be a domain in which objects to be found and classified are located. Let  $\tilde{\mathbf{q}}$  be an arbitrary point in  $\mathcal{D}$ . Assume there are  $N_{\text{tot}}$  cells in  $\mathcal{D}$ . Let  $1 \leq N_o \leq N_{\text{tot}}$  be the number of objects, which are spatially i.i.d. over  $\mathcal{D}$ . Denote the position of the static object  $\mathcal{O}_j$ ,  $j \in \{1, 2, \dots, N_o\}$  as  $\mathbf{p}_j$  and  $\mathcal{P}$  as the set of all object positions. Both  $N_o$  and  $\mathcal{P}$  are unknown beforehand.  $N_o$  is a binomial random variable with parameters  $N_{\text{tot}}$  and  $\text{Prob}(\tilde{\mathbf{q}} \in \mathcal{P})$ , where  $\text{Prob}(\tilde{\mathbf{q}} \in \mathcal{P})$  is the probability of object presence at point  $\tilde{\mathbf{q}}$ . We assume that there exists a single autonomous sensor-equipped vehicle (denoted by  $\mathcal{V}$ ) that performs the search and classification tasks. The vehicle  $\mathcal{V}$  satisfies the following simple first order discrete-time equation of motion

$$\mathbf{q}(t+1) = \mathbf{q}(t) + \mathbf{u}(t),$$

where  $\mathbf{q} \in \mathcal{D} \subset \mathbb{R}^2$  represents the position of  $\mathcal{V}$ , and  $\mathbf{u} \in \mathcal{U} \subset \mathbb{R}^2$  is the control input, where  $\mathcal{U}$  is the set of allowable controls. At any time  $t$ , the vehicle can either perform the search task or the classification task, but it is not capable of performing both at the same time.

In this work, for both the search and classification processes, we use a sensor model with Bernoulli distribution, which gives binary outputs for a single observation, however, with different observation contents: object ‘‘present’’ or ‘‘absent’’ for search, and property ‘‘F’’ or ‘‘G’’ for classification.

### A. Sensor Model for Search

In the search process, let  $X_s(\tilde{\mathbf{q}})$  be a binary state variable, where 0 corresponds to object absent, and 1 corresponds to object present. Note that the realization of  $X_s(\tilde{\mathbf{q}})$  depends on the position of the observed point  $\tilde{\mathbf{q}}$ , that is,

$$X_s(\tilde{\mathbf{q}}) = \begin{cases} 1 & \tilde{\mathbf{q}} \in \mathcal{P}, \\ 0 & \text{otherwise.} \end{cases}$$

Since  $\mathcal{P}$  is unknown and random,  $X_s(\tilde{\mathbf{q}})$  is a random variable with respect to every  $\tilde{\mathbf{q}} \in \mathcal{D}$ . Since we assume that the objects are immobile,  $X_s(\tilde{\mathbf{q}})$  is invariant with respect to time. Define the observation indicating object present as a positive observation and the observation indicating object absent as a negative observation. Let  $Y_s(\tilde{\mathbf{q}})$  be a binary observation variable, where 0 corresponds to a negative

observation, and 1 corresponds to a positive one taken at point  $\tilde{\mathbf{q}}$ .

Conditioned on the state  $X_s(\tilde{\mathbf{q}})$  at a particular point  $\tilde{\mathbf{q}}$ , let  $t$  be time index, the observations  $Y_{s,t}(\tilde{\mathbf{q}})$  taken along time are temporally i.i.d. Therefore, if take an observation at each time step at  $\tilde{\mathbf{q}}$ , for a window of  $L$  time steps, there will be  $L + 1$  different combinations of unordered scalar observations, that is, ranging from zero positive observation to  $L$  positive ones. Let the variable  $Z_s(\tilde{\mathbf{q}})$  be the number of positive observations at point  $\tilde{\mathbf{q}}$ , which is a number in the set  $\{0, \dots, L\}$ . The following  $(L + 1) \times 2$  matrix gives the general conditional probability matrix for the search task:

$$B_s = \begin{bmatrix} \beta_s^L & (1 - \beta_s)^L \\ L(1 - \beta_s)\beta_s^{L-1} & L\beta_s(1 - \beta_s)^{L-1} \\ \vdots & \vdots \\ (1 - \beta_s)^L & \beta_s^L \end{bmatrix}, \quad (1)$$

where  $\beta_s$  is the parameter of the Binomial distribution  $(L, \beta_s)$ , which describes the sensor probability of making a correct observation. The matrix element  $(B_s)_{ij}$  corresponds to  $\text{Prob}[Z_s = i | X_s(\tilde{\mathbf{q}}) = j]$ , i.e., the probability of having  $i$  positive observations given state  $j$ .

