NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

OPTIMAL SEMI-ADAPTIVE SEARCH WITH FALSE TARGETS

by

John P. McCray

December 2017

Thesis Advisor:	Johannes O. Royset
Second Reader:	Dashi I. Singham

Approved for public release. Distribution is unlimited.

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 0704–0188*

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202–4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.

1. AGENCY USE ONLY *(Leave Blank)*	2. REPORT DATE December 2017	3. REPORT TYPE AND DATES COVERED Master's Thesis 01-01-2016 to 12-15-2017

4. TITLE AND SUBTITLE

OPTIMAL SEMI-ADAPTIVE SEARCH WITH FALSE TARGETS

5. FUNDING NUMBERS

6. AUTHOR(S)

John P. McCray

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Postgraduate School
Monterey, CA 93943

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

N/A

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

The views expressed in this document are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol Number: N/A.

12a. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release. Distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(maximum 200 words)*

Searchers frequently encounter the presence of false targets or clutter, which appears indistinguishable from the real target and must be identified in a second stage of the search. False targets can significantly impede search operations, such as underwater recovery and mine warfare, when contact investigation is costly. Current literature optimizes these searches by applying Bayesian updates to the prior distributions for the real and false targets, in what is called a "semi-adaptive" search. We take full advantage of intermediate search results, along with soft information about the target, to build up-to-date maximum likelihood estimates of the location of the real target and the distribution of the clutter. Using these estimates in place of the priors, we update and improve the allocation of search effort as the operation progresses. In a detailed simulation study, this new approach increases the probability of finding the target by up to 12% over the optimal semi-adaptive plan without such estimates. These gains are robust to variation in the false target density, time to identify false targets, and total search time available.

14. SUBJECT TERMS

optimal search, semi-adaptive, false target, probability estimation

15. NUMBER OF PAGES 115

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UU

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release. Distribution is unlimited.

OPTIMAL SEMI-ADAPTIVE SEARCH WITH FALSE TARGETS

John P. McCray
Lieutenant Commander, United States Navy
A.B., Harvard University, 2000

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
December 2017

Approved by: Johannes O. Royset, Ph.D.
Thesis Advisor

Dashi I. Singham, Ph.D.
Second Reader

Patricia A. Jacobs, Ph.D.
Chair, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Searchers frequently encounter the presence of false targets or clutter, which appears indistinguishable from the real target and must be identified in a second stage of the search. False targets can significantly impede search operations, such as underwater recovery and mine warfare, when contact investigation is costly. Current literature optimizes these searches by applying Bayesian updates to the prior distributions for the real and false targets, in what is called a “semi-adaptive” search. We take full advantage of intermediate search results, along with soft information about the target, to build up-to-date maximum likelihood estimates of the location of the real target and the distribution of the clutter. Using these estimates in place of the priors, we update and improve the allocation of search effort as the operation progresses. In a detailed simulation study, this new approach increases the probability of finding the target by up to 12% over the optimal semi-adaptive plan without such estimates. These gains are robust to variation in the false target density, time to identify false targets, and total search time available.

THIS PAGE INTENTIONALLY LEFT BLANK

Table of Contents

List of Figures

List of Tables

THIS PAGE INTENTIONALLY LEFT BLANK

List of Acronyms and Abbreviations

ABC	Approximate Bootstrap Confidence
BCA	Bias-Corrected and Accelerated
CDF	Cumulative Distribution Function
CI	Confidence Interval
COI	Contact of Interest
CSR	Completely Spatially Random
GAMS	General Algebraic Modeling System
GPS	Global Positioning System
HPP	Homogeneous Poisson Process
ICCI	Immediate and Conclusive Contact Identification
ICCI-E	Immediate and Conclusive Contact Identification with Estimates
ID	identification
MLE	Maximum Likelihood Estimate
NHPP	Non-Homogeneous Poisson Process
OAP	Optimal Adaptive Plan
ONAP	Optimal Non-Adaptive Plan
ONAP-E	Optimal Non-Adaptive Plan with Estimates
OSAP	Optimal Semi-Adaptive Plan
OSAP-AE	Optimal Semi-Adaptive Plan with Accepted Estimates

OSAP-E	Optimal Semi-Adaptive Plan with Estimates
PDF	Probability Density Function
ROV	Remotely Operated Vehicle
USV	Unmanned Surface Vehicle
UUV	Unmanned Underwater Vehicle

Executive Summary

Many searches for stationary targets are complicated by the presence of other objects that appear similar to the target during broad-area search. Frequently, a broad-area sensor will be unable to discriminate between these false targets and the object of the search, requiring a two-stage process of detection followed by identification of each contact of interest. As contact investigation can be quite costly, and false targets numerous, it can add significantly to the difficulty and time requirements of a search operation. In the context of an underwater search, these false targets may be rocks, sea trash, or other debris that a side-scan sonar might detect.

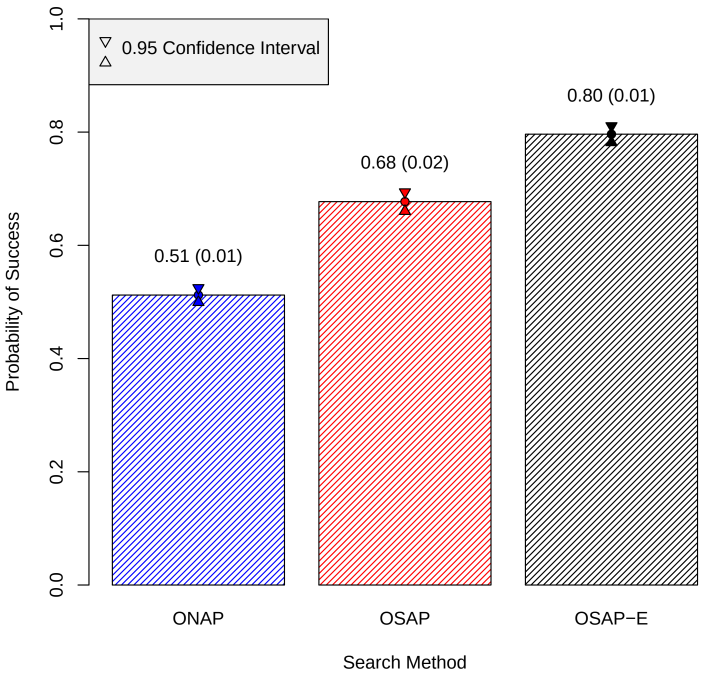
Published methods to address this complication optimize searches with false targets by balancing the application of broad area search with contact identification over a set amount of broad search time [1], [2]. These methods can learn and adapt from the unsuccessful application of search effort to better balance broad area search with contact investigation as the search operation progresses [3]. This is called semi-adaptive search – it recalculates the optimal allocation of search effort at regular intervals, taking advantage of the search results up to that point. Semi-adaptive search performs at least as well as non-adaptive search that does not use such updates, but not necessarily as well as adaptive search that can anticipate and adapt to information as it is received [2]. However, semi-adaptive search relies on prior distributions for the real target and false targets that remain fixed over the course of the search, allocating effort based on failure to locate and identify the target.

In contrast to relying on fixed priors, we propose to use interim search results to build new estimates of the probability density for the location of the target and the distribution of false targets over the search region. These estimates require additional prior information about the spatial distribution of false targets, and how they are related to the real target. In this study, we assume the target is located in an elliptical region of higher than average intensity of false targets. We use data-derived estimates of the contact intensity in a modified version of the existing false-target search construct [1] to optimize the allocation of future search effort.

We implement these methods numerically to build probability density estimates and optimal

allocations of search effort over a continuous, two-dimensional region representing a search in progress. This implementation also gives the probability of success over various durations of search operations, to support decisions prior to and during a search in progress.

We use simulation to study and compare the effectiveness of these search methods, modeling searches for a target within search regions that are randomly populated with false targets. Our statistic of interest is the fraction of successful searches over a number of randomly generated regions; we compare the effectiveness of all methods by simulating searches multiple times. Semi-adaptive searches using estimates of target probability density and false target intensity have significantly greater probability of success compared to searches that do not use such estimates, provided assumptions are at least weakly met. Under one set of conditions in the presence of 15-20 false targets, use of estimates improves the optimal semi-adaptive search success rate from 68% to 80% of searches, as shown in Figure 1.



Summary of results of over 10000 simulated searches in randomly generated search regions. Optimal semi-adaptive search with estimates (OSAP-E) outperforms traditional semi-adaptive search (OSAP) and non-adaptive search (ONAP).

Figure 1. Probability of Success of Search Plans

Improvements in the success rate are significant when the prior accurately reflects the actual process that generated the false targets. Further study can balance how to blend the prior information with estimates from incomplete data when the prior is a poor approximation. Implementation under real-world constraints on employment of physical search assets will involve discrete approximations to the continuous solutions given by these techniques. These improvements will support operationalization with a field-deployable decision support system to make recommendations on search allocation for search operations in progress. Applications include two-stage searches requiring detection and identification, where there is a spatial relationship between the actual target and a subset of the false targets. Examples are high-profile search events such as searches for missing airliners, and mine warfare.

References

[1] L. D. Stone, J. A. Stanshine, and C. A. Persinger, "Optimal search in the presence of Poisson-distributed false targets," *SIAM Journal on Applied Mathematics*, vol. 23, no. 1, pp. 6–27, July 1972.

[2] L. D. Stone, *Theory of Optimal Search*, 1st ed. New York, NY: Academic Press, 1975.

[3] L. D. Stone, "Semi-adaptive search plans," Daniel H. Wagner, Associates, Tech. Rep. AD785295, Sep. 1973.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 1:
Introduction and Background

1.1 Background

The study of how to better conduct search operations is a problem with a long history dating to the birth of operations research in World War Two. Its history continues with Cold-War era searches for missing submarines and lost nuclear weapons at sea, later searches for shipwrecks and lost treasures, and several more recent searches for lost airliners [1]–[4].

Real-world search operations often take place in "noisy" environments that contain more than just the target. Many undersea searches take place in two stages. In the first stage, searchers conduct a broad area search using a side-scan sonar to detect low-resolution contacts that may or may not be the search target. Sonars from ships and Unmanned Underwater Vehicles (UUV) detect reflected acoustic energy, ideally from the desired target but also from other underwater objects such as rock formations or uncharted shipwrecks. These contacts are merely blips on the sonar without enough information to determine what kind of object generated the return. In the second stage, searchers investigate contacts detected during the broad area search with a different type of capability, such as a Remotely Operated Vehicle (ROV) equipped with searchlights and a camera, to investigate whether each contact is, in fact, the target of the search. If it is not, it was a false target. False targets may be sea trash, rocks, or other unrelated wreckage. The search problem is made considerably more difficult by the significant effort required for the ROV to investigate each contact.

Optimal search theory is concerned with the best way, in some sense, to find a desired target during a search over a specified region. We consider an optimal search to be the search plan that provides the greatest probability of success, that is, the highest probability of positively identifying the target or object being sought within a specified time period. This research explores a new approach for how to find the best way to search for stationary targets in the presence of false targets, or clutter.

There are many possible applications of this type of search. We use an airliner lost at sea, such as Malaysian Air Flight 370, as a motivating example. A missing airliner could have belly-landed on the water and be sitting intact at the bottom of the ocean in a way that is easily identified even by broad-area search, side-scan sonar. Alternatively, it could have exploded mid-air into pieces no larger than a jet engine, creating a large debris field that may be difficult to distinguish from the background clutter present everywhere on the sea floor. Other contexts include hunting mines in the presence of underwater clutter. The broader approach detailed in this research – of learning from false targets as they are observed, to improve future allocations of search effort – may also be applicable in different kinds of searches, such as finding smugglers intermixed with innocuous maritime traffic.

1.2 Scope

This study develops a method for conducting searches to locate and identify stationary targets in the presence of clutter, or false targets. False targets are initial broad-area search results that appear indistinguishable from the actual target, but are not actually the target. It may require significant additional effort to discriminate a false target from a real target; this can greatly increase the amount of time and effort required to successfully prosecute a search. The searcher must decide how to allocate limited search resources between broad area search, and investigation and classification of previously located contacts. We assume that our broad-area sensor cannot distinguish between real and false targets - even in gross measures such as size. Thus every Contact of Interest (COI) is equivalent. This is a conservative assumption that also simplifies our ability to calculate and simulate search plans.

We study the continuous case where any continuous application of broad area and contact investigation effort over the search region is feasible. This is not necessarily realistic; search plans in practice must account for the reality of the assets in use such as ships and UUVs. Optimal continuous search allocations tend to spread search effort thinly over a large area. This is either inefficient or impossible with actual search assets, which follow paths in the real world, and cannot simply spread incremental search effort over a broad area as our model assumes. We optimize in the continuous case, to be able then to make the best possible discrete approximations if needed, given the constraints of a particular search. For further discussion of this issue and recommendations, see [5] and Chapter 7 of [6].

We do not consider all sources of uncertainty. The searcher does not know how clearly targets will appear to sensors. Sensor performance can be affected by any number of environmental factors, such as water conditions and sea bottom type, as well as the age and condition of the targets themselves, so the detection rates and probabilities are also uncertain. We do not attempt to model or predict sensor performance; instead we assume that the search functions – which relate the search effort expended to the probability of detecting, or identifying, targets – are well known. This allows us to focus on the uncertainty in the location of the real target and the distribution of false targets.

To do so, we need to have sufficient soft information to make several assumptions about the search, the probability distribution of the target, the distribution of false targets, and how they are related. In our motivating search for a missing airplane, for example, the searcher's objective is to recover the cockpit flight recorder or "black box". In this case, we can imagine a large number of false targets of two types: the true sea-bottom clutter, such as rocks, reefs, and trash; but also the other pieces of the crashed jetliner itself. Searchers want to find the section of the fuselage containing the black box, not to spend time on the tail, wings, engines or other assemblies that may have separated when the plane impacted the water. Even though these are pieces of the wreckage, for the searcher who needs the black box, they are these false targets. Therefore, we consider a search with both *unrelated* and *related* false targets.

We expect, over the given search region, that the intensity of unrelated false targets is uniform. We model this with a homogeneous Poisson process. Further, we assume both the debris field of related false targets, and the black box, are to be found around the central location of the crash site. Thus we expect that the probability density of the target is proportional to the intensity of the related false targets, such that it scales to a total probability of 1.0. This method is appropriate in the general case for different models of the target intensity. Here, we expect the prior densities for the debris to have a roughly elliptical pattern, based on position errors in the last known location prior to the crash.

1.2.1 Study Objectives

In this study, our first goal is to optimize a search in progress to maximize the likelihood of finding the target over a finite series of discrete sorties, using all information available.

We want to extract as much information as possible from completed sorties, by fusing false target detections with soft information and context to improve our probability estimates for the target location and for the false contact density. We use these updated estimates to decide how best to continue prosecuting the search, using the capabilities and time available. Given a set of initial conditions and context, we demonstrate the ability to estimate the probability for success of a proposed search. This will aid in planning search operations before they begin. An alternate measure of effectiveness is the smallest mean time to positive detection; we do not consider this directly, but provide the means for planners to estimate time requirements. Finally, we build simulation tools to study effectiveness of several methods for allocating search effort, in terms of mean probability of success for a search of a given length. We explore the effects of varying factors such as sensor capabilities, time available, and false target density. This provides an understanding of what factors have the greatest impact on search success, and what we can do to improve outcomes under various sets of conditions.

1.3 Literature Review

Optimal search theory for stationary targets was first developed by Bernard Koopman and others to support the Navy in World War Two; see [7]. The standard work in the field is [6]. The most recently published monograph is [8]; we adapt its presentation of the problem in this section and include it in Appendix B.

