[temp] optimal searching

The teeeeeeam

February 7, 2015

Contents

Our model is partitioned into two separate searches. First, a visual aerial search is performed for floating debris, and locations of debris are indexed. A transformation is applied to convert positional data of debris into a probable area of the plane crash. An existing model for the oceanic currents of the region is presumed to already to exist. This model is used to predict the locations of crash debris as a probability distribution. This positional prediction data is combined with data from the ocean current model in order to create a more localized search area of the wreckage on the sea floor. A second underwater search, informed by the aerial search, is then conducted via underwater vehicles utilizing sonar.

1 General Discrete-effort Optimal Searching

The discrete-effort method of Lawrence Stone's *Theory of Optimal Search* is used in both searches to prioritize and distribute search efforts. An overview of this method is provided here.

1.1 Defining the Search Method

The search space is divided into an integer number of "cell" segments $\mathcal{J}$. Each cell j has a $\gamma_t(j,k)$, being the cost of the k^{th} search in that cell using technology $t \in \mathcal{T}$. The efficiency of technologies used in the search are indexed for convenience in a set $\mathcal{T}$. This generalization of technology can be implemented for accurate measurements in subsequent uses.

We denote the probability of debris existing in a given cell as $p(j)$. Assuming there truly is debris in cell j, the probability of locating an object there using technology t on the k^{th} search is referred to as $\beta_t(j,k)$.

The quantity $p(j)\beta(j,k)$ represents the actual chance of finding debris in the jth cell on the k^{th} search. β assumes the object is in the current cell being searched, where the product $\beta * p$ does not assume so.

Next, the optimization problem is stated. The optimal individual cell to focus the search methods on is the one which provides the highest chance of success per unit search cost. We denote this value as ϵ. This value is defined as

$$\epsilon_t \equiv \epsilon_t(j,k) = \frac{p(j)\beta_t(j,k)}{\gamma_t(j,k)}$$

.

An ϵ can be computed for each cell, accounting for the number of searches k for each one. The highest values are the ones with the greatest probability of success per unit cost.

1.2 Assigning Values to $\beta_t(j,k)$

Suppose α_t is the success rate for locating surface wreckage, with technology and search aircraft as parameters. Then, from *Theory of Optimal Search*:

$$\beta_t(j,k) = \alpha_t \cdot (1-\alpha_t)^{k-1}$$

This representation follows that each successive search has an exponentially decreasing chance of success. The derivation is not copied here. It is worth noting, however, that the α_t value will significantly depend on the technology used and the context within the model is applied.

1.3 Cost Functions as Prioritization

The cost $\gamma(j,k)$ provides flexibility for the model in terms of prioritization. Suppose the cost per search is a constant value that is not one; then, a change of units can be easily applied in order to achieve $\gamma(j,k) = 1$. In this case, the search regime simply optimizes for the highest chance of success.

However, this approach is naive; considerations such as distance, time, and money may confound the cost of searching. Should these factors be disregarded, a $\gamma(j,k) = 1$ will suffice. The resulting function γ will depend entirely on the technology being used.

For instance, one may include a distance parameter in the cost of the function, such that $\gamma(j,k) = 1 + d_{travel}$, where d_{travel} is the distance from the current position to the position under consideration. This has the benefit of moving ships around as little as possible. One may also wish to make cost a function of time to search - in other words, $\gamma(j,k) = t_{search} + d_{travel}$. These ideas will be explored further in the later discussion on accounting for assumptions.

1.4 Iterating Failures

In the event that the model does not find an object in cell k, the probability distribution must be modified to match. This modification, discussed in Bayesian search theory, occurs as follows. Let T represent the probability that the plane is in the cell, and F the probability that it is found. Then:

$$P(T|F) = \frac{P(T)P(F|T)}{P(F)}$$

$$P(T|\sim F) = \frac{p(k)[1-\beta_t(j,k)]}{(p(k)[1-\beta_t(k,j)] + (1-p(k))}$$

Thus, the posterior probability that it is there but was not found is:

$$p'(k) = p(k)\frac{1-\beta_t(j,k)}{1-\beta_t(k,j)p(k)}$$

.

The posterior probability of every other square can be similarly adjusted (though its proof is omitted):

$$p'(k) = p(k)\frac{1}{1 - p(k)\beta_t(j,k)}$$

(The notation here may be confusing. For clarification, the first $p'(k)$ adjustment applies only to the square that was searched; the second applies to every other square.)

After a failed search, the probability of the plane being in cell k becomes this probability.

1.5 Executing the Search Method

Suppose there is a set $\mathcal{T}$ of device types and n_t devices of type $t \in \mathcal{T}$. For each device, the ϵ value is computed. This represents the probability of discovering the plane in comparison to the cost of its search.

Each device is allocated sequentially to the square that is a) not already being searched, and b) has the highest ϵ value. Should two devices conflict over the optimal square, the closest device of that type claims it. Though travel time is not modeled, it is reasonable to make preliminary decisions in deference to it.

Whenever a device finishes searching, the p-distribution is recomputed, the device's ϵ value is recomputed based on the new distribution, and it is sent out again.

In the case where the time allowed to search is not limited, this is optimal for detection. However, this is not the ideal when the time for searching is limited, as would be the case in an aerial flow model. In future applications, it would be possible to extend the aerial search model to allow for best-guessing after a constrained amount of time has expired.

2 Current Flow Modeling

Existing current models are sufficient for determining approximately where a piece of debris could have come from. These current models are not included in this paper, but their outputs important, and are declared as follows:

A current model is assumed to generate two probability distributions: the first is the distribution of debris, and the second is the positional distribution of the plane. The debris distribution is $p_d(j)$ (for $j \in \mathcal{J}$). The plane's positional distribution is considered naive at the outset, as it is not yet informed by the position of debris. The naive positional distribution is $p_n(j)$.

Additionally, it is expected that the current model be able to adjust the positional probability of the plane based on the originating point of the debris. This is expressed with the operator $\mathtt{T}$. This will be defined later.

3 Aerial Debris Search

The first step in this model for object search is to perform an aerial debris search.

Planes will search through cells in a manner specified by the general optimal search model. The highest discovery rate to cost ratio will be investigated first, then each other cell sequentially.

The presence of debris at cell j is $d_j \in \{0, 1\}$. It is assumed that there either is or is not debris in a cell. If debris is present, then $\mathtt{T}$ modifies the probability distribution p_n in order to represent the likelihood that debris came from a plane crash. This is expressed as:

$$\mathtt{T}(d_j, p_n) = p'_n$$

Repeated application of $\mathtt{T}$ over each cell d_j will create the new positional distribution informed by debris, p_w. This is created by:

$$p_w = \mathtt{T}(d_1, \mathtt{T}(d_2, \ldots \mathtt{T}(d_j, p_n) \ldots))$$