The value of  $\beta_s$  depends on the range between the sensor and the observed point. Here we assume a simple model for  $\beta_s$  that is a fourth order polynomial function of  $s = \|\mathbf{q}(t) - \tilde{\mathbf{q}}\|$  within the sensor range  $r_s$  and  $b_n = 0.5$  otherwise,

$$\beta_s(s) = \begin{cases} \frac{M_s}{r_s^4} (s^2 - r_s^2)^2 + b_n & \text{if } s \leq r_s \\ b_n & \text{if } s > r_s \end{cases}, \quad (2)$$

where  $M_s + b_n$  gives the peak value of  $\beta_s$  if  $\tilde{\mathbf{q}}$  being observed is located at the sensor vehicle’s location. The sensing capability decreases with range and becomes 0.5 outside of the limited sensory range  $\mathcal{W}$ .

### B. Sensor Model for Classification

In the classification process, let  $X_c(\tilde{\mathbf{p}}_k)$  be a binary state variable for object  $k$ , where 0 corresponds to object having Property ‘‘G’’, and 1 corresponds to Property ‘‘F’’. Let  $Y_c(\tilde{\mathbf{p}}_k)$  be a binary observation variable, where 0 corresponds to observation showing Property ‘‘G’’, and 1 corresponds to observation showing Property ‘‘F’’. For a window of  $L$  time steps, let  $Z_c(\tilde{\mathbf{p}}_k)$  be the number of observations showing Property ‘‘F’’ for object  $k$ , the following  $(L + 1) \times 2$  matrix gives the general conditional probability matrix for the classification task:

$$B_c = \begin{bmatrix} \beta_c^L & (1 - \beta_c)^L \\ L(1 - \beta_c)\beta_c^{L-1} & L\beta_c(1 - \beta_c)^{L-1} \\ \vdots & \vdots \\ (1 - \beta_c)^L & \beta_c^L \end{bmatrix}.$$

The value of  $\beta_c$  follows the some form as  $\beta_s$  but with parameters  $M_c$  and  $r_c$ .

## III. BAYESIAN UPDATES AND UNCERTAINTY MAP

In this section, we summarize the main algorithms presented by the authors in [11].

### A. Bayesian Updates for Search and Classification

For the search process, the general update equation for the probability of object presence is:

$$P_s(\tilde{\mathbf{q}}, t+1) = y_{s,t}(\tilde{\mathbf{q}}) \frac{\beta_s P_s(\tilde{\mathbf{q}}, t)}{2\beta_s P_s(\tilde{\mathbf{q}}, t) - \beta_s - P_s(\tilde{\mathbf{q}}, t) + 1} + (1 - y_{s,t}(\tilde{\mathbf{q}})) \frac{(1 - \beta_s) P_s(\tilde{\mathbf{q}}, t)}{-2\beta_s P_s(\tilde{\mathbf{q}}, t) + \beta_s + P_s(\tilde{\mathbf{q}}, t)}, \quad (3)$$

where  $y_{s,t}(\tilde{\mathbf{q}})$  is the actual realization of the random variable  $Y_{s,t}(\tilde{\mathbf{q}})$ . Note that the probability of object absence is given by  $1 - P_s(\tilde{\mathbf{q}}, t+1)$ .

For the classification process, the general update equation for the probability of a found object  $k$  having property ‘‘F’’ is:

$$P_c(\tilde{\mathbf{p}}_k, t+1) = y_{c,t}(\tilde{\mathbf{p}}_k) \frac{\beta_c P_c(\tilde{\mathbf{p}}_k, t)}{2\beta_c P_c(\tilde{\mathbf{p}}_k, t) - \beta_c - P_c(\tilde{\mathbf{p}}_k, t) + 1} + (1 - y_{c,t}(\tilde{\mathbf{p}}_k)) \frac{(1 - \beta_c) P_c(\tilde{\mathbf{p}}_k, t)}{-2\beta_c P_c(\tilde{\mathbf{p}}_k, t) + \beta_c + P_c(\tilde{\mathbf{p}}_k, t)}. \quad (4)$$

The probability of having property ‘‘G’’ is  $1 - P_c(\tilde{\mathbf{p}}_k, t+1)$ .

### B. The Uncertainty Map

For the search process, we use the information entropy function of the probability distribution of object presence to construct an uncertainty map over the search domain. The uncertainty map will be used to guide the vehicle to regions of high uncertainty in the search domain. We define the information entropy distribution for discrete probability distribution  $P_{H_s} = \{P_s(\tilde{\mathbf{q}}, t), 1 - P_s(\tilde{\mathbf{q}}, t)\}$  (i.e.,  $P_{H_s}$  is the probability distribution for object present and absent) at  $\tilde{\mathbf{q}}$  at each time step  $t$  as  $H_s(P_{H_s}, \tilde{\mathbf{q}}, t) = -P_s(\tilde{\mathbf{q}}, t) \ln P_s(\tilde{\mathbf{q}}, t) - (1 - P_s(\tilde{\mathbf{q}}, t)) \ln(1 - P_s(\tilde{\mathbf{q}}, t))$ .