1.3.1 Optimal Search

The algorithm for optimal search allocation in the case of continuous space, continuous allocation effort uses as inputs the search region R, a detection function giving the capability of the sensor used, the cost (or time) to apply search effort, a prior distribution for the location of the target, and the total cost, in terms of effort or time, available to complete the search K. It then calculates a Lagrangian multiplier, which represents the optimal “rate of return” of search effort in terms of probability of success. Using this Lagrangian multiplier, the algorithm can then return the optimal allocation of search effort over the given region. It is possible to find the optimal value of the Lagrangian multiplier by a one-dimensional linear search, although this involves a number of inverse function evaluations and an integral over R, and is not trivial to calculate. The search plan returned is a function $f(\boldsymbol{x})$ over space R

returning the optimal value of effort to apply. The integral $\int\limits_S f(\mathbf{x})\, \mathrm{d}x$ then sums to K. Note that this formulation assumes an input probability distribution function for the target that does not change over time.

See Figure 1.1 for an example of an optimal search allocation given a bivariate normal probability distribution for the target. The optimal search proceeds by steadily expanding the area of search, in order to search the area of maximum probability first.

1.3.2 False Targets

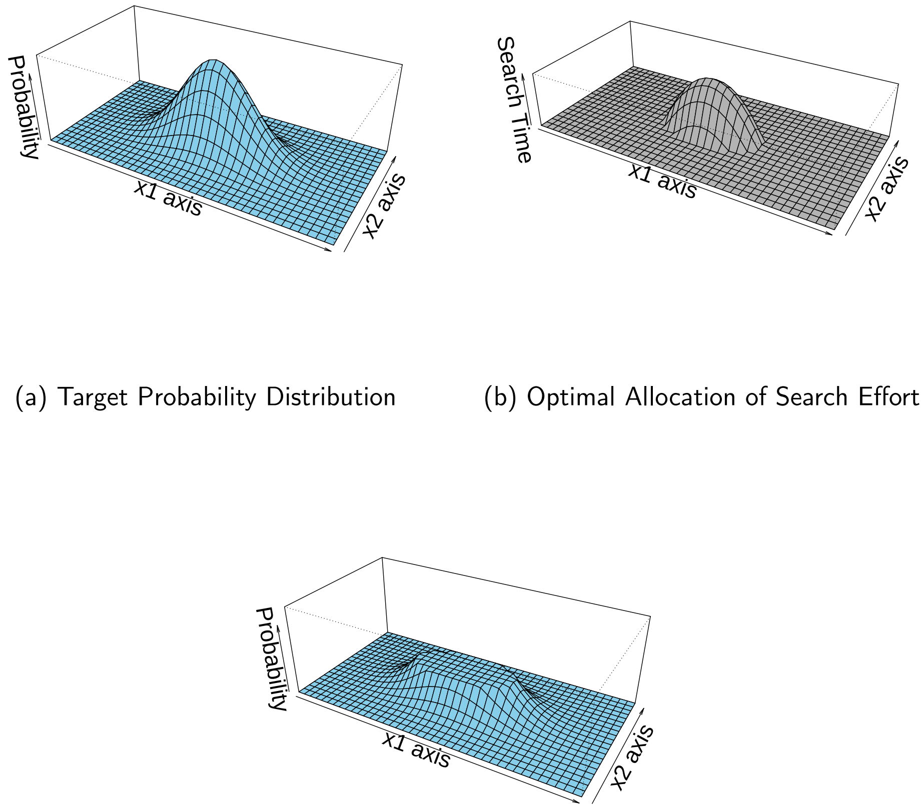
In the presence of false targets, a search plan consists of both an allocation of broad area search effort, and a contact investigation policy to deal with any contact detected in the broad search.

There is no recently published work with an equivalent method for optimizing searches with false targets. In Chapter 4 we summarize the methodology for optimal false target search provided in [6], [9], [10] and outline the modifications we have made for this study. We use the formulations from [9], [10] extensively, using them as a starting point for our model.

Some researchers study contacts that may be ghost returns that are not real objects, but rather spurious returns or noise generated by the sensor. We do not consider these types of false targets here; see [11], [12].

Kalbaugh discusses the immediate and final classification of contacts as either targets or false targets [13]. Kalbaugh considers a searcher that is able to discriminate a false from a real target with a certain probability, without the requirement for a two-stage search of detection and classification. The structure of the problem reflects possible application to a missile seeker that must decide between a target to destroy, and a decoy to bypass in search of a more promising target.

Conolly and Pierce write on the intersection between information theory and search [14]. In a search, information theory proposes that the best allocation of search resources is that which makes the most use of available information, and most rapidly learns information about the region. They study one and two-cell systems and assume there is no cost of contact investigation.



(a) Target Probability Distribution

(b) Optimal Allocation of Search Effort

(c) Posterior Target Probability Distribution

With a bivariate normal prior target PDF, the optimal search allocation "slices" off the highest probability region, leaving a posterior or leftover region of probability.

Figure 1.1. Example of Optimal Search

Kress, Lin, and Szechtman consider not false targets, but false positives, in the context of locating a hostage whose location must be verified before he or she can be rescued [15]. They consider detections made by a surveillance system that must then be verified in a second search stage (or rescue team), which could be costly. They develop a greedy rule in their Theorem 3.1 to select the discrete search cell with the highest payoff in terms of probability of target detection, to cost in terms of expended effort on false targets. This is similar to the ratio r used in [10] and as Equation (1) in [16] to maximize the multiplier (and dual variable) k, which gives the probability rate of return for continuous search effort applied.

Dobbie, in [17], continues the work of [9], [10] on false target search. He describes some properties of searches and summarizes the difficulties of real-world searches with false targets, discussing false target detections as contingencies.

1.3.3 Adaptive and Semi-adaptive Plans

Stone divides false target search plans into three classes: adaptive, non-adaptive, and semi-adaptive: *Adaptive* search plans are those that use - on a continuous basis - all information generated in the search to optimize the mean time to finding the target [6], [16]. *Non-adaptive* plans do not use information gathered during the search. *Semi-adaptive* plans execute a non-adaptively planned search, update with all available information at a discrete number of time intervals, and then re-embark on a new non-adaptive plan generated using that information until the next update time.

Dobbie solves a simple adaptive search plan [17]. It includes two discrete locations or boxes, and a single false target. Dobbie constructs a Optimal Adaptive Plan (OAP) with several contingencies and cut-off times to optimally allocate broad search and contact investigation effort. No other adaptive search plan is solved in the literature to date.

Stone reports on a simulation study including a solved Optimal Semi-Adaptive Plan (OSAP) [16]. He provides a method for constructing semi-adaptive search plans, which we follow.

1.3.4 Past Findings

Stone et al. in [10] provide the means to optimize a search in the presence of false targets. However, the search plans so built do not have flexibility to adapt. They take as fixed the real target distribution and false target intensity.

Stone et al. in [6], [9], [10] show the optimality of a contact investigation policy that is “immediate and conclusive” when false targets are Poisson distributed and investigations are uninterrupted, but possibly delayed. This means that any contact that is found in the course of broad area search is immediately investigated until it is identified. This further implies that the Optimal Non-Adaptive Plan (ONAP) is the same as the OSAP, under their assumptions. As specified in [10], this results applies when the contact identification for the real target takes an equal or greater time than contact identification (ID) for false targets. If the real target may be identified more quickly and interrupted contact investigation is allowed, then there are diminishing returns to continued contact identification effort. In this case, broad search effort and contact identification effort are split between the two to maximize returns across both. This is what we consider in Chapter 4.

Stone in [6] shows that if μ_a, μ_s, μ_n are the mean times to target identification for the OAP, OSAP, and ONAP, that under a finite sequence of update times,

$$\mu_a \leq \mu_s \leq \mu_n. \tag{1.1}$$

Stone in [16] provides several interesting results from the simulation comparing ONAP and OSAP plans with Immediate and Conclusive Contact Identification (ICCI). It shows that the OSAP can have up to 30% improved mean time to identify the target over the ONAP. This advantage is strongest when there are few false targets, and the cost in terms of search effort of investigating targets is high. Additionally, it shows that a non-uniform PDF for the target – where its probability was clustered in a few cells rather than spread evenly over the search region – tended to reduce the advantage of the OSAP.

1.4 Overview

We extend the work described in Section 1.3 by:

- Updating estimates of the prior real target distribution based on both prior knowledge and information obtained during the search, rather than solely updating real target distribution in a Bayesian manner based on lack of success for the search already conducted
- Considering total search time for planning purposes, not just broad search time
- Modeling divided effort between contact investigation and broad search, not solely ICCI
- Implementing and simulating multiple ONAP and OSAP search plans over continuous 2-dimensional space

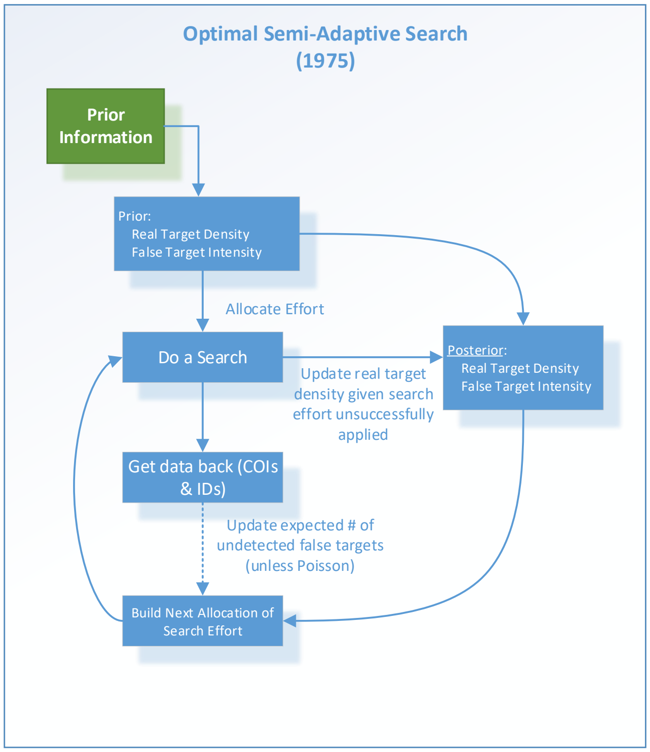
Usually, searches without false targets only need to maintain a distribution function for the target, that is updated as the search progresses. With false targets, we also need to track a distribution for the false targets as well. Current methods update these distributions in a Bayesian manner based on the lack of success implicit in a continued search, as in Figure 1.2. In contrast, we assume prior information is available about the target and the search region. This allows us to build a model for the location of the real target that we can update with data on false targets. We use this model to make improved estimates of the real and false target distributions as we obtain more information from our searches. That is, we do not just update the posterior distributions, we actually update the priors as well using this additional information, as outlined in Figure 1.3 with the additional red lines.

In order to apply the methods of search theory, we first need estimates of the probability distributions. We start in Chapter 2 by developing an approach to updating the probability distributions for the location of real targets, and for the density of false targets. We use broad-area search results and classification results from identification of contacts, but this could potentially include external updates to the probability distributions gained from outside sources.

In Chapter 3, we develop a way to use these estimates of the false target distribution to improve the distributions used in the search optimization methods developed by Stone et al.

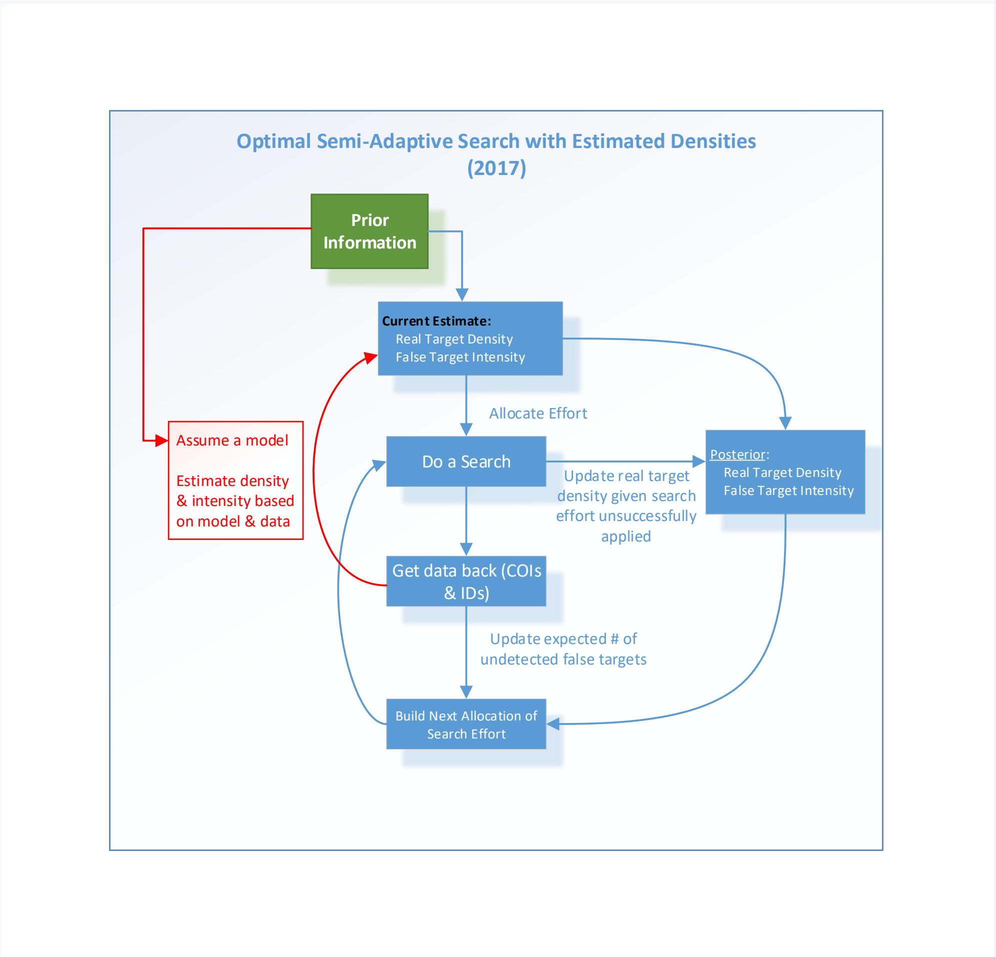
In Chapter 4, we extend the false target search optimization procedures in [10]. We use our estimated probability distributions to find the optimal allocation of effort over the next time increment that maximizes the probability of success in the expected total search time available. This allows us to investigate some initial features of searches with false targets. We then add to the optimization model by relaxing assumptions to make it more realistic and applicable in practice, considering a broader range of detection functions.

We test the model in simulations of search operations in Chapter 5. We conduct simulation experiments to investigate the effect that key input factors have on the overall search results, showing the potential advantage in probability of success by using the estimation methods developed here. This method compares favorably to existing procedures across a range of search conditions.


Optimal Semi-Adaptive Search
(1975)
Prior Information
Prior:
Real Target Density
False Target Intensity
Allocate Effort
Do a Search
Posterior:
Real Target Density
False Target Intensity
Update real target density given search effort unsuccessfully applied
Get data back (COIs & IDs)
Update expected # of undetected false targets (unless Poisson)
Build Next Allocation of Search Effort

Previously published methods for false target search relied on strongly held prior beliefs about the distributions of false and real targets and only updated those probabilities based on the failure of past search efforts implicit in a continued search. Adapted from [6], [16].

Figure 1.2. False Target Search Optimization


Optimal Semi-Adaptive Search with Estimated Densities (2017)
Prior Information
Current Estimate:
Real Target Density
False Target Intensity
Allocate Effort
Assume a model
Estimate density & intensity based on model & data
Do a Search
Update real target density given search effort unsuccessfully applied
Posterior:
Real Target Density
False Target Intensity
Get data back (COIs & IDs)
Update expected # of undetected false targets
Build Next Allocation of Search Effort

In this work we use the data to improve our estimates of where the real and false targets are distributed to improve our decisions allocating search effort. We have prior information about the real and false targets that allows us to assume a model and use data from the search to improve these prior estimates. Differences are highlighted in red.

Figure 1.3. False Target Search Optimization with Estimates

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 2:
False Target Density Estimation

2.1 Background and Overview

At a given time in the middle of a search operation, we will have detected a number of sonar contacts. We can use these contacts as a point pattern to estimate the source distribution for the false targets. We then use that source distribution to better inform our search for the real target, when we optimize our allocation of search effort.

2.2 Model

We use a Non-Homogeneous Poisson Process (NHPP) as the source distribution for false targets in our search region. In a NHPP, points are located over space according to an underlying intensity function. This intensity – which can vary over space – gives the expected number of points (here, targets) to be found in a given area.

A NHPP is flexible enough to be used to model the source distribution of false targets for a variety of search regions. An elliptical feature may describe the likely debris pattern for a plane crash, but other types of models are possible. For instance, a conic pattern could spread from a known point along an approximately known azimuth; or a more linear pattern of returns could be present if we were looking for a linear feature; or boxes and even grid patterns from regularly spaced mines in a field. We consider our model to be the form of the intensity function $\lambda(\boldsymbol{x})$, how it varies over space, and its parameters.

Note that many other models – for example, a kernel density estimate – can be used to estimate the intensity function for a given set of points in R, some providing greater computational efficiency. The use of a parameterized model such as this one allows us to do two things. First, it allows us to specify bounds and conditions on the parameters of the model in accordance with any soft information we have prior to the search. This helps us to use all available information to get the best estimate of the location of the targets. Second, we build the structure of our model in a way that we can extract key information from the estimated parameters that will be useful in locating the real target. By taking the false target intensity in conjunction with additional soft information, we can refine our prior probability distribution for the real target.

In this study, we use an unimodal, elliptical peak of intensity added to a constant term for our model. We refer to the elliptical intensity as the "feature," and the constant level the "background." We consider the background intensity to represent the ocean-floor debris, rocks and other hard objects that are present and spread randomly anywhere in the ocean. We consider the elliptical feature to be debris from the wreckage of the plane, that is concentrated in a single location. This debris – part of the feature – helps lead us to the target.

In order to estimate the parameters of our model, we use the method of maximum likelihood estimation. This is a solidly grounded method that works well when computationally tractable, as it is in our case [18].

We make the following assumptions in this chapter:

1. Assume points are false targets generated by a NHPP. False targets are distributed according to the location-dependent intensity $\lambda(x)$ with no interaction between points.
2. Assume the NHPP is the sum of two underlying processes:
   (a) A constant background Homogeneous Poisson Process (HPP)
   (b) A single cluster or 'feature' (NHPP) that we have parameterized
3. Assume this feature has some shape, orientation, and/or other properties, such as the elliptical feature we consider here.
4. There is a defined region of interest R.
5. Since we assume there is only one real target, we treat all contacts as false targets. Once a contact is identified as the real target, the search is over.

2.3 Formulation

The maximum likelihood estimator is simply a set of parameters $\boldsymbol{\theta}$ for our model that, of all possible parameters for our model, maximize the likelihood of having generated the data that we have observed. To find the Maximum Likelihood Estimate (MLE), we first must state the likelihood function, or, in this case, the log-likelihood function. We use the log-likelihood L for a NHPP over continuous space as a function of the intensity λ, from [18] Section 8.2. First we define:

R	Region of space
$\lambda(\boldsymbol{x})$	Estimated intensity of contacts at point $\boldsymbol{x} \in R$
ν	Number of contacts observed
$i \in I$	Set of collected observations or contacts($I = \{1, ..., \nu\}$)
$\boldsymbol{x}^i$	Locations of the i^{th} observation (contact) .

We can then write the log-likelihood as a function of λ,

$$L(\lambda) = \sum_{i=1}^{\nu} \log \lambda(\boldsymbol{x}^i) - \int_R \lambda(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}, \tag{2.1}$$

where $\int_R \lambda(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}$ is the expected number of points (or contacts) from the NHPP with intensity $\lambda(\boldsymbol{x})$ in region R [18].

An alternative formulation would be to maximize entropy. We use maximum likelihood as it is in broader use in spatial statistics, and we are using probabilistic rather than information based methods to optimize our search.

We use a parametric model for λ giving our elliptical feature as follows.

Sets:

$i \in I$: Set of sonar contacts from search results ($I = \{1, ..., \nu\}$)

Data:

$\boldsymbol{x}^i$: Location of sonar contact i observed at point $\boldsymbol{x} \in R$

Variables:

α : Angle or azimuth of major axis of ellipse

β : Exponent giving how steeply the feature drops off to zero

ρ : Background density of false targets (constant)

C : Location of center of feature $C = \begin{pmatrix} C_1 \\ C_2 \end{pmatrix}$

D : Density parameter or height of the cluster

η_1 : Major axis of ellipse: length of feature

η_2 : Minor axis of ellipse: width of feature

$\boldsymbol{\theta}$: Single variable containing key parameters describing the intensity
$\boldsymbol{\theta} = (C_1, C_2, \eta_1, \eta_2, \alpha, \rho, D, \beta)^T$

L : Log-likelihood

$\lambda(\boldsymbol{x})$: intensity of contacts at $\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$, s.t. $E[\text{contacts in } R] = \int_A \lambda(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}$

$\xi(\boldsymbol{x})$: calculated exponent to simplify equations

Formulation:

$$\max_{\alpha,\rho,C,D,\eta_1,\eta_2,\beta} L(\lambda) = \sum_{i=1}^{\nu} \log \lambda(\boldsymbol{x}^i) - \int_A \lambda(\boldsymbol{x}) d\boldsymbol{x} \tag{2.2}$$

s.t.

$$\begin{aligned} \xi(\boldsymbol{x}) = &\left(\frac{(x_1 - C_1)\cos\alpha + (x_2 - C_2)\sin\alpha}{\eta_1} \right)^{2\beta} + \\ &\left(\frac{(x_1 - C_1)\sin\alpha - (x_2 - C_2)\cos\alpha}{\eta_2} \right)^{2\beta} \end{aligned} \tag{2.3}$$

$$\lambda(\boldsymbol{x}) = \rho + De^{-\xi(\boldsymbol{x})} \tag{2.4}$$

$$\rho \geq 0 \tag{2.5}$$

$$(C_1, C_2) \in R \tag{2.6}$$

$$\eta_1, \eta_1 \geq 0 \tag{2.7}$$

$$D \geq 0 \tag{2.8}$$

$$\alpha \in [0, \pi) \tag{2.9}$$

$$\beta \in \mathbf{Z}^+ \tag{2.10}$$

The objective function (2.2) maximizes the log-likelihood that this is the density distribution of points over the region, given the observed points $\boldsymbol{x}^i$. Constraint (2.3) gives the exponent ξ as a function of $\boldsymbol{x}$ for the feature in the search space, with an exponential fall-off with increasing distance from the feature. Constraint (2.4) gives the actual contact intensity as function of $\boldsymbol{x}$, by adding a constant background clutter density ρ to the intensity resulting from the feature at any given point. Constraint (2.5) ensures a positive background process intensity. Constraint (2.6) restricts the coordinates of the center of the feature to the bounds of the search region R. Constraints (2.7) ensure positive dimensions for each feature. Constraint (2.8) ensures the intensity is non-negative everywhere, although we could allow negative features in the general case. Constraint (2.9) sets the bounds on the azimuth of the ellipse, demonstrating how this formulation allows us to use prior soft information to improve our estimates. Finally, constraint (2.10) defines our drop-off exponent as an integer.

The center of each feature is at $(C_1, C_2)^T$. We use the angle α to enable rotation of features. α could represent the direction of travel of an airplane when it crashed.

We use β as a factor in the exponent to modulate the rate of fall-off for the feature's intensity. In general a $\beta > 1$ causes the intensity to fall off more sharply beyond η_1 and η_2.

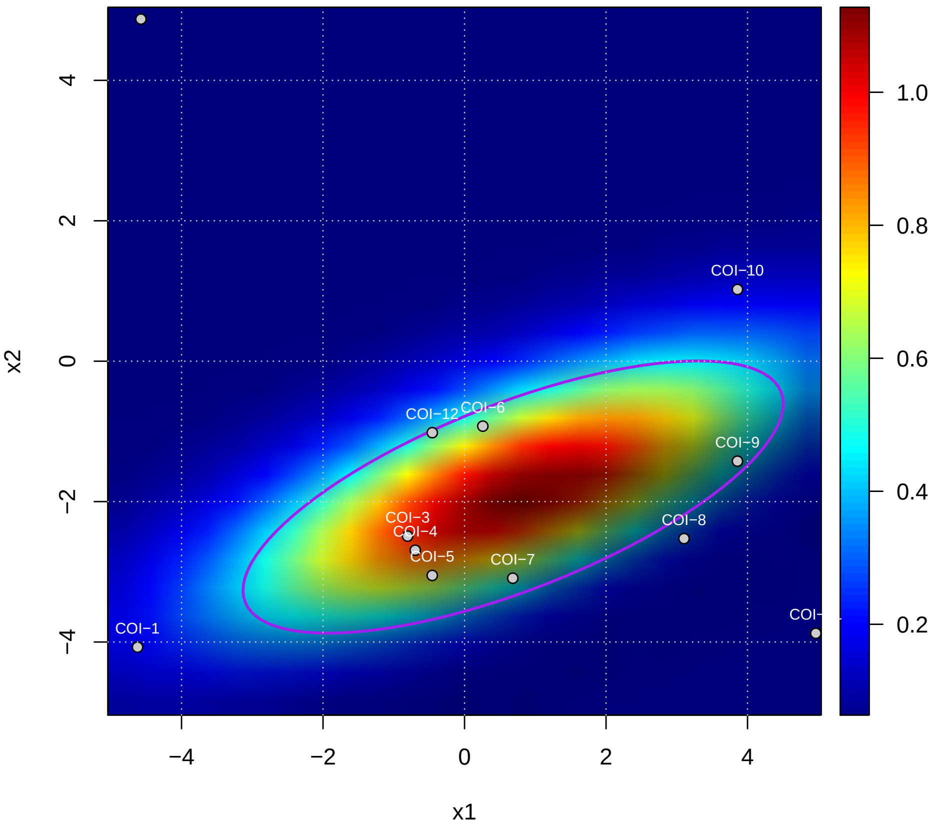
If we have prior information on, for example, flight path, we can specify the azimuth at which we expect the ellipse to lie. We can specify an estimate azimuth $\hat{\alpha}$ and a width of the allowed interval α_w. We use these to create bounds in (2.9) to constrain the angle α to the specified interval $[\alpha - \alpha_w, \alpha + \alpha_w]$.

This is a non-convex problem in our decision variables; the division by the feature size parameters η ensure it is not convex, even if all other variables were fixed. As such, global optimality is difficult to prove; we discuss implications in Section 2.5.

More generally, we could use a similar model for features of a variety of different shapes.

These can support circles, ellipses, and rectangles or lines. We could allow in the general case a number of features $J > 1$ in linear combination over space. In practice J should be small, generally 1 or 2, given the number of parameters required to optimize. We could use a bivariate normal function as well, which has a similar number of degrees of freedom, but its parameters do not support use of prior soft information as naturally.

We demonstrate the ability of this formulation to estimate the intensity model in Figure 2.1.



We can see COIs plotted in R, where we assume a homogeneous application of search effort. The purple ellipse represents the MLE feature with C_1, C_2, η_1, η_2 as in the subtitle, here with background level of clutter $\rho = 0.06$.

Figure 2.1. Example of Contact Intensity Estimate Given Contact of Interest Locations

2.4 Comparison with Homogeneous Process

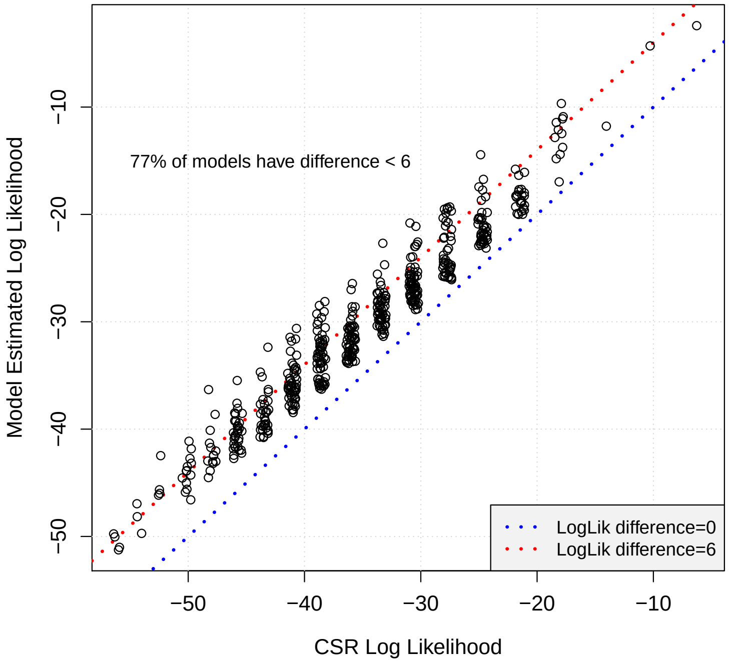
We can compare a non-homogeneous intensity to a homogeneous intensity and see if the MLE feature found by the optimization is backed up by the data. In effect, we are testing the null hypothesis that the intensity is homogeneous, against the alternative hypothesis that there is a feature parameterized as described in Section 2.3.

We can use spatial statistics and functions to calculate a p-value for whether a point pattern is completely spatially random, for example, the J-test or spatial scan test [18]. These require some modification if our application of search effort is not uniform over the region. However, since we are calculating the likelihood based on our model's parameterization, it is straightforward to also calculate the likelihood based on a homogeneous intensity $\lambda(\boldsymbol{x})$ as well, by modifying (2.4) to $\lambda(\boldsymbol{x}) = \rho$ with no spatial dependence. Then we can set up a likelihood ratio test using Wilk's Theorem as a rough approximation:

$$\Delta = 2(L_a - L_0) \sim \chi_n^2, \tag{2.11}$$

where L_a and L_0 are the alternative and null hypothesis log-likelihoods, and n is their difference in degrees of freedom. Our elliptical model estimates seven parameters, to a single parameter in the HPP, so $n = 6$. Thus, a difference in L of 6 gives us a p-value of $\chi_6^2(12) = 0.06$, supporting the alternative hypothesis that the model for λ including a feature is the better fit.

To test overfitting, we can generate spatially random data and evaluate the intensity of the features detected by the model. This model has seven degrees of freedom describing the ellipse and the background noise. Therefore, we expect that it will have a better fit – and higher log-likelihood – for a set of data than a Completely Spatially Random (CSR) model, which only has a single degree of freedom for the constant level of background noise. We can see in Figure 2.2 that the model does return higher estimated log-likelihood values. The L_{est} are close to the L value given by the CSR data, giving us confidence the model is not creating strong, phantom features out of random noise. We see that 77% of models so generated as CSR have $(L_a - L_0) < 6$; this gives our test for CSR an estimated statistical power of 0.77.



Our intensity estimation model will detect a feature in CSR data but it does not depart far from the HPP that generated the data. The log-likelihood of the model L_{est} closely tracks the CSR log-likelihood L_{CSR}, shown by the blue dotted line, over 500 randomly generated search states with constant intensity between 0.01 and 0.30 (1 to 30 false targets in R). Note that L_{CSR} depends on the integer number of data points.