For the classification process, we define a similar entropy function  $H_c(P_{H_c}, \tilde{\mathbf{p}}_k, t)$ , with  $P_{H_c} = \{P_c(\tilde{\mathbf{p}}_k, t), 1 - P_c(\tilde{\mathbf{p}}_k, t)\}$ , for every found object  $k$  (located at  $\tilde{\mathbf{p}}_k$ ) to evaluate classification uncertainty:  $H_c(P_{H_c}, \tilde{\mathbf{p}}_k, t) = -P_c(\tilde{\mathbf{p}}_k, t) \ln P_c(\tilde{\mathbf{p}}_k, t) - (1 - P_c(\tilde{\mathbf{p}}_k, t)) \ln(1 - P_c(\tilde{\mathbf{p}}_k, t))$ . There are as many scalar  $H_c$ 's as there are found objects  $k$  up to time  $t$ .

## IV. SEARCH AND CLASSIFICATION METRICS

In this section we develop metrics to be used for the search versus classification decision-making process. We define the cost of not carrying on further search as

$$\mathcal{J}(t) = \frac{\int_{\mathcal{D}} H_s(P_{H_s}, \tilde{\mathbf{q}}, t) d\tilde{\mathbf{q}}}{H_{s,\max} A_{\mathcal{D}}}. \quad (5)$$

The cost  $\mathcal{J}$  is proportional to the total integral of the search uncertainty over  $\mathcal{D}$ . We divide the integral by the area of the domain  $A_{\mathcal{D}}$  multiplied by  $H_{s,\max}$  in order to normalize  $\mathcal{J}(t)$ . According to this definition, we have  $0 \leq \mathcal{J}(t) \leq 1$ . Initially, we have  $\mathcal{J}(0) = \frac{H_s(P_{H_s}, \tilde{\mathbf{q}}, 0)}{H_{s,\max}} \leq 1$ . If for some  $t_s$  we have  $H_s(P_{H_s}, \tilde{\mathbf{q}}, t_s) = 0$  for all  $\tilde{\mathbf{q}} \in \mathcal{D}$ , then  $\mathcal{J}(t_s) = 0$  and the entire domain has been satisfactorily covered and we know with 100% certainty that there are no objects yet to be found.

For the classification process, let  $\bar{N}_o(t)$  be the number of objects found by the autonomous sensor vehicle up to time  $t$ . For each found object  $j \in \{1, 2, \dots, \bar{N}_o(t)\}$ , define the classification metric  $H_d(\tilde{\mathbf{p}}_j, t)$  to be

$$H_d(\tilde{\mathbf{p}}_j, t) = \epsilon_c \mathcal{J}(t), \quad (6)$$

where  $\epsilon_c$  is a preset upper bound on the desired uncertainty level for classification.

Let us define the classification conditions as follows:

$$\begin{cases} \|\mathbf{q}(t) - \tilde{\mathbf{p}}_j\| \leq r_c & \text{(a)} \\ H_c(P_{H_c}, \tilde{\mathbf{p}}_j, t) > H_d(\tilde{\mathbf{p}}_j, t) & \text{(b)} \\ P_s(\tilde{\mathbf{p}}_j, t) \geq \pi_U & \text{(c)} \\ \text{No Decision about } \mathcal{O}_j \text{ from } t_{cj} \text{ up to } t & \text{(d)} \end{cases},$$

where  $\pi_U$ <sup>1</sup> is some lower bound on the probability of object presence to be met before a classification task can be carried on. The time quantity  $t_{cj}$  indicates the time instant at which the current classification loop begins. Only when all the classification conditions are satisfied, i.e., (a) the object  $j$  is within the vehicle's classification sensory range, (b) the classification uncertainty of  $j$  is larger than the desired uncertainty, (c) the probability of object present for this point is higher than some boundary probability, and (d) no decision has yet been made about the property of  $j$  from the beginning of this classification loop up to time  $t$ , then the vehicle will start to classify  $j$ . If any one of the above condition fails, the vehicle  $\mathcal{V}$  stops classifying the found object and switches to searching again. It can resume classifying an object that has been detected and completely or partially classified in the past if it finds it again during the search process. When this occurs, the value of  $H_d$  will be smaller than the last time the object has been detected.

## V. COST-AWARE SEQUENTIAL DECISION-MAKING

Assuming a Uniform Cost Assignment (UCA) [9], we define the decision cost component as  $C_{ij} = 1$ ,  $i \neq j$ ,  $C_{ii} = 0$ , where  $i = 0, 1$  represent deciding object absent ( $i = 0$ ) and deciding object present ( $i = 1$ ) and  $j = 0, 1$  correspond to state  $X_s(\tilde{\mathbf{q}}) = 0$  and  $X_s(\tilde{\mathbf{q}}) = 1$ , respectively. Hence  $C_{ij}$  is the cost of deciding  $i$  when the state is  $j$ .