Figure 2.2. Log-likelihood of Feature Detection for Completely Spatially Random Data

2.5 Implementation

We implement this model in General Algebraic Modeling System (GAMS) Version 24.9 with an interface to R via R-Studio and the package gdxrrw [19]. We use the CONOPT nonlinear solver to optimize the MLE with close attention to the bounds and initial values for key variables as outlined in Table 2.1. Note that our search region R is defined by $R = \{(x_1, x_2)|x_1 \in [-5, 5], x_2 \in [-5, 5]\}$.

Table 2.1. Bounds on Variables

Parameter	Lower Bound	Upper Bound	Initial Value
C_1	-5	5	$\bar{x}_1$
C_2	-5	5	$\bar{x}_2$
η_1	0.25	2.5	$\sigma(x_1)$
η_2	0.25	2.5	$\sigma(x_2)$
α	0.0	π	$\pi/2$

We bound all decision variables above and below for the solver. We also provide reasonable initial values to get us close to a likely solution.

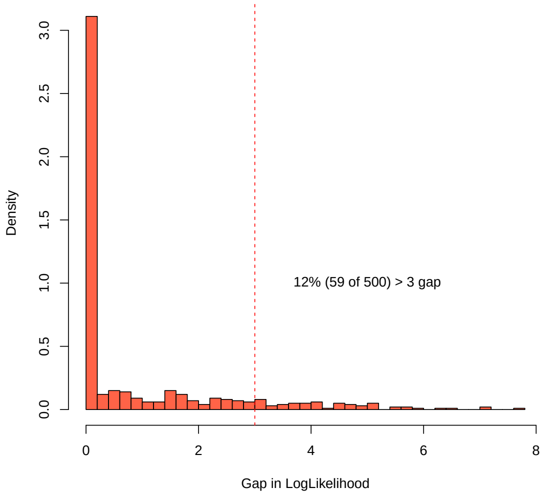
As this is a non-convex optimization problem, we also apply multiple random restarts to improve our results. We can compare the improvement using up to n restarts to our first effort with no randomly initialized solves. Figure 2.3 shows this data and justifies our decision to run a single solve for the model in the interest of computational efficiency. Implementations using multiple restarts may be able to gain marginal improvements in efficiency, but at considerable computational expense.

2.6 Results

We simulate data in some basic cases to generate estimates for comparison with the known source distribution. We can then evaluate its ability to accurately and consistently detect a feature within data.

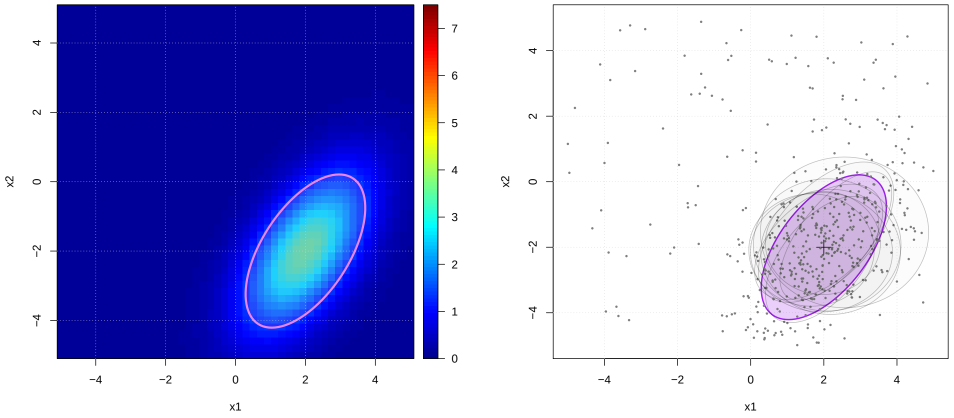
2.6.1 Accuracy

We show that this estimation method can successfully detect features in data consistent with the model. We use as a source distribution $x^i \in \boldsymbol{X} \sim \text{Pois}(\lambda(\boldsymbol{x}, \boldsymbol{\theta}_{\text{SRC}}))$, where $\boldsymbol{\theta}_{\text{SRC}} = (C_1 = 2.0, C_2 = -2.0, \eta_1 = 2.5, \eta_2 = 1.25, \rho = 0.1, D = 3.5, A = 1.0, \beta = 1.0)$. This source distribution is plotted in Figure 2.4 along with ten estimates.



This plot shows the improvement (or "gap") in log-likelihood resulting from running 10 random restarts in addition to the initial model solve. This shows that restarts are not needed to produce good results in terms of log-likelihood from estimation, so we can solve this problem once and accept that initial solution.

Figure 2.3. Effect of Restarts on Likelihood of Intensity Estimation

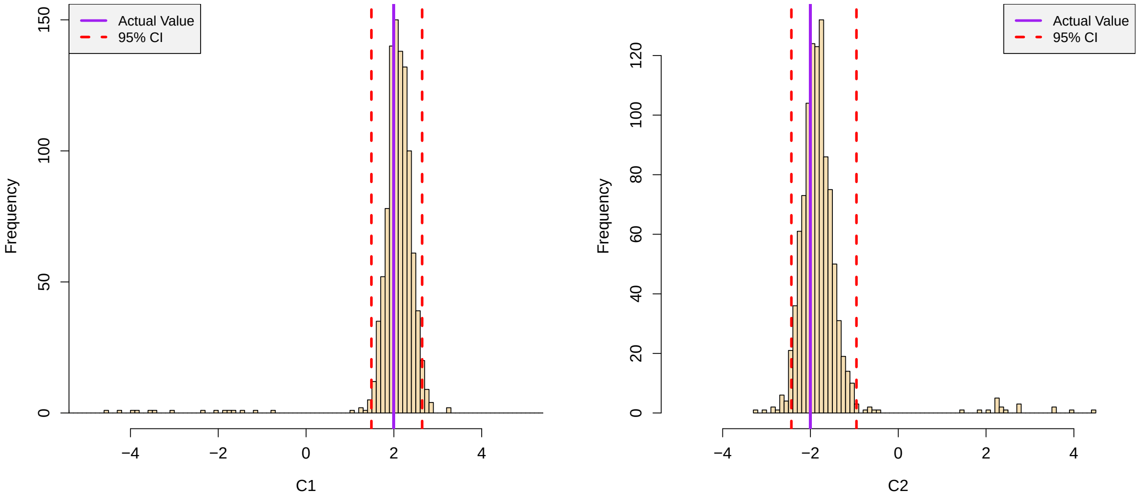


(a) Actual Source Intensity $\lambda(\boldsymbol{x}, \boldsymbol{\theta}_{\mathsf{SRC}})$ (b) Ten Samples and Intensity Estimates $\hat{\boldsymbol{\theta}}_{\mathsf{SRC}}$

These plots show the intensity of the NHPP used to generate search regions in this chapter, following the elliptical model of Section 2.3. The purple ellipse represents $\boldsymbol{\theta}_{\mathsf{SRC}}$. The second plot shows ten samples drawn from this distribution and overlaid, along with the ten MLE ellipses representing the estimated parameters $\hat{\boldsymbol{\theta}}$.

Figure 2.4. Source Distribution with Samples

We can see that there may be some bias in our estimator for some of the parameters; in particular it has difficulty capturing the correct dimensions of the ellipse η_1 and η_2, and the rotation angle of the ellipse A. These are not independent biases but result from the structure of our intensity function and $\boldsymbol{\theta}_{\text{SRC}}$. However, the location of the feature is quite accurate $C = (C_1, C_2)^T$, as plotted in Figure 2.5 over 1000 samples.



(a) C_1 Location

(b) C_2 Location

The MLE estimation method is quite accurate at estimating the location of feature from $\boldsymbol{\theta}_{\text{SRC}}$ within the region R. These histograms represent estimates of the location parameters C_1 and C_2 from 1000 contact data samples drawn from $\boldsymbol{\theta}_{\text{SRC}}$.

Figure 2.5. Accuracy of Feature Location

2.6.2 Consistency: Bootstrap

We can use the bootstrap to estimate the variance of these parameter estimates. To do so, we generate bootstrap samples of the contacts detected in R, and run our MLE optimization on each bootstrapped sample. The results of a number $nBoot$ of bootstrap samples allow us to estimate the variance of our estimated parameters. This is computationally intensive.

There is some discussion of use of the bootstrap in estimating the intensity of a non-homogeneous spatial process. One paper [20] evaluates bootstrapping a spatial Poisson process to determine statistics describing the point process. Section 2 of [20] discusses a traditional bootstrap of the data points observed. One drawback pointed out is that the resulting bootstrap samples - n points sampled with replacement from the n observed data points - will have some points repeated multiple times, which is not the case in the original data and could lead to differences in behavior of the estimator. Evaluation of these methods shows that these drawbacks are minimal, and the bootstrap can give quite accurate intervals with as few as 50 samples. See Appendix A for details on finding confidence intervals for the estimated parameters of a spatial Poisson process. From there, we can see that a rough bound given by the Modified Wald Interval the order of

$$100(1-\alpha)\%\text{CI for } \lambda(\boldsymbol{x}) = \left[\hat{\lambda}(\boldsymbol{x}) - Z_{\alpha/2}\sqrt{\hat{\lambda}(\boldsymbol{x})}, \hat{\lambda}(\boldsymbol{x}) + Z_{\alpha/2}\sqrt{\hat{\lambda}(\boldsymbol{x})}\right] \tag{2.12}$$

is acceptable, where $Z_{\alpha/2}$ is the appropriate normal critical value. Taking the computational time to perform the bootstrap significantly tightens this bound over R away from the feature where λ is near zero.

CHAPTER 3: Estimated Posterior Probability Distributions

3.1 Background and Overview

In Chapter 2 we discussed our method to estimate the intensity of a NHPP. In this chapter, we discuss how we apply additional soft information to these estimates to improve our prior probability distributions for the real and false targets. We first extend the formulation of Section 2.3 to account for the fact that the search region R has not been uniformly searched, and then develop the tools to build better estimates of the target densities.

3.1.1 Overview

In a search with no false targets, it is straightforward to find the PDF for the target's location. Starting with the prior, searchers use Bayes' Theorem to reduce the probability density proportional to the search function of effort applied. In a search with false targets, we have the opportunity to consider more information.

At any point in the search, we have three types of information and data available to us. First, there are the priors, including the probability distribution for the real target and the Poisson intensity for false targets. Second, there is the search progress, of broad area search effort that has been applied over R. Third, we have the search results, including both the false targets found and identified, as well as COIs that have been found, investigated with some amount of contact identification effort, but not yet identified.

We calculate a COI intensity function as an intermediate step. With this, we separate a COI intensity into target density and false target intensity by making assumptions that enable us to separate the two in certain cases with real-world application. This allows us to use all available information to output the updated target location probability, both over the search region R and for each unidentified COI, and the updated intensity of false targets that have not yet been detected.

3.2 Assumptions

We make the following assumptions, extending and adding to those made in Section 2.2:

1. *Assume a background level of false target clutter represented by a HPP.* This means the sea trash is randomly and evenly distributed, and the region R is not so large that the density of the background clutter varies. These are the unrelated false targets.
2. *Assume the event that created the target also created false targets represented by an NHPP.* These are the related false targets. The NHPP has a region of higher intensity, or feature, that has some shape, orientation, and other properties. Here we assume the feature is elliptical.
3. *Assume the distribution of the real target is the intensity of the NHPP divided by its integral over R, i.e. scaled to 1.0.* This is not required in all cases to use this method.
4. *Assume COIs cannot be distinguished by the broad search sensor.* That is, we do not have any information whether a COI is a target or false target until we investigate it. Further, we do not discriminate whether a false target is "related" or "unrelated" to the target even after we have investigated it. This is a simplifying assumption that may not always hold in practice.

We also assume we have the following additional prior information:

1. *Search region R.* This is the area over which the search takes place. We assume R contains almost all of the PDF for the target's location ($> 99\%$). It is related to but not entirely determined by the prior target distribution. We assume the target is somewhere within the search region, or that the search region can be expanded if needed.
2. *Prior target distribution $f(\boldsymbol{x})$.* This is derived from whatever information we have about the target already. For example, there could be satellite data from when the missing airliner reported its location via a Global Positioning System (GPS) transponder. This is the distribution for the probability the target is located at any given point $\boldsymbol{x}$ in R.
3. *Prior false target distribution $\delta(\boldsymbol{x})$.* This may be an informed estimate based on past operations in this area, or a best guess.
4. *Search functions $b(\boldsymbol{x}, z), B(\boldsymbol{x}, z), a(\boldsymbol{x}, w), A(\boldsymbol{x}, w)$.* These are the search functions giving the probability of success for any given application of search effort applied -

assuming there is such a target there. We assume detection events are independent.

- $b(\boldsymbol{x}, z)$: Real target broad search detection function given broad search effort z applied at $\boldsymbol{x}$
- $B(\boldsymbol{x}, z)$: False target broad search detection function given broad search effort z applied at $\boldsymbol{x}$
- $a(\boldsymbol{x}, w)$: Real target contact ID function given contact ID effort w applied at $\boldsymbol{x}$
- $A(\boldsymbol{x}, w)$: False target contact ID function given contact ID effort w applied at $\boldsymbol{x}$
- U: Broad search effort application rate (per time)
- Λ: Contact ID effort application rate (per time)

3.3 Non-Uniform Broad Search Effort

The formulation in Equation (2.2) estimates an intensity uniformly over the entire region. However, in an actual search, we will not have applied broad search effort uniformly over the region. There may be part of the region where we apply very little effort, and have a low probability of detecting any COIs present; and other locations where we have applied a large amount of effort and are almost certain to have detected any COIs there. This means that not all COIs are equally probable, so they should have different contributions to the estimated intensity. A COI found in a location with a low probability of detection based on applied broad search effort should count for more intensity in the estimate than a COI detected in a region of high detection probability.

We deal with this in the following way. We consider the searcher to be not just detecting contacts, but also observing intensity. The observed intensity $\lambda_{\text{obs}}(\boldsymbol{x})$ at any point $\boldsymbol{x}$ is then the product of the actual intensity $\lambda_{\text{act}}(\boldsymbol{x})$ and the probability of detection at $\boldsymbol{x}$ given the amount of effort the searcher has applied there $w(\boldsymbol{x})$. Thus,

$$\lambda_{\text{obs}} = \lambda_{\text{act}} B(\boldsymbol{x}, z(\boldsymbol{x})). \tag{3.1}$$

We use $B(\boldsymbol{x}, w(\boldsymbol{x}))$ rather than $b()$, because we are considering all of our COIs to be false targets as discussed in 3.4.

To avoid confusing the actual and observed intensities, we need to be clear about which λ

we are estimating in our MLE formulation 2.3. There, we are maximizing the likelihood of the data and contacts we actually observed, $L(\lambda_{\text{obs}})$ – but we want our result in terms of the *actual* intensity λ_{act}. We then have:

$$L(\lambda_{\text{obs}}) = \sum_{i=1}^{\nu} \log \lambda_{\text{obs}}(\boldsymbol{x}^i) - \int_R \lambda_{\text{obs}}(\boldsymbol{x})\, d\boldsymbol{x} \tag{3.2}$$

$$L(\lambda_{\text{obs}}) = \sum_{i=1}^{\nu} \log \left(\lambda_{\text{act}}(\boldsymbol{x}^i) B(\boldsymbol{x}^i, z(\boldsymbol{x}^i))\right) - \int_R \lambda_{\text{act}}(\boldsymbol{x}) B(\boldsymbol{x}, z(\boldsymbol{x}))\, d\boldsymbol{x} \tag{3.3}$$

$$\lambda_{\text{act}}(\boldsymbol{x}) = \rho + De^{-\xi(\boldsymbol{x})}. \tag{3.4}$$

We can use (3.3) to replace (2.2) in this case with non-uniform application of effort. (3.4) clarifies that our parameters $\boldsymbol{\theta}$ refer to the *actual* intensity λ_{act} as desired.