Let  $\tilde{R}_0(\tilde{\mathbf{q}}, L, \Delta)$ ,  $L \geq 1$ , be the conditional Bayes risk of deciding there is an object at  $\tilde{\mathbf{q}}$  given that there is actually none over at least one observation,

$$\tilde{R}_0(\tilde{\mathbf{q}}, L, \Delta) = c_0^T \Delta b_0, \quad c_0 = [C_{00} \ C_{10}]^T. \quad (7)$$

The quantity  $b_0$  is the first column of the general conditional probability matrix  $B_s$  and contains the probabilities of having zero ( $Z_s = 0$ ) to  $L$  ( $Z_s = L$ ) positive observations when there is nothing at  $\tilde{\mathbf{q}}$ . The quantity  $\Delta$  is a deterministic decision rule. For  $L \geq 1$ ,  $\Delta$  is a  $2 \times (L+1)$  matrix. The number 2 is the number of possible final decisions in this case, corresponding to ‘‘object absent’’ and ‘‘object present’’, respectively. The quantity  $L \geq 1$  is the number of observations that one can make over a window of  $L$  time steps, and the first to  $(L+1)$ th columns in the  $\Delta$  matrix correspond to zero to  $L$  positive observations. The element  $\Delta_i^l$ ,  $i = 0, 1$ ,  $l = 0, 1, \dots, L$  can be either 0 or 1, and  $\sum_{i=0}^1 \Delta_i^l = 1$ . If  $\Delta_i^l = 1$ , it means that the vehicle will make decision  $i$  given there are  $l$  positive observations. Therefore,  $\Delta$  can have  $2^{L+1}$  different matrix values. When  $L = 0$ , i.e., there is

<sup>1</sup>Note that  $\pi_U$  is a variable with respect to  $\tilde{\mathbf{q}}$  and  $t$ . We will elaborate it in Section V.

no observation taken,  $\Delta$  could be either “always decide there is an object” or “always decide there is no object”, regardless of the observations, and there will be no explicit matrix form for  $\Delta$ .

Similarly, the conditional Bayes risk

$$\tilde{R}_1(\tilde{\mathbf{q}}, L, \Delta) = c_1^T \Delta b_1, \quad c_1 = [C_{01} \ C_{11}]^T \quad (8)$$

gives the cost of deciding that there is no object at  $\tilde{\mathbf{q}}$  given that there is actually something over  $L \geq 1$  observations.

In this paper, we assume that the sensor is a “good” one, that is to say, the detection probability is higher than the error probability of the sensor, i.e.,  $\beta_s > 0.5$ . Therefore, there are only a small number of “reasonable” deterministic decision rules. Given  $L$  observations, the set of “reasonable” deterministic decision rules is the set of all rules of the type

$$\Delta_1^l = \begin{cases} 1 & l \geq v \\ 0 & \text{otherwise} \end{cases}$$

where  $l \in \{0, \dots, L\}$  is the total number of positive observations and  $v \in \{0, \dots, L + 1\}$  is the threshold where we make a positive decision. Note that “reasonable” decision rules grows linearly with  $L$  and dominates any other type of decision rules with the same value of  $L$ .

Therefore, under UCA, there is no cost if the decision is the actual state, and the conditional risk  $\tilde{R}_0(\tilde{R}_1)$  can be interpreted as the error probability of deciding there is an (no) object given that there is actually none (one) under a certain decision rule  $\Delta$  over  $L$  observations for point  $\tilde{\mathbf{q}}$ .

Now let us assign each observation a cost  $c_{\text{obs}}(t)$ . For both the search and classification processes, we define a dynamic observation cost  $c_{\text{obs}}$  that depends on how much more search uncertainty there remains to be eliminated:  $c_{\text{obs}}(t) = \gamma \mathcal{J}(t)$ , where  $\gamma > 0$  is some constant. That is to say, at the outset of the mission with a relatively high value of  $\mathcal{J}$ , the observation cost is high since there are still many uncovered points in the domain and it is “expensive” to make an observation at the point being observed. The cost-aware Bayesian sequential decision-making strategy tends to make a decision with a few observation samples in that case, which may yield large number of false detections and missed detections. When the vehicle has surveyed most points in the domain, both the uncertainty and the observation cost decrease. The vehicle is able to revisit some already searched/classified points to take observations again and make better decisions with lower uncertainty. The process is repeated until  $\mathcal{J}(t) \rightarrow 0$  for the search task, and the classification metric  $H_c < H_d, \forall \tilde{\mathbf{p}}_k \in \mathcal{P}$  is satisfied for the classification task.

At every time step, the vehicle has to choose to (i) decide object present, (ii) decide object absent or (iii) take one more observation and postpone making any decisions regarding object presence to the following time step. This same decision procedure is repeated until the cost of making a wrong decision based on the current observation is less than that of taking one more observation for a possibly better decision. The cost-aware sequential Bayesian decision-making strategy is such that the Bayes risk at

each time step is minimized. Denote  $\phi = \{\phi_k\}_{k=0}^\infty$  to be the stopping rule and  $\delta = \{\delta_k\}_{k=0}^\infty$  the terminal decision rule. If  $\phi_k = 0$ , we take another measurement, if  $\phi_k = 1$ , we stop taking further observations. At every time step  $k$ , the quantity  $\delta_k$  can be either one of three possibilities: decide object present, decide object absent or take one more observation. However, note that the final decision is either object present or absent. Define the stopping time as  $N(\phi) = \min\{k : \phi_k = 1\}$ , which is a random variable due to the randomness of the observations. The expected stopping time under state  $X_s(\tilde{\mathbf{q}}) = j$  is then given by  $E_j[N(\phi)] = E[N(\phi)|X_s(\tilde{\mathbf{q}}) = j]$ .

Since now we assign a cost  $c_{\text{obs}}$  for each observation, the conditional Bayes risk (7,8) under UCA over  $L \geq 0$  observations can be modified to be:

$$R_i(\tilde{\mathbf{q}}, L, \Delta) = \text{Prob}(\text{decide } X_s(\tilde{\mathbf{q}}) = j | X_s(\tilde{\mathbf{q}}) = i) + c_{\text{obs}} E_i[N(\phi)], \quad i, j = 0, 1. \quad (9)$$

If  $L \geq 1$ ,  $\Delta$  has explicit matrix form and we can further rewrite the above equations as:

$$R_i(\tilde{\mathbf{q}}, L, \Delta) = c_i^T \Delta b_i + c_{\text{obs}} E_i[N(\phi)], \quad i = 0, 1. \quad (10)$$

Define the Bayes risk as the expected conditional Bayes risk of making a wrong decision under decision rule  $\Delta$  for  $L \geq 0$ :

$$r(\tilde{\mathbf{q}}, L, \pi_1, \Delta) = \pi_0 R_0(\tilde{\mathbf{q}}, L, \Delta) + \pi_1 R_1(\tilde{\mathbf{q}}, L, \Delta), \quad (11)$$

where  $\pi_1 = P_s(\tilde{\mathbf{q}}, t = t_v)$  is the prior probability of object present at point  $\tilde{\mathbf{q}}$  and  $\pi_0 = 1 - \pi_1$  gives the prior probability of object absent. Here,  $t_v$  is the time instant when the point  $\tilde{\mathbf{q}}$  is visited. Note that the prior is a fixed number ranging from 0 to 1 at time  $t_v$  and is used to construct the minimum Bayes risk curve over all possible lengths of observations. Fix a  $\pi_1 \in [0, 1]$ , the minimum Bayes risk curve at this particular prior has the minimum  $r$  value over all possible choices of  $\Delta$  with  $L \geq 0$ .

If the sensor does not take any observations ( $L = 0$ ) and directly makes a decision, the Bayes risks corresponding to the  $2^1 = 2$  different decision rules  $\Delta$  are as follows:  $r(\tilde{\mathbf{q}}, L = 0, \pi_1, \Delta = \text{always decide there is an object}) = 1 - \pi_1$ , and  $r(\tilde{\mathbf{q}}, L = 0, \pi_1, \Delta = \text{always decide there is no object}) = \pi_1$ .

If the sensor takes an observation ( $L = 1$ ) at  $t = 0$ , the minimum Bayes risk over all possible choices of  $\Delta$  is

$$r_{\min}(\tilde{\mathbf{q}}, L = 1, \pi_1) = \min_{\Delta \in \mathcal{G}_L} (1 - \pi_1) R_0(\tilde{\mathbf{q}}, L = 1, \Delta) + \pi_1 R_1(\tilde{\mathbf{q}}, L = 1, \Delta) \geq c_{\text{obs}}(1)$$

where  $\mathcal{G}_L$  is defined as the set of all deterministic decision rules that are based on exactly  $L$  observations (Here,  $L = 1$  since we have only taken one observation).

Following the same procedure, compute the minimum Bayes risk functions under different observation numbers and the overall minimum Bayes risk over all decision rules ( $L \geq 0$ ) is:  $r_{\min}^*(\tilde{\mathbf{q}}, \pi_1) = \min_{L=0,1,2,\dots} r_{\min}(\tilde{\mathbf{q}}, L, \pi_1)$ .

The basic idea of the cost-aware sequential Bayesian decision-making method is as follows: With an initial prior probability of object presence  $P_s(\tilde{\mathbf{q}}, t)$ , check its corresponding  $r_{\min}^*$  value in the overall minimum Bayes risk curve. If  $r_{\min}^*(\tilde{\mathbf{q}}, \pi_1)$  is given by the line with  $L \geq 1$ , the Bayes risk is lowered by taking an observation  $Y_{s,t}$ ,

compute the posterior probability  $P_s(\tilde{\mathbf{q}}, t+1)$  according to Equation (3) and again check its corresponding minimum Bayesian risk  $r_{\min}^*(\tilde{\mathbf{q}}, \pi_1)$  to make decision. That is, an observation is taken if and only if the prior  $\pi_1 = P_s(\tilde{\mathbf{q}}, t)$  is such that  $r_{\min}(\tilde{\mathbf{q}}, L \geq 1, \pi_1) < \min(\pi_1, 1 - \pi_1)$ . The same procedure is repeated until the Bayes risk of the taking one more observation is higher than the cost of making a wrong decision. That is to say,  $r_{\min}^*$  is given by  $r(\tilde{\mathbf{q}}, L = 0, \pi_1, \Delta)$ .

Let us illustrate the detailed scheme by the following simple single point simulation.