If broad search was evenly applied over R there should be no difference, but COIs $\boldsymbol{x}^i$ where little broad search effort $z(\boldsymbol{x})$ was applied, will "count" more. In the second term, there is a reduced penalty for parts of R that have had very little search effort applied and thus have low B. We demonstrate an example of how this works in Figure 3.1.

When calculating the intensity for non-uniform application of search effort as in Section 3.3, we use a floor value of $B(\boldsymbol{x}, z)$ of 0.05. This is necessary to ensure there is a counterweight term in the MLE objective pulling the intensity down; this does not represent actual probability of detection, but our preference to penalize unjustified estimates of high intensity in areas we have not searched yet.

3.4 Target Distribution Functions

Since we assume only a single real target, we assume that all COIs found are false targets for purposes of intensity estimation.

3.4.1 Separation of PDFs

If we have a COI distribution, we need to pull out the real target PDF from the false target intensity. We can use our assumptions from Section 3.2 to do so. When our MLE estimation locates the single feature, we assume that this is where the target is located. We use the feature parameters, and scale the intensity so it integrates to 1.0 over R. We use this as our estimated real target PDF, $f_{\text{est}}(\boldsymbol{x})$:

$$f_{\text{est}}(\boldsymbol{x}) = \frac{De^{-\xi(\boldsymbol{x})}}{\int\limits_{R} De^{-\xi(\boldsymbol{x}')}\,\mathrm{d}\boldsymbol{x}'}. \tag{3.5}$$

Our estimated false target intensity δ_{est} then is the sum of the background intensity – the unrelated false targets – and the remaining feature intensity, the related false targets:

$$\delta_{\text{est}}(\boldsymbol{x}) = \rho + \left(1 - \frac{1}{\int_{R} De^{-\xi(\boldsymbol{x}')}\,\mathrm{d}\boldsymbol{x}'}\right) De^{-\xi(\boldsymbol{x})}. \tag{3.6}$$

As we can see, equations (3.5) and (3.6) sum to λ_{act} as in Equation (3.4).

3.5 Updated Distributions

At any point in the middle of the search, we are able to estimate the real and false target intensity based on the search conducted so far and the COIs detected. In order to use all available information, we need to combine the prior information about the location of the target with the new information we gained through estimates of the intensity based on search effort and COIs.

One approach would be to simply accept the estimated intensity given by (3.4). However, it is possible our search has covered only a small portion of R. In this case, we may believe we have more, better information available from the prior target location distribution than from the search. Alternately, if R has been almost fully searched, we may no longer trust the prior distribution, instead feeling confident in our estimate based on a nearly complete understanding of the various targets present in R. We consider two ways to combine the prior information with the new estimates: a Bayesian method, and an heuristic.

3.5.1 Bayesian Update to Real Target Distribution

Bayes' Theorem offers a way to include all information. To proceed, we first assume that our description of the prior distribution function for the real target, $f_{\text{pri}}(\boldsymbol{x})$, is not just a single PDF but a set of distributions for each element of $\boldsymbol{\theta}_{\text{prior}} = (C_1, C_2, \eta_1, \eta_2, \alpha)^T$.

Note that we do not need ρ or D in the hyperparameter vector $\boldsymbol{\theta}$, since ρ subtracts out, and D divides out, when we are finding the probability of the real target as in Equation (3.5).

Briefly, the posterior is proportional to the likelihood times the prior. If the known, current state of our search is given by $X = \{\boldsymbol{x}^i, z(\boldsymbol{x})\}$, including both the points observed and effort applied, then in terms of probabilities $p(\cdot)$ and likelihood l:

$$p(\lambda \mid X) \propto l(\lambda \mid X)\, p(\lambda). \tag{3.7}$$

Since λ is determined by its parameters $\boldsymbol{\theta}$, $\lambda(\boldsymbol{x}) = \lambda(\boldsymbol{x}, \boldsymbol{\theta})$, we can write the posterior probability in terms of $\boldsymbol{\theta}$,

$$\begin{aligned} p(\lambda) &= p(\boldsymbol{\theta}) \\ p(\boldsymbol{\theta} \mid X) &\propto l(\boldsymbol{\theta} \mid X)\, p(\boldsymbol{\theta}). \end{aligned} \tag{3.8}$$

Posterior probabilities are only proportional until normalized by multiplying by $1/p(X)$, the inverse of the prior (marginal) probability of X.

We use a hierarchical Bayesian point process model; see [21] Chapter 8. The hyperparameters in the terms of Bayesian hierarchical models are the components of $\boldsymbol{\theta} = (C_1, C_2, \eta_1, \eta_2, \alpha)^T$; $\boldsymbol{\theta} \in \Theta$, which is defined as in Table 2.1.

We start by finding the log-likelihood. We have L from Equation (3.3); we drop the "actual" from our λ, clarify the variable of integration, and state in terms of hyperparameters $\boldsymbol{\theta}$ given the known search state X:

$$L(\boldsymbol{\theta} \mid X) = \sum_{i=1}^{\nu} \log\left(\lambda(\boldsymbol{x}^i, \boldsymbol{\theta}) B(\boldsymbol{x}^i, z(\boldsymbol{x}^i))\right) - \int_R \lambda(\boldsymbol{x}', \boldsymbol{\theta}) B(\boldsymbol{x}', z(\boldsymbol{x}'))\, \mathrm{d}\boldsymbol{x}' \tag{3.9}$$

or in terms of likelihood l,

$$l(\boldsymbol{\theta} \mid X) = \prod_{i=1}^{\nu} \left(\lambda(x_i, \boldsymbol{\theta}) B(x_i, z(x_i))\right) \exp\left(- \int\limits_R \lambda(x', \boldsymbol{\theta}) B(x', z(x'))\, \mathrm{d}x' \right) \tag{3.10}$$

This is Equation (8.12) from [21]. Now we multiply the prior (in terms of $\boldsymbol{\theta}$) times the likelihood (also in terms of $\boldsymbol{\theta}$ as well as x_i). This gives us the posterior probability for λ.

$$p(\lambda \mid X) \propto l(\boldsymbol{\theta} \mid X)\, p(\boldsymbol{\theta}) \tag{3.11}$$

Now we can find the posterior expected intensity at any given point $\boldsymbol{x}$ given X,

$$E[\lambda] = \int\limits_{\lambda} p(\lambda)\lambda \, \mathrm{d}\lambda \tag{3.12}$$

$$\propto \int\limits_{\boldsymbol{\theta}\in\Theta} l(\boldsymbol{\theta} \mid X) p(\boldsymbol{\theta}) \lambda(\boldsymbol{\theta})\, \mathrm{d}\boldsymbol{\theta} \tag{3.13}$$

$$E[\lambda(\boldsymbol{x}) \mid X] \propto \int_{\boldsymbol{\theta}\in\Theta} p(\boldsymbol{\theta}) \prod_i \lambda(\boldsymbol{x}_i, \boldsymbol{\theta}) B(\boldsymbol{x}_i, z(x_i))\, e^{-\int_R \lambda(x', \boldsymbol{\theta}) B(x', z(x'))\, \mathrm{d}x'} \lambda(\boldsymbol{x}, \boldsymbol{\theta})\, \mathrm{d}\boldsymbol{\theta}. \tag{3.14}$$

The posterior expected intensity $E[\lambda \mid X]$ in (3.14) is the equivalent of the maximum likelihood intensity calculated in Chapter 2. If we are considering the intensity λ in this discussion to be the real target PDF, $f(\boldsymbol{x})$, we can normalize it to a proper probability by integrating $\lambda(\boldsymbol{x})$ over all of our search region R to total probability 1.0. This is a difficult integral to calculate numerically. We plot one example, assuming homogeneous application of broad search effort, in Figure 3.2.

This method has the advantage that it can replace the need for the GAMS/CONOPT solver-based maximization of the likelihood function to estimate the intensity. However, it has two significant disadvantages. First, it is computationally demanding; the posterior target PDF in Figure 3.2 took 24 hours to compute using 35 cores at 2.30 GHz in a brute-force approach. Efficiency gains are possible with more sophisticated implementation, but these calculations must be done many times in order to apply the optimization techniques in Chapter 4. Second, the Bayesian posterior provides an intensity that generally fits the given model with a single elliptical feature (in this case). This means the intensity tends to "split the difference" between the prior and the data, if the two are different. This leaves relatively little probability at either the prior or the data-estimated location, with most of the probability mass between the two – where there may be no information to suggest the target is. It would be generally preferable to output a bi-modal probability distribution rather than a unimodal one that inadequately represents either component of the convolution. We could attempt to refine the model to allow for this, but instead – rather than adding to the already daunting computational demands of the current Bayesian method – we use another technique.

3.5.2 Heuristic

In lieu of using this Bayesian methodology throughout our search optimization method, we use a simple heuristic in combination with our existing MLE estimate from Chapter 2. Instead of completely discarding the prior and adopting the estimate from the data, we first determine the percentage coverage of the search region R, ϕ_{cov}. We then use this fraction of the estimated target probability, and its complement, the unsearched factor $1 - \phi_{cov}$, of the prior. This allows us to use more of the data as we progress through the search, not fully discarding the prior until we have full coverage of the entire search area. We define the coverage factor as:

$$\phi_{\text{cov}} = \int\limits_R b(\boldsymbol{x}, z(\boldsymbol{x}))\,\mathrm{d}\boldsymbol{x}. \tag{3.15}$$

We could also consider not just how much searching was done to date, but also how strongly the feature identified by our estimation of $\hat{\lambda}$ presented itself. Features that strongly popped out above the background intensity would be weighted more heavily than those which barely increased likelihood over a constant background level of noise. This can be achieved by using a ratio of the log-likelihood of $\hat{\lambda}_{\text{est}}$ (L_{MLE}) to the log-likelihood of a CSR where the feature intensity parameter D is 0 and there is only constant background COI intensity ρ (L_{const}).

We can use the χ^2 p-values from Section 2.4 as a approximate value for our confidence in the feature we have detected, or a logistic function of the difference in L, which is similar. This allows us to construct a weighted coverage factor for the heuristic, ϕ_{heur}, and real target distribution f_{heur},

$$\begin{aligned}\phi_{\text{heur}} &= \text{coverage} \times \text{feature confidence} \\ &= \phi_{\text{cov}} P(\chi_6^2 \geq \Delta) \qquad (3.16) \\ &\approx \phi_{\text{cov}} \frac{1}{1 + e^{-(L_{\text{MLE}} - L_{\text{const}} - 3)}} \qquad (3.17) \\ f_{\text{heur}}(\boldsymbol{x}) &= \phi_{\text{heur}} f_{\text{est}}(\boldsymbol{x}) + (1 - \phi_{\text{heur}}) f_{\text{prior}}(\boldsymbol{x}). \qquad (3.18)\end{aligned}$$

For estimating the false target intensity, we are not as concerned about the feature detection. Since we are assuming a constant level of background clutter, we do not need to have covered as much of the search region to make a good estimate of the background density. Therefore we propose a more lenient rule for the false target intensity, using a scaling factor ψ_{false}:

$$\begin{aligned}\psi_{\text{false}} &= 1 - \exp\left(-\phi_{cov}/\phi_0\right) \qquad (3.19) \\ \delta_{\text{heur}}(\boldsymbol{x}) &= \psi_{\text{false}} \delta_{\text{est}}(\boldsymbol{x}) + (1 - \psi_{\text{false}}) \delta_{\text{prior}}(\boldsymbol{x}). \qquad (3.20)\end{aligned}$$

We use a value of $\phi_0 = 0.25$ as a reasonable percentage of the search region to give us confidence in our estimate of the false target intensity. We show an example of results from the heuristic in Figure 3.3. We do not claim that this heuristic is optimal, or near it, in any search conditions. It is a reasonable, efficient way of combining the information available while attempting to avoid over-fitting data. As such, it is sufficient as a starting point to demonstrate an improvement over current methods that do not incorporate improved estimates of target densities.

3.6 Posterior Distributions

After finding $f_{\text{heur}}(\boldsymbol{x})$ and $\delta_{\text{heur}}(\boldsymbol{x})$, we then need to calculate $\tilde{f}_{\text{heur}}(\boldsymbol{x})$ and $\tilde{\delta}_{\text{heur}}(\boldsymbol{x})$ to ensure we are using the residual densities considering the search effort we have already applied.

3.6.1 Unidentified COIs

Now we calculate the probability that each COI which has been detected but not identified is the target. We use the methods from Section 7 of [9]. We define π_i to be the probability that a COI at $x_i \in X = \{x_1, ..., x_v\}$ is the target of our search. We assume we have found our best estimates for f and δ (such as $f_{\text{heur}}(\boldsymbol{x})$ and $\delta_{\text{heur}}(\boldsymbol{x})$).

From [9], we begin with Equation (7.2) with updated notation, giving the probability of detecting at $\boldsymbol{x}$ a real target $u(\boldsymbol{x})$ or a false target $r(\boldsymbol{x})$, and the probability a contact $\boldsymbol{x}^i$ is the real target π:

$$
\begin{aligned}
u(\boldsymbol{x}) &= f(\boldsymbol{x})b(\boldsymbol{x}, z(\boldsymbol{x})) && (3.21)\\
r(\boldsymbol{x}) &= \delta(\boldsymbol{x})B(\boldsymbol{x}, z(\boldsymbol{x})) && (3.22)\\
Z(\boldsymbol{x}) &= \text{Total broad search effort applied at } \boldsymbol{x}\\
P[Z] &= \text{Probability of detecting the (real) target with broad search allocation } Z\\
&= \int\limits_R f(\boldsymbol{x})b(\boldsymbol{x}, Z(\boldsymbol{x}))\,\mathrm{d}\boldsymbol{x}. && (3.23)\\
\pi(x_i) &= \frac{u(x_i)/r(x^i)}{(1 - P[Z]) + \sum\limits_{j=1}^{v} u(\boldsymbol{x}^j)/r(\boldsymbol{x}^j)}. && (3.24)
\end{aligned}
$$

These formulas from [9] assume that no contact investigation effort has been applied to any COI. This may be considered equivalent to the assumption that contact investigation is memoryless, so the length of time a contact has been unsuccessfully investigated tells us nothing about whether it is or is not the false target. Neither assumption is fully justifiable in our intended context, however, as contacts may be partially investigated in the course of the search, and a and A are not required to be equal. In our semi-adaptive search construct, we anticipate having multiple COIs that have been unsuccessfully investigated. We therefore modify these equations by allowing for the possibility that contacts have been unsuccessfully investigated with contact identification effort w_i.