## VI. SIMULATION FOR A SINGLE POINT

In this simulation, we fix a point  $\tilde{\mathbf{q}}$ , choose  $\beta_s = 0.8$ , and set the observation cost as a fixed number  $c_{\text{obs}} = 0.05$  to demonstrate the sequential Bayesian-based decision rule. Figure 1(a) shows all the Bayes risk functions  $r$  under 0 (black lines), 1 (blue lines) or 2 (green lines) observations with  $\pi_1 \in [0, 1]$ . In Figure 1(b), the red line segments indicate the overall minimum Bayes risk  $r_{\min}^*(\tilde{\mathbf{q}}, \pi_1)$ . Here, we only list the equation of the lines that constitute these red line segments. The Bayes risk functions under more than 3 observations ( $L \geq 3$ ) have larger  $r$  values and do not contribute to  $r_{\min}^*(\tilde{\mathbf{q}}, \pi_1)$  for the particular choice of  $\beta_s$  and  $c_{\text{obs}}$  here.

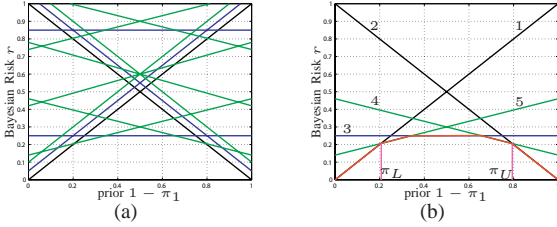


Fig. 1. (a) Bayes Risk Functions under 0, 1, 2 observations; (b) Minimum Bayes Risk Function.

The following are the expressions for the lines in Figure 1(b) annotated by the numerals 1 – 5.

**Line 1.** This line represents the decision rules without any observation. Always decide there is an object at the cell regardless of the observations. According to Equation (11),  $r(\tilde{\mathbf{q}}, L = 0, \pi_1, \Delta = \Delta_{11})$  always decide there is an object).

**Line 2.** This line also represents the decision rules without any observation. Always decide there is no object regardless of the observations:  $r(\tilde{\mathbf{q}}, L = 0, \pi_1, \Delta = \Delta_{22}) = 1 - \pi_1$ .

**Line 3.** The blue line corresponds to the decision rule 3 after taking one observation: decide the actual state according to the only one observation, that is, if  $Z_s = 1$ , decide there is actually an object. We have  $r(\tilde{\mathbf{q}}, L = 1, \pi_1, \Delta = \Delta_{11}) = 1 - \beta_s + c_{\text{obs}}$ .

**Line 4.** This line gives the decision rules after two observations. Line 4 corresponds to the decision rule that decides there is actually an object if and only if all the two observations are positive ( $Z_s = 2$ ). Following the same procedure as above, we have  $r(\tilde{\mathbf{q}}, L = 2, \pi_1, \Delta = \Delta_{21}) = (1 - \beta_s)^2(1 - \pi_1) + (2\beta_s(1 - \beta_s) + (1 - \beta_s)^2)\pi_1 + 2c_{\text{obs}}$ .

**Line 5.** This line also gives the decision rules after two observations. Line 5 corresponds to the decision rule that decides there is no object if and only if none of the two observations is object present,  $r(\tilde{\mathbf{q}}, L = 2, \pi_1, \Delta = \Delta_{22}) = (2\beta_s(1 - \beta_s) + (1 - \beta_s)^2)(1 - \pi_1) + (1 - \beta_s)^2\pi_1 + 2c_{\text{obs}}$ .

Thus, the red line segments give the minimum Bayesian risk  $r_{\min}^*(\tilde{\mathbf{q}}, \pi_1)$  over 0, 1, 2 observations. The intersection of lines 1, 5 is the lower prior probability  $\pi_L = 0.2059$ . When the posterior probability  $1 - P_s(\tilde{\mathbf{q}}, t)$  updated through Equation (3) is below  $\pi_L$  (i.e.,  $P_s(\tilde{\mathbf{q}}, t) \geq \pi_U$ , an example of the boundary probability for the classification condition (c) in Section IV), the vehicle stops taking observation and decide that the actual state is object present. The intersection of lines 2, 4 is the upper prior probability  $\pi_U = 0.7941$ . When  $1 - P_s(\tilde{\mathbf{q}}, t)$  is above  $\pi_U$ , the vehicle decides that there is actually no object.

## VII. VEHICLE MOTION CONTROL

In this section, we summarize the main results of the search control strategy presented in [11]. We will consider a motion control strategy for the vehicle that guarantees finding all objects in  $\mathcal{D}$  (i.e., achieve  $\mathcal{J} \rightarrow 0$ ) with the minimum Bayes risk at every time step.

Let the control  $\mathbf{u}(t)$  be restricted to a set  $\mathcal{U}$ . Based on this constraint on the control, we define the set of points in  $\mathcal{W}$  reachable from the current location of the vehicle at time  $t$  as  $\mathcal{Q}_{\mathcal{W}}(t) = \{\tilde{\mathbf{q}} \in \mathcal{W} : \tilde{\mathbf{q}} - \mathbf{q}(t) \in \mathcal{U}\}$ . We use a control law that drives the vehicle to some point  $\tilde{\mathbf{q}} \in \mathcal{Q}_{\mathcal{W}}(t)$  that has the highest uncertainty, and switch to a perturbation control law when the vehicle is trapped in a region where no such point exists. Let us first consider the following condition, whose utility will become obvious shortly.

**Condition C1.**  $H_s(P_{H_s}, \tilde{\mathbf{q}}, t) \leq \epsilon$ ,  $\forall \tilde{\mathbf{q}} \in \mathcal{Q}_{\mathcal{W}}(t)$ , where  $\epsilon$  is a preset threshold of some small value.

Consider the following control law

$$\mathbf{u}^*(t) = \begin{cases} \bar{\mathbf{u}}(t) & \text{if C1 does not hold} \\ \bar{\bar{\mathbf{u}}}(t) & \text{if C1 holds} \end{cases} \quad (12)$$

where  $\bar{\mathbf{u}}(t)$  is the *nominal control law*, and  $\bar{\bar{\mathbf{u}}}(t)$  is the *perturbation control law*.

The nominal control law is set to be  $\bar{\mathbf{u}}(t) = \tilde{\mathbf{q}}_*(t + 1) - \mathbf{q}(t) \in \mathcal{U}$ , where  $\tilde{\mathbf{q}}_*$  is the point that has the highest uncertainty within  $\mathcal{Q}_{\mathcal{W}}(t)$ . This choice for the nominal control law is inspired by the nominal control law in [12]. If Condition C1 holds, then the perturbation controller  $\bar{\bar{\mathbf{u}}}(t)$  is used:  $\bar{\bar{\mathbf{u}}}(t) = -\bar{k}(\mathbf{q}(t) - \bar{\mathbf{q}}^*)$ , where  $0 < \bar{k} \leq 1$  is the controller gain, and  $\bar{\mathbf{q}}^* \in \mathcal{Q}_{\mathcal{D}}(t) := \{\tilde{\mathbf{q}} \in \mathcal{D} : \tilde{\mathbf{q}} - \mathbf{q}(t) \in \mathcal{U}\}$  such that  $H_s(P_{H_s}, \bar{\mathbf{q}}^*, t) > \epsilon$ . We assume that  $\mathcal{U}$  is such that  $\mathcal{Q}_{\mathcal{D}}(t) = \mathcal{D}$  for all time  $t$ . The controller is used to drive the vehicle out of the region with low uncertainty  $\epsilon$  to some  $\bar{\mathbf{q}}^* \in \mathcal{Q}_{\mathcal{D}}(t)$  such that  $H_s(P_{H_s}, \bar{\mathbf{q}}^*, t) > \epsilon$ , if such a point exists.

## VIII. SIMULATION

In this simulation, we consider all the points  $\tilde{\mathbf{q}}$  within a  $20 \times 20$  square domain  $\mathcal{D}$ . For each  $\tilde{\mathbf{q}} \in \mathcal{D}$ , we assume an i.i.d. prior probability of object presence equals to

$P_s(\bar{\mathbf{q}}, 0) = 0.2$ . The number and locations of the objects are randomly generated. The number of objects generated for this simulation turns out to be 70 with locations as indicated by the magenta dots in Figure 2. The radius  $r_s$  of the search sensor is chosen to be 8 and the classification radius  $r_c$  is chosen to be 6, as shown by the magenta and green circle in Figure 2. The black dot represents the position of the vehicle. Figure 2 shows the evolution of  $H_s$ . From Figure 2(d), we can conclude that at most  $H_s = 1.1 \times 10^{-8}$  has been achieved everywhere within  $\mathcal{D}$ . In this case, we set the maximum sensing capacity as  $M = 0.5$ . The parameter  $\gamma = 0.05$ . For the classification process, let the desired upper bound for classification uncertainty be  $\epsilon_c = 0.01$ . The priors  $P_c(\bar{\mathbf{q}}, 0) = 0.5$  and all the objects with even number have property ‘‘F’’. Here we use the control law in equation (12) with control gain  $\bar{k} = 0.2$ . The set  $\mathcal{U}$  is chosen to be  $\mathcal{D}$ .

Figure 3(a) shows the number of false and missed detections versus time. The number of missed detections (22) is much larger than that of false detections (2) at  $t = 0$ . This is because the initial prior probability  $P_s(\bar{\mathbf{q}}, 0)$  we start with is closer to zero, which makes it easier to have a wrong decision with one negative observation given that the actual state is object present. The simulation results also suggest that the number of incorrect classifications for the two properties are similar and this is because we have  $P_c(\bar{\mathbf{q}}, 0) = 0.5$ . As time increases, both error numbers decrease to zero with zero uncertainty at the end of the task. This implies that we can balance the number of errors within the tolerance range and the limited time we have to decide when to stop.