To do so, we redefine $r(\boldsymbol{x})$ and $u(\boldsymbol{x})$ as follows, since $(1 - a(\boldsymbol{x}, w))$ and $(1 - A(\boldsymbol{x}, w))$ are the probabilities that a real and false target will still be unidentified after the application of contact ID effort w^i to a contact $\boldsymbol{x}^i$:

$$
\begin{aligned}
\tilde{u}(\boldsymbol{x}^i, w^i) &= f(\boldsymbol{x}^i)b(\boldsymbol{x}^i, z(\boldsymbol{x}^i))(1 - A(\boldsymbol{x}^i, w^i)) && (3.25)\\
\tilde{r}(\boldsymbol{x}^i, w^i) &= \delta(\boldsymbol{x}^i)B(\boldsymbol{x}^i, z(\boldsymbol{x}^i))(1 - a(\boldsymbol{x}^i, w^i)). && (3.26)
\end{aligned}
$$

We can now use these posterior values in our expression for the posterior probability each COI is the target, $\tilde{\pi}_i$, as follows:

$$\tilde{\pi}(\boldsymbol{x}^i, w^i) = \frac{\tilde{u}(\boldsymbol{x}^i, w^i)/\tilde{r}(\boldsymbol{x}^i, w^i)}{1 - P[Z] + \sum_{j=1}^{\nu} \tilde{u}(\boldsymbol{x}^j, w^j)/\tilde{r}(\boldsymbol{x}^j, w^j)}. \tag{3.27}$$

This will be close to π_i if a is close to A, and $\tilde{\pi}_i = \pi_i$ if $w_i = 0$.

3.6.2 Undetected COIs

We need to find the posterior distributions for undetected real and false targets after we have applied an amount of broad search effort over R. The estimates f_{est} and δ_{est} in Equations (3.5) and (3.6) represent the actual, total distribution of targets, but we have already detected some of them. We need to find the intensity of undetected targets, or the non-observed intensity. As in optimal search, we use Bayes' Theorem to reduce them by the applied broad search probability. Again, we start by following Stone in [10], defining:

Stone uses this to give the posterior real target distribution, given the single real target has not been found (also listed as 6.6.4 in [6]):

$$\tilde{f}^{(1)}(\boldsymbol{x}) = \frac{f(\boldsymbol{x})\,[1 - b(\boldsymbol{x}, Z(\boldsymbol{x}))]}{1 - P[Z]}. \tag{3.28}$$

We now allow for unidentified COIs. Therefore we must add a term to the denominator to account for that in the total probability and ensure it sums to 1, similar to in Equation (3.27) and Section 7 of [9],

$$\tilde{f}(\boldsymbol{x}) = \frac{f(\boldsymbol{x})\,[1 - b(\boldsymbol{x}, Z(\boldsymbol{x}))]}{1 - P[Z] + \sum_{i=1}^{\nu} \tilde{u}_i/\tilde{r}_i}. \tag{3.29}$$

Posterior false target intensity is not affected by the partial investigation of contacts as it is not required to scale to 1.0; we use Equation (6.6.12) from [6] without modification:

$$\tilde{\delta}(\boldsymbol{x}) = \delta(\boldsymbol{x})[1 - B(\boldsymbol{x}, Z(\boldsymbol{x}))]. \tag{3.30}$$

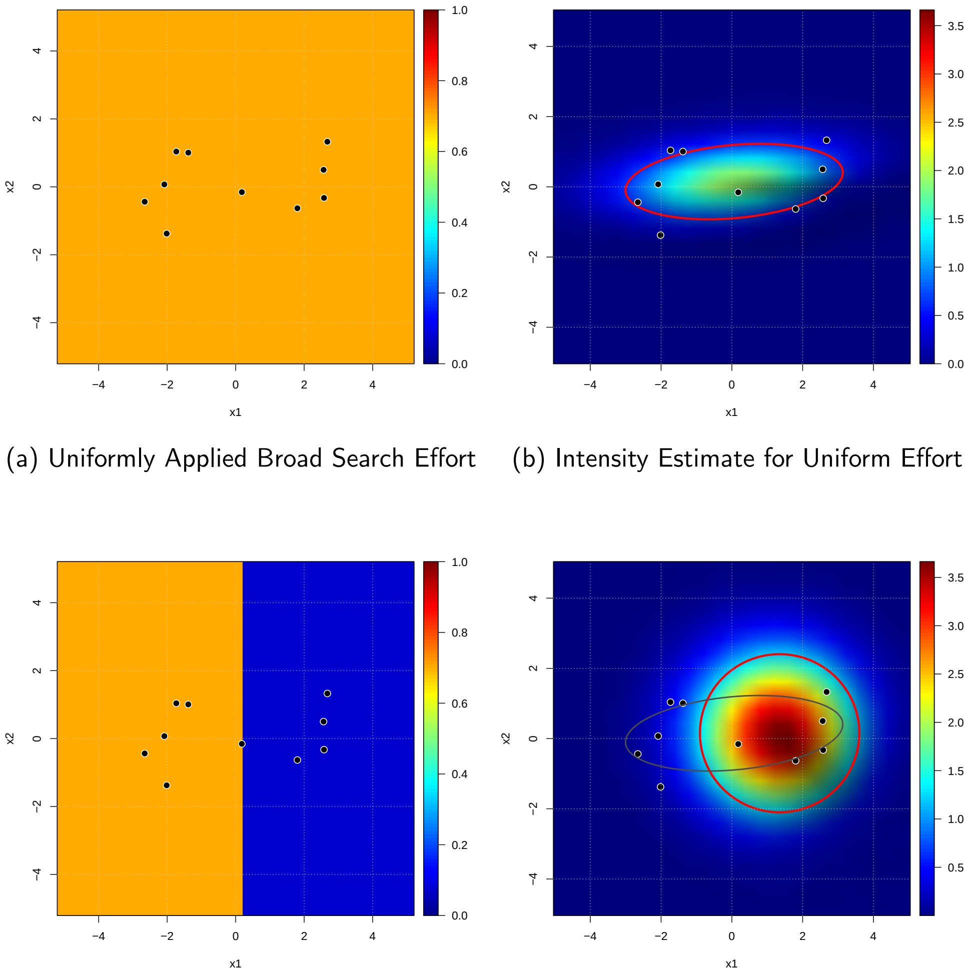
3.7 Implementation

We now have the a mixture distribution for the real target probability: over space and over each COI. We also have a mixture distribution for false targets: over space and for each COI; in addition to all COIs that have already been identified as false targets. We summarize these distributions in Table 3.1. See also Figure 3.4 for an updated outline of the enhancements to the search model.

Table 3.1. Summary of Probability Distributions

Target	**Status**	**Prior**	**Data**	**Estimate**	**Posterior**
Real	Undetected	f_{prior}	f_{est}	f_{heur}	$\tilde{f}_{\text{heur}}$
	COI	-	-	π_i	$\tilde{\pi}_i$
False	Undetected	δ_{prior}	δ_{est}	δ_{heur}	$\tilde{\delta}_{\text{heur}}$
	COI	-	-	$1 - \pi_i$	$1 - \tilde{\pi}_i$
	Identified	-	Known		

We implement the techniques described in this chapter onto the MLE formulation from Chapter 2 in R. This provides the set of inputs needed to optimize the application of search effort, which we discuss in the next chapter.

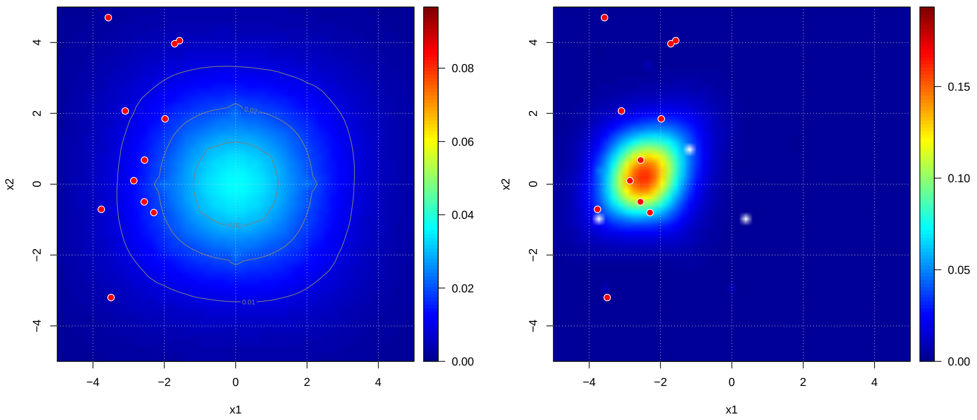


(a) Uniformly Applied Broad Search Effort (b) Intensity Estimate for Uniform Effort

(c) Non-uniform Broad Search Effort (d) Intensity Estimate for Non-uniform Effort

Here we show how the estimated intensity is affected by the broad search effort applied. Both rows use the same randomly generated COIs. The top row uses a search effort uniformly applied over R. The bottom row uses 10x more effort applied in the left half of the region than the right half. This has the expected effect of pulling the intensity estimate to the right while greatly increasing its intensity, since those 5 points may be the equivalent of 50 if they were on the left side of R.

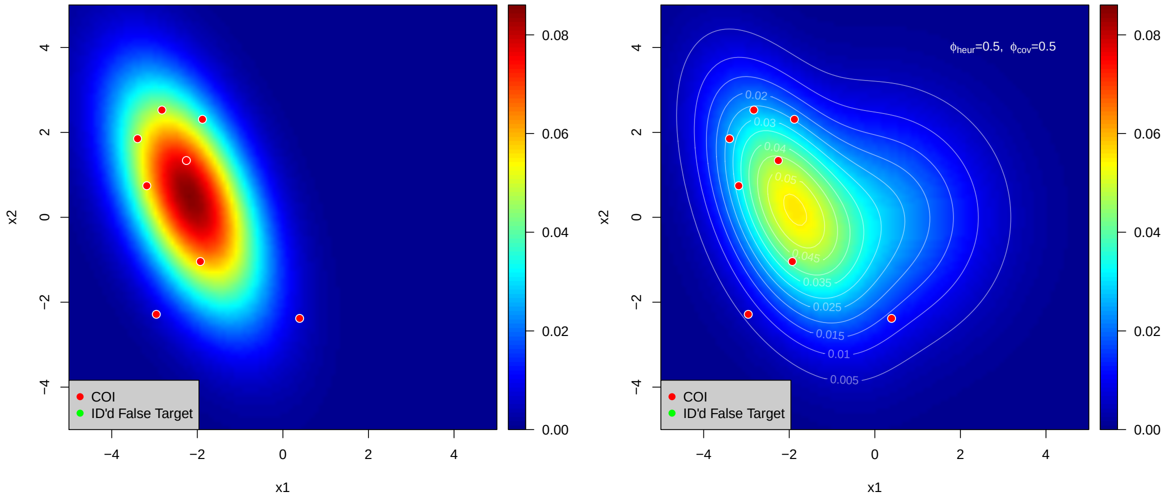
Figure 3.1. Example: Effect of Non-uniformly Applied Effort on Intensity Estimate



(a) Prior Real Target Density (b) Bayesian Posterior Real Target Density

Here we show example results of using a Bayesian Hierarchical Model to find the posterior target density, showing both the prior and posterior over the search region with observed COI locations. This method retains the elliptical model in combining the prior with the data. The posterior plot took 24 hours to produce on a 36-core workstation, yet still has some white pixels with numerical errors from the computation involved. Note differing intensity scales; the prior is at half the scale of the posterior.

Figure 3.2. Bayesian Prior and Posterior Density

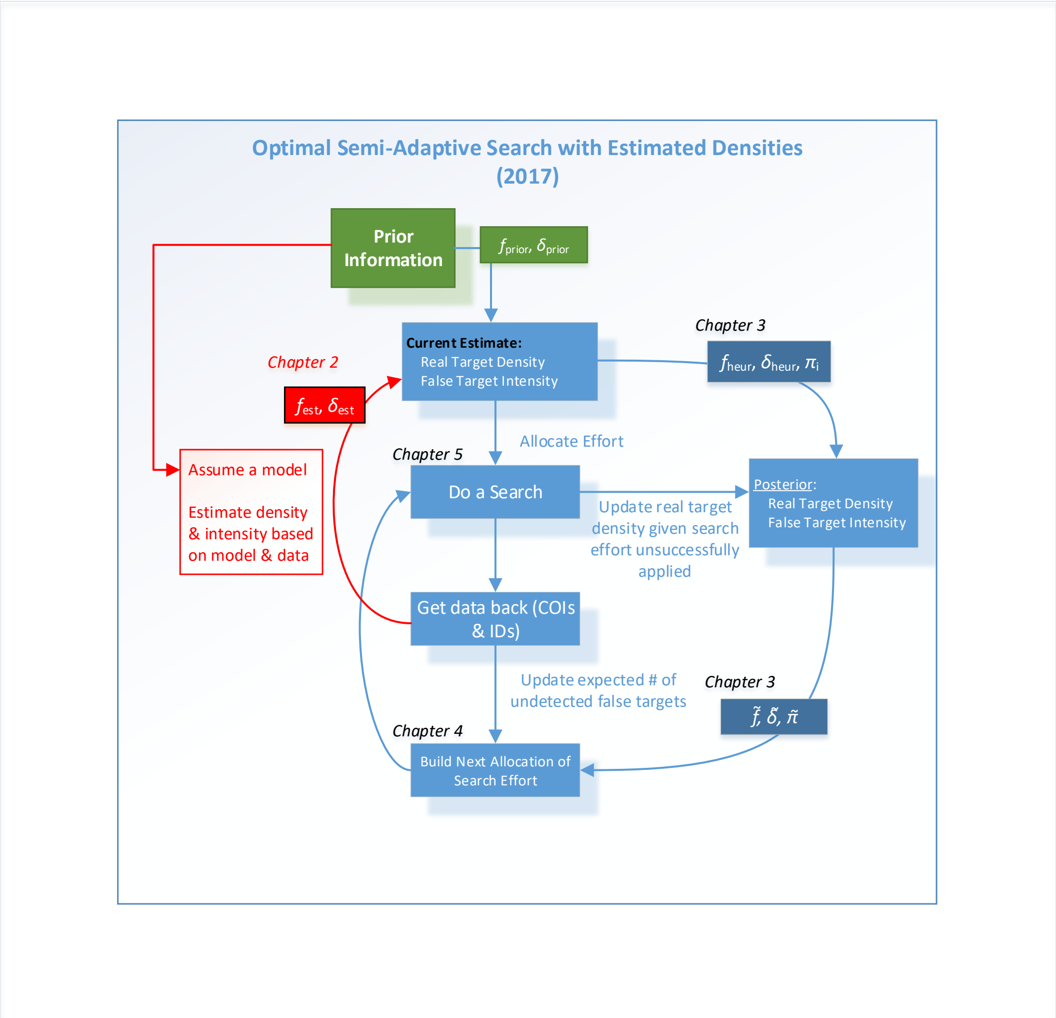


(a) MLE Estimate of Real Target Probability

(b) Heuristic Posterior Real Target Probability

Here we show results of building a usable estimate of real target distribution and false target intensity with the heuristic. In contrast with Figure 3.2, the heuristic combines the prior with the data without attempting to retain an elliptical distribution.

Figure 3.3. Example: Estimate of Target Locations


Optimal Semi-Adaptive Search with Estimated Densities (2017)
Prior Information
f_{prior}, δ_{prior}
Current Estimate:
Real Target Density
False Target Intensity
Chapter 3
f_{heur}, δ_{heur}, π_i
Chapter 2
f_{est}, δ_{est}
Allocate Effort
Chapter 5
Do a Search
Assume a model
Estimate density & intensity based on model & data
Update real target density given search effort unsuccessfully applied
Posterior:
Real Target Density
False Target Intensity
Get data back (COIs & IDs)
Update expected # of undetected false targets
Chapter 3
$\tilde{f}$, $\tilde{\delta}$, $\tilde{\pi}$
Chapter 4
Build Next Allocation of Search Effort

This updates Figure 1.3 showing how we create updated estimates for the density of real and false targets to pass to the search optimization in Chapter 4.

Figure 3.4. False Target Search with Updated Probability Distributions

CHAPTER 4: Search Optimization

4.1 Background and Overview

We can now develop a search plan that optimizes the allocation of future search effort. At any time in the search, we can use the methods in Chapter 3 to update our estimates for the distributions of real and false targets using all available information. We use these estimates to find the "payoff" of each possible allocation of search effort. Payoff is measured in terms of increased probability of detection per unit of time spent searching. We can then select the search effort allocation that gives us the highest payoff.