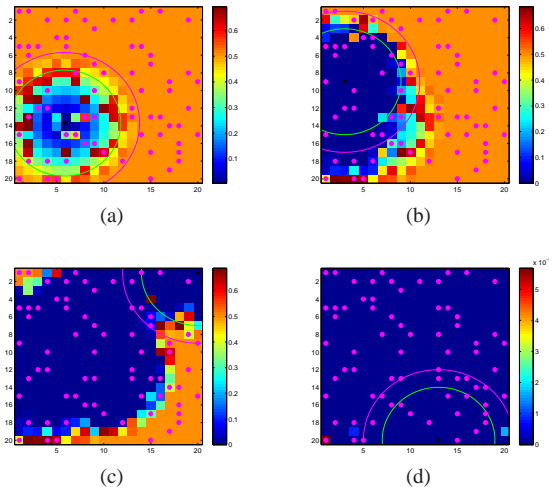


Fig. 2. Uncertainty map (dark red for highest uncertainty and dark blue for lowest uncertainty) at (a)  $t = 1$ , (b)  $t = 120$ , (c)  $t = 280$ , and (d)  $t = 360$  (with initial uncertainty  $H_s(P_s(\bar{\mathbf{q}}, 0))$ ).

Figure 3(b) shows the property of object 3, which eventually has zero probability of having Property ‘‘F’’ with zero uncertainty, i.e., we are 100% sure that object 3 has Property ‘‘G’’. Similar high certainty results were obtained for all other objects.

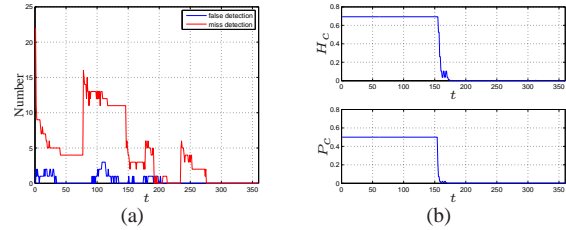


Fig. 3. (a) Number of false and missed detections, and (b) Probability of object 3 having Property ‘‘F’’ and the corresponding uncertainty function  $H_c$ .

## IX. CONCLUSION

Based on a Bayesian probabilistic framework, a decision-making strategy was developed to guarantee the detection and classification of all objects in a domain using Bayesian risk analysis. Future research will focus on locating and classifying dynamic objects with multiple autonomous sensor vehicles. The question of unknown environment geometries (i.e., unknown  $\mathcal{D}$ ) will also be addressed. Objects with uniform distributions over the domain will be investigated, where the decision-making at one point is affected by all the decisions made at other points. SPRT method will also be investigated for the cases where no prior information is available.

## REFERENCES

- [1] J. R. Spletzer and C. J. Taylor, ‘‘Dynamic Sensor Planning and Control for Optimally Tracking Targets,’’ *The International Journal of Robotics Research*, no. 1, pp. 7–20, January 2003.
- [2] L. F. Bertuccelli and J. P. How, ‘‘Robust UAV Search for Environments with Imprecise Probability Maps,’’ *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference*, December 2005.
- [3] —, ‘‘Bayesian Forecasting in Multi-vehicle Search Operations,’’ *AIAA Guidance, Navigation, and Control Conference and Exhibit*, August 2006.
- [4] J. J. Leonard and H. F. Durrant-Whyte, ‘‘Simultaneous Map Building and Localization for an Autonomous Mobile Robot,’’ in *IEEE/RSJ International Workshop on Intelligent Robots and Systems IROS ’91*, Osaka, Japan, November 1991, pp. 1442–1447.
- [5] T. Furukawa, F. Bourgault, B. Lavis, and H. F. Durrant-Whyte, ‘‘Recursive Bayesian Search-and-Tracking using Coordinated UAVs for Lost Targets,’’ *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, May 2006.
- [6] B. Lavis, T. Furukawa, and H. F. Durrant-Whyte, ‘‘Dynamic Space Reconfiguration for Bayesian Search-and-Tracking with Moving Targets,’’ *Autonomous Robots*, vol. 24, pp. 387–399, May 2008.
- [7] R. Mahler, ‘‘Objective Functions for Bayesian Control-Theoretic Sensor Management, I: Multitarget First-Moment Approximation,’’ *Proceedings of IEEE Aerospace Conference*, 2003.
- [8] M. Flint, M. Polycarpou, and E. Fernandez-Gaucherand, ‘‘Cooperative Control for Multiple Autonomous UAV’s Searching for Targets,’’ *Proceedings of the 41st IEEE Conference on Decision and Control*, December 2002.
- [9] H. V. Poor, *An Introduction to Signal Detection and Estimation*, 2nd ed. Springer-Verlag, 1994.
- [10] A. Wald, *Sequential Analysis*. Dover Publications, 2004.
- [11] Y. Wang and I. I. Hussein, ‘‘Bayesian-Based Decision Making for Object Search and Characterization,’’ *IEEE American Control Conference*, 2009.
- [12] I. I. Hussein, ‘‘A Kalman-filter based control strategy for dynamic coverage control,’’ *Proceedings of the American Control Conference*, pp. 3271–3276, 2007.