Possible allocations of effort include broad area search at any location in the search region R, and contact identification effort applied to any previously detected COI. To allow us to include COIs that have already been discovered, we modify the method from [10] as described in Section 4.5. Following Stone et al., we define our search plan in terms of *contingent* contact identification effort [10]. This is the amount of contact identification effort $w(x)$ our plan is willing to apply in case a COI is detected at the location x. The actual amount of effort needed to identify any given contact is a random variable, but we can calculate its expectation using our knowledge of the search functions and probability distributions from Chapter 3.

For any given amount of broad or total search time, we can calculate the optimal payoff rate leveled over all possible allocations of search effort. Then we simply allocate search effort to achieve that payoff rate equally everywhere. Our end result is a search plan Φ that gives the optimal broad area search density function m^*, contingent contact identification density function ω^*, and unidentified COI identification effort allocation $C^* = \{C_1^*, ...C_\nu^*\}$. Φ is then a non-adaptive search plan, given the input real and false target distributions; in Chapter 5 we show how to repeatedly calculate Φ over time to make a semi-adaptive plan.

4.2 Optimal Search

As a starting point and for comparison, we include the algorithm for optimal search allocation in the continuous case without false targets in Appendix B [8].

4.3 Definitions

We adopt the notation in [10], with some modifications and extensions. For clarity, we define our key terms that that we use throughout this chapter:

R = continuous search region in 2-dimensional space

$\boldsymbol{x}$ = a location in R, $\boldsymbol{x} = (x_1, x_2)$

$m(\boldsymbol{x}, s)$ = broad search effort density function

s = an amount of *broad* search time

t = an amount of *total* search time

$z(\boldsymbol{x})$ = a quantity of broad search effort applied at $\boldsymbol{x}$

w = a quantity of contact investigation effort

$\omega(\boldsymbol{x}, t)$ = *contingent* contact investigation effort density function

= contact ID effort willing to exert at $\boldsymbol{x}$ by total time t if a COI is found there

$\boldsymbol{x}^i$ = All COIs previously located but not confirmed as false targets for $i \in \{1, \ldots, \nu\}$

C_i = Amount of contact ID effort applied to unidentified COI i

Φ = Search plan, in terms of broad area density, contingent COI density, and COI effort.

= $\{m(x), \omega(\boldsymbol{x}), C = \{C_1, \ldots, C_\nu\}\}$.

We continue to use the search functions as defined in Section 3.2, repeated here for convenience:

$b(\boldsymbol{x}, z)$ = Real target broad search detection function given broad search effort z applied at $\boldsymbol{x}$

$B(\boldsymbol{x}, z)$ = False target broad search detection function given broad search effort z applied at $\boldsymbol{x}$

$a(\boldsymbol{x}, w)$ = Real target contact ID function given contact ID effort w applied at $\boldsymbol{x}$

$A(\boldsymbol{x}, w)$ = False target contact ID function given contact ID effort w applied at $\boldsymbol{x}$

U = Broad search effort application rate (per time)

Λ = Contact ID effort application rate (per time).

4.4 Divided Search Effort

We begin by discussing the class of search plans when it is optimal to divide search effort between broad search and contact investigation. We reproduce Stone et al.'s method for optimization of a search under these conditions from [10] Section (4.1). We use this as a basis for modifications to better implement it in the context of a semi-adaptive search with density estimates.

4.4.1 Assumptions

We adopt and comment on the assumptions from [10] (4.1).

1. The first assumption requires that the broad area sensor is equally as effective against any false targets as against the real target, and that it follows a "law of diminishing returns" as search effort increases. In practice there may be differences in a sensor's ability to detect false targets and the real target; these are difficult to know ahead of time due to a variety of factors.[1] Diminishing returns is a reasonable assumption in practice.
2. The second assumption considers contact identification. In practice, there may be discontinuities to contact identification associated with deep ocean operations, but to satisfy this assumption these must be approximated by continuous functions.[2]
3. The third assumption requires that the return on additional contact investigation effort, in terms of probability of successful identification, is decreasing. It must be easier or faster to identify a real target than a false target, so that the longer the contact identification continues unsuccessfully, the less likely that particular COI is to be the real target.
4. The fourth assumption requires either an upper bound on time for false target identification, or on the return on additional contact investigation. These allow us to match

[1] The condition of the target, and the nature of the false targets, are often unknown, and both are significant in a sensor's detection rate.

[2] One example would be the use of a ROV to investigate a sonar contact. The ROV will need to be launched from the host platform, and lowered to the sea floor, before it can begin its search; and it will then need to be raised and recovered once complete.

contact identification with broad search – ensuring they are not finite to infinite. Both are reasonable in practice.

4.4.2 Optimal Search Plan

In implementing our semi-adaptive search, we use the following method from [10] to find the optimal allocation of search effort.

The multiplier $k > 0$ corresponds to an allocation of broad search effort. It is the rate of return, in terms of probability of search success, for increased allocation of search time, in units of 1/time. For optimality, k should be equal across all possible allocations of effort, region-wide.

The return on additional contact investigation effort ζ, given in probability of success, is [10]

$$\zeta(w) \equiv a'(w)/(1 - A(w)), \tag{4.1}$$

where $a'(w)$ is the derivative of a with respect to w. The amount of effort a searcher can expect to expend investigating a false target at x, if the searcher is willing to expend a maximum of w effort on contact investigation [10], is

$$\alpha(x, w) = \int_0^w [1 - A(\boldsymbol{x}, u)]\, \mathrm{d}u \quad \text{for } w \geq 0. \tag{4.2}$$

Stone et al. account for the additional effort required to investigate false targets by defining the rate of change of the real target detection probability r [10]:

$$r(\boldsymbol{x}, k) = \frac{\Lambda k}{U\Big[\Lambda f(\boldsymbol{x}) a(v(\boldsymbol{x}, k)) - k\delta\alpha(v(\boldsymbol{x}, k))\Big]}. \tag{4.3}$$

Here the rate $r(\boldsymbol{x}, k)$ is a value of $b'(z)$, the rate of change of the real target detection probability, with units 1/effort. It levels the broad area search over R by ensuring broad search effort is applied over the region with the same rate of return, so that broad search effort z could not be moved from one location to another and increase the real target detection probability.

The broad search effort u is defined in [10] as an inverse function of the rate of return of broad search effort, bound by $b'(0)$:

$$\begin{aligned} u(\boldsymbol{x},k) &= \text{an amount of broad search effort } m \text{ corresponding to a multiplier } k \\ &= \begin{cases} b'^{-1}(r(\boldsymbol{x},k)) & \text{if } 0 < r(\boldsymbol{x},k) \le b'(0), \\ 0 & \text{otherwise.} \end{cases} \end{aligned} \tag{4.4}$$

The contact identification effort v is expressed as the inverse of the rate of return of contact identification effort ζ [10]:

$$\begin{aligned} v(\boldsymbol{x},k) &= \text{an amount of contingent contact ID effort } \omega \text{ corresponding to a multiplier } k \\ &= \begin{cases} \infty & \text{if } k \le \dfrac{\Lambda f(\boldsymbol{x})}{\delta}\zeta(\infty), \\ \zeta^{-1}\left(\dfrac{\delta k}{\Lambda f(\boldsymbol{x})}\right) & \text{if } \dfrac{\Lambda f(\boldsymbol{x})}{\delta}\zeta(\infty) < k \le \dfrac{\Lambda f(\boldsymbol{x})}{\delta}\zeta(0), \\ 0 & \text{if } \dfrac{\Lambda f(\boldsymbol{x})}{\delta}\zeta(0) < k. \end{cases} \end{aligned} \tag{4.5}$$

If the region-wide search rate of return k is greater than the $\omega = 0$ rate of return on contact identification available at x, $\Lambda f \zeta(0)/\delta$, then no contact identification effort is applied and $v = 0$. If the region-wide search rate of return is less than that available with infinite application of effort, $\Lambda f \zeta(\infty)/\delta$, then v will be infinite, requiring infinite contact identification effort for optimality. Infinite contingent allocation of contact identification effort is equivalent to ICCI.

To find the broad search time associated with this multiplier k, Stone et al. define $H(k)$, the broad search time s corresponding to a multiplier k [10],

$$H(k) = \begin{cases} \infty & \text{for } k = 0 \\ \displaystyle\int_R \frac{u(\boldsymbol{x},k)}{U}\,dx & \text{for } 0 < k < \infty \\ 0 & \text{for } k = \infty. \end{cases} \tag{4.6}$$

Lastly, γ is the value of the multiplier k that is optimal for the available broad search time

s – leveling the rate of return of effort across all possible allocations [10];

$$\gamma(s) = H^{-1}(s). \tag{4.7}$$

The optimal allocation of broad search effort m and contingent contact identification effort ω for available broad search time s is then [10]:

$$m^*(\boldsymbol{x}, s) \equiv u(\boldsymbol{x}, \gamma(s)) \tag{4.8}$$

$$\omega^*(\boldsymbol{x}, s) \equiv v(\boldsymbol{x}, \gamma(s)) \quad \text{for } \boldsymbol{x} \in R, \quad 0 \leq s \leq \infty. \tag{4.9}$$

4.5 Extensions to Method

Note that the method in Section 4.4.2 does not consider any unidentified COIs when optimizing over the space. This is appropriate for a single plan, such as an ONAP, which starts with no contacts and ends when no more search time is available. A semi-adaptive plan, however, may need to optimize search allocation when there are still unidentified contacts from previous increments of effort.

Additionally, this plan optimizes for a given *broad* search time, disregarding contact identification time as irrelevant to the broad search allocation. However, *total* search time is often what is constraining the search operation, given availability and cost of the physical search assets.

4.5.1 Unidentified COIs

We extend this model to allocate contact identification effort to unidentified COIs, C_i. We use the contact identification effort function (4.5), at each contact's location, using the optimal multiplier, $v(\boldsymbol{x}^i, \gamma(s))$. Instead of calculating f and δ at $\boldsymbol{x}^i$, we use the probabilities $\tilde{\pi}_i$ for f, and – since each COI must be either a real or false target – $(1 - \tilde{\pi}_i)$ for δ. Otherwise, calculation and bounds on $v_{\text{COI}}(\boldsymbol{x}^i)$ are the same as in Equation (4.5):

$$v_{\mathrm{COI}}(\boldsymbol{x}^i, k) = \begin{cases} \infty & \text{if } k \leq \dfrac{\Lambda \tilde{\pi}_i}{(\tilde{\pi}_i)} \zeta(\infty) \\ \zeta^{-1}\left(\dfrac{\delta k}{\Lambda \tilde{\pi}_i}\right) & \text{if } \dfrac{\Lambda \tilde{\pi}_i}{(1-\tilde{\pi}_i)} \zeta(\infty) < k \leq \dfrac{\Lambda \tilde{\pi}_i}{(1-\tilde{\pi}_i)} \zeta(0) \\ 0 & \text{if } \dfrac{\Lambda \tilde{\pi}_i}{(1-\tilde{\pi}_i)} \zeta(0) < k, \end{cases} \tag{4.10}$$

$$C_i(s) = v_{\mathrm{COI}}(\boldsymbol{x}^i, \gamma(s)). \tag{4.11}$$

We take these contact investigation effort values for the $\boldsymbol{x}^i$ to complete our optimal search plan $\Phi^*(s) = \left\{m^*(s), \omega^*(s), C^*(s) = \{C_1^*(s), ..., C_\nu^*(s)\}\right\}$. Since C_i is contact investigation time, its inclusion in Φ does not alter the allocation of effort constrained by broad search time.

4.5.2 Optimizing for Total Time

Our input parameter of interest to a search planner is likely a *total* search time available or remaining. Planning a search in terms of broad search time has reduced practical utility without consideration for the additional time required to investigate false targets present in the search area.

In order to find the optimal search for a given broad time, we can redefine $H(k) = s$ as $H_t(k) = t$, giving the expected total search time for a given multiplier k. H in (4.6) calculates the duration of broad area search. H_t needs to add terms for the expected contingent contact identification effort ω for each COI discovered and the allocated identification of known contacts C_i,

$$H_t(k) = \begin{cases} \infty & \text{for } k = 0 \\ \displaystyle\int_R \frac{u(\boldsymbol{x}, k)}{U}\, dx + \int_R \delta(\boldsymbol{x}) B(\boldsymbol{x}, u(\boldsymbol{x}, k)) \frac{\alpha(\boldsymbol{x}, v(\boldsymbol{x}, k))}{\Lambda}\, d\boldsymbol{x} & \\ \qquad + \displaystyle\sum_i^{\nu} v_{\mathrm{COI}}(\boldsymbol{x}^i, k)/\Lambda & \text{for } 0 < k < \infty \\ 0 & \text{for } k = \infty. \end{cases} \tag{4.12}$$

We define γ_t to be the optimal value of the multiplier k associated with this available amount of total search time t:

$$\gamma_t(t) = H_t^{-1}(t). \tag{4.13}$$

The optimal search plan Φ^* is then defined by $\gamma_t(t)$, substituting t for s and $\gamma_t(t)$ for $\gamma(s)$ in Equations (4.8), (4.9), and (4.11):

$$\Phi^*(t) = \left\{m^*(t), \omega^*(t), C^*(t) = \{C_1^*(t), ..., C_\nu^*(t)\}\right\}. \tag{4.14}$$

4.6 Immediate and Conclusive Contact Investigation

Our search plans allow for inconclusive contact investigation, in contrast to the discussion of ICCI and its relaxation to allow "breathers" but with conclusive investigations in [6]. In [10], breathers and *in*conclusive contact investigations, rather than ICCI, are recommended when conditions (4.1), which we also apply here, hold.

We do not study the alternative set of assumptions (4.2) studied in [10], under which ICCI is optimal. Those state that α/a is nondecreasing as a function of w from $(0, \infty)$. This means that as more contact identification effort is applied to a COI, the expected investigation time per unit probability never increases. Thus there are no diminishing returns to contact investigation. This implies that it is always a beneficial allocation of effort to investigate a COI more, implying immediate and conclusive investigation is the optimal policy.

We simulate ICCI search plans plans for comparison to semi-adaptive plans that allow breathers and interruptions in Chapter 5.

4.7 Expected Outcomes

With an optimal search plan, we can calculate some expected outcomes that are of interest to search planners.

We can write the probability that the time to identify the target is less than or equal to the search time t, as:

$$P(m, \lambda, t) \equiv \int_R f(x)b(x, m(x,t))a(x, \omega(x,t))\,\mathrm{d}x + \sum_i^{\nu} \tilde{\pi}_i a(C_i(t)), \tag{4.15}$$

where P is the probability of having identified the target by total search time t using plan (m, λ). The first term is the probability of identifying the target if it is currently undetected; this is Equation (1.3) from [10]. The second term is the probability of identifying the target if it is currently detected but unidentified - that is, a COI. $P(t)$ has the form of a Cumulative Distribution Function (CDF).

We can also state the mean time to identification of the target, (1.6) in [10]:

$$\mu(\Phi) = \int_0^{\infty} tP'(m, \lambda, t)\mathrm{d}t. \tag{4.16}$$

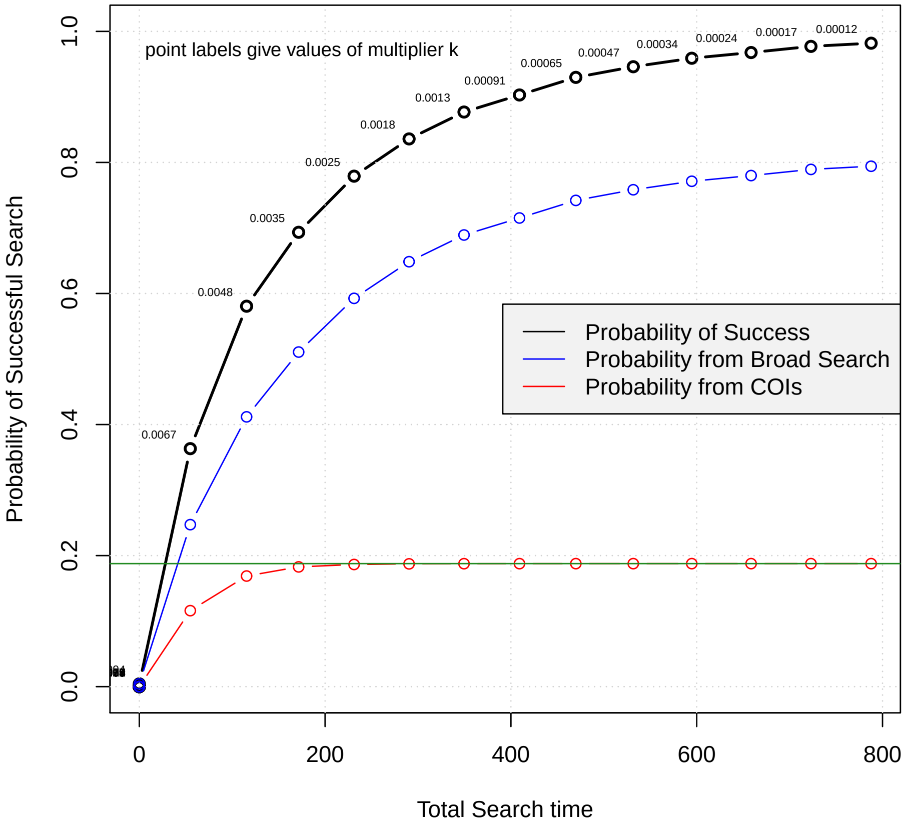
Here $P'(m, \lambda, t)$ is necessarily the PDF, rather than the CDF, found by taking the derivative of P with respect to t. This means it is the rate of return, in terms of increased probability, of additional total search time, which we defined as the multiplier k. Since $H_t(k) = t$, we can then say:

$$\begin{aligned} P'(m, \lambda, t) = k &= H_t^{-1}(t) \\ \mu(\Phi) &= \int_{t=0}^{t=\infty} tH_t^{-1}(t)\mathrm{d}t. \end{aligned} \tag{4.17}$$

We could substitute any reasonable maximum total search time for the upper limit of the integral, and then have the mean time to target identification given t total time available and a success probability of $P(t)$.

Note in real-world search operations, there is uncertainty in the total search time available. There is risk that our search could terminate early, or there is an opportunity to extend it if needed. We could then use tools from decision theory to decide on an optimal plan. One example would be to maximize the expected probability of success given the possible search times available, although there are many possibilities. In this paper, we take the search time available as a given and do not attempt to account for its uncertainty.

We plot an example of the calculated probability of success against total time available *t* in Figure 4.1. We can use the two terms in (4.15) to break out the probability that the real target is detected by broad search or as one of the current COIs by contact identification.



Here we show the relationship between chances of success and total effort applied for an example search, given current estimates mid-operation. It is based on an ONAP snapshot that may be taken at any point in an OSAP. As expected, more search time gives a higher probability of success, which may be broken out as the sum of the probability the target is found through broad search with contingent identification effort (in blue), or by identifying one of the current COIs (in red).

Figure 4.1. Probability of Success vs. Search Time

4.8 Implementation

We implemented this model in R in order to support the simulation study of Chapter 5. This required extensive use of numerical methods for calculations. Solving Equations (4.7) and (4.13) in particular involve two-dimensional integrals, numerical and symbolic derivatives, and inverses calculated by root-finding.

4.8.1 Numerical Approximations

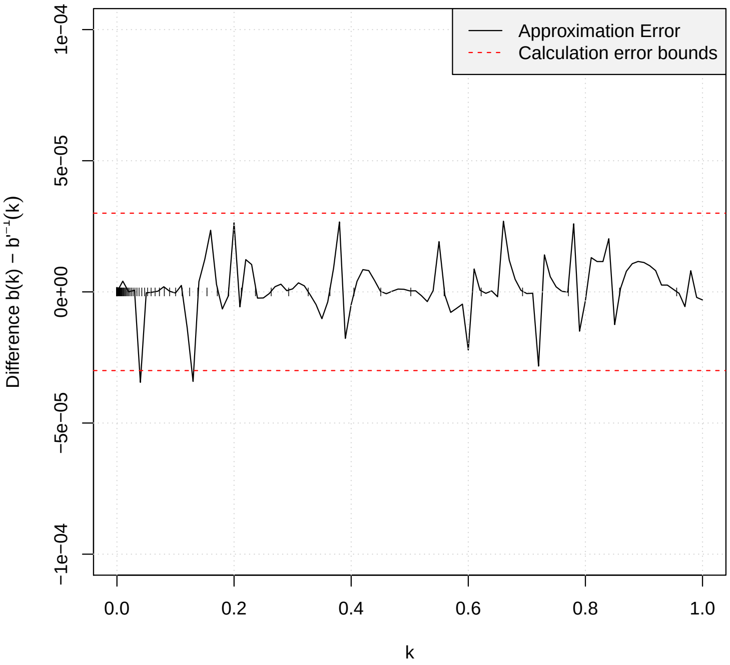
We approximate some functions to speed computation. To do so, we calculate a number of points – which is computationally expensive – and store those values for interpolation in the future, which is very fast. Interpolations are as accurate as the numerical approximations themselves.

For example, we approximate $b'^{-1}(x, z)$ from Equation (4.4) in the common case where b does not vary with x. We sample $b'^{-1}(z)$ over 250 z-values in its domain $[0, b'(0)]$. We then use a cubic spline interpolation of these points as our approximation b'^{-1}_{approx}. The difference between b'^{-1} and b'^{-1}_{approx} lies within the accuracy of the numeric functions used for inverses and derivatives; see Figure 4.2. This reduces the computation time for this function u, which can be evaluated more than 30,000 times in a single optimization at the level of detail we specified, by a factor of 650. We also implement approximations for ζ, f, and δ using similar methods and with similar precision and reduction in computation time.

We show search effort allocations for example searches using simulated data in Figures 4.3 and 4.4.

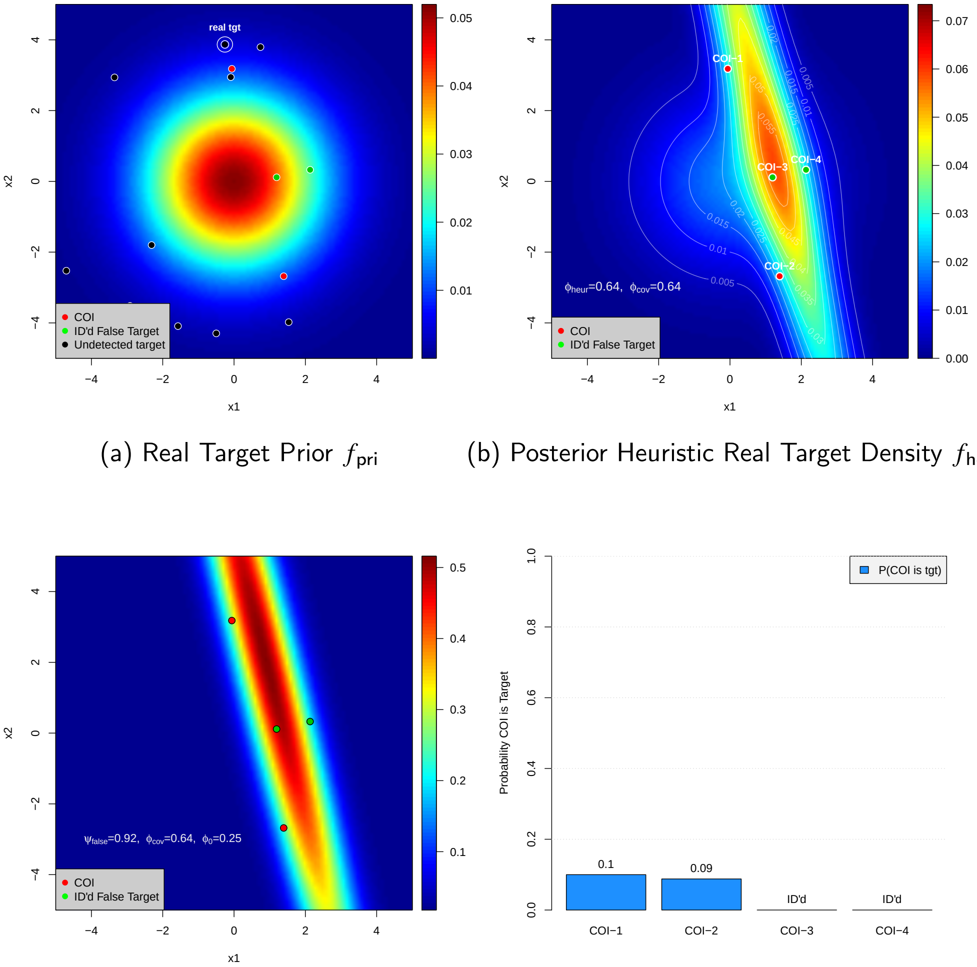
4.8.2 Semi-Adaptive Plans

The optimization discussed in this chapter generates a static, non-adaptive, search plan Φ. It allocates conditional contact investigation effort for contacts that may be detected, but it is non-adaptive. To make it a semi-adaptive plan, we need to apply it, execute some amount of search, and then re-evaluate our plan. We discuss this in Chapter 5.



The error in the approximation of b'^{-1} is smaller than the error of the numerical functions defining b'^{-1} itself, on the order of 3×10^{-5}.

Figure 4.2. Numerical Approximation Error of b'^{-1}



(a) Real Target Prior f_{pri}

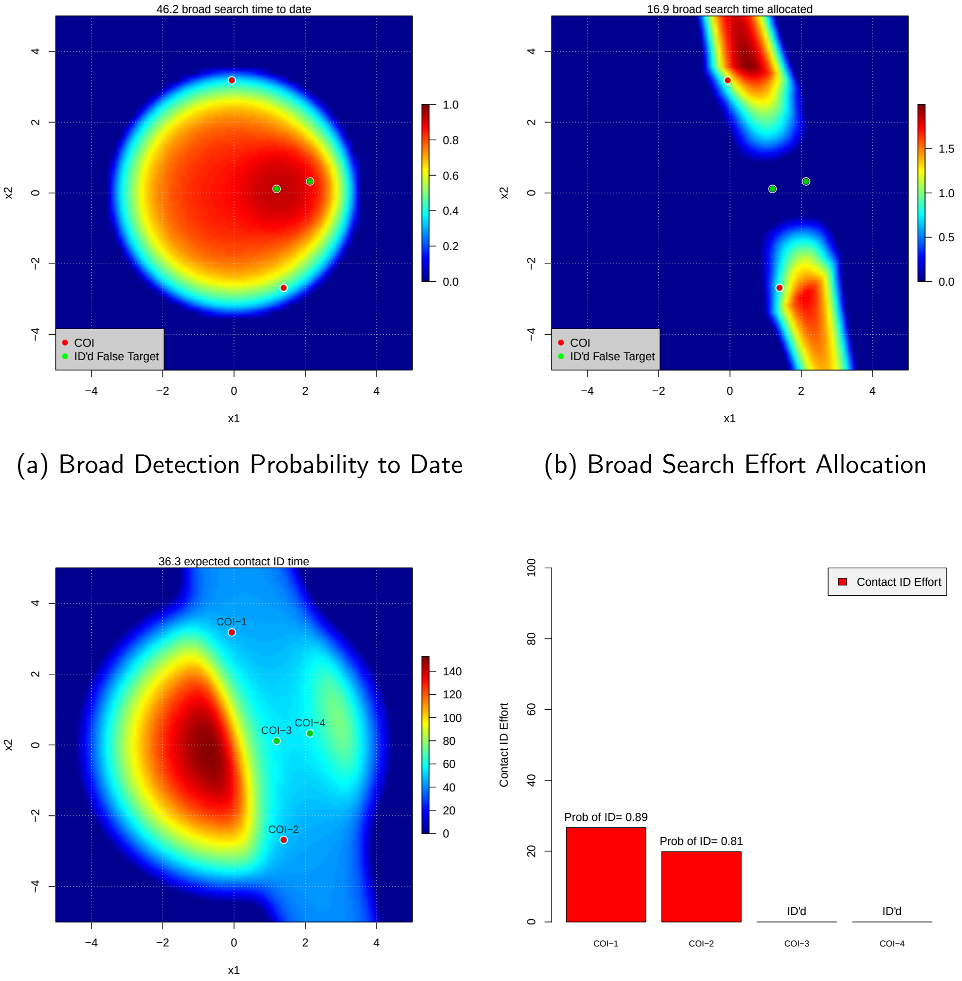
(b) Posterior Heuristic Real Target Density f_{heur}

(c) Posterior Heuristic False Target Density δ_{heur}

(d) Posterior COI Probabilities $\tilde{\pi}_i$

Results of the optimal allocation of search effort with simulated data. Here the searcher has already completed three sorties of search, detected four contacts, and identified two as false targets.

Figure 4.3. Example of Search Optimization



(a) Broad Detection Probability to Date

(b) Broad Search Effort Allocation

(c) Contingent ID Effort

(d) Allocation of Contact ID Effort

Results of the optimal allocation of the next 100 units of search time. The searcher splits effort among broad area search, contingent identification of any new COIs detected, and attempting to identify the two COIs.

Figure 4.4. Example of Search Optimization (continued)

CHAPTER 5: Simulation

5.1 Semi-Adaptive Search

We use some additional terms to describe the more complex searches studied in this chapter.

A *sortie* is a single asset applying search effort in a discrete time, usually with limited ability to dynamically re-task. For example, a UUV conducts a sortie as a single mission from launch to recovery from its host platform, and typically will not update its programmed navigation and sensor tasking.

A search *operation* is a sequence of one or more sorties searching for the same target.

In Section 5 of [9], Stone and Stanshine show that in general, there is no guarantee that a search plan is uniformly optimal – that is, that a search plan optimal in total time $t_2 > t_1$ is an extension of the optimal plan for time t_1. This presents a difficulty in our simulation given the sortie construct; if an operation is planned as a sequence of optimal sorties, the sum of the sorties may not be optimal for the operation as a whole. We plan for the total search time remaining in the operation, and plan each sortie to execute the appropriate fraction of the search plan that is optimal for the operation. As with any optimal allocation of effort, this presents difficulties in practice as discussed in Section 1.2, but it demonstrates the potential advantages of this method.

Stone discusses that an OSAP can be updated continuously – with the interval between updates approaching zero – to make it have the greatest performance gains [16]. We do not model such a continuous OSAP, but use a discrete semi-adaptive plan with discrete updates on fixed time intervals. In practice, many searches are performed in sorties. One or more search assets will be launched for a sortie of a specific duration. All data will be downloaded upon recovery and analyzed. The sortie provides a natural point at which to re-set the search. In high-risk environments relevant for military searches but also many rescue and recovery efforts, it is advantageous for the search asset to follow a predictable path so that other units will know where it is at any given time. It would be difficult to track the searcher if it were dynamically re-tasking itself during a search sortie. This also simplifies our computations considerably, as we would have to update not just the posterior and a new search plan, but re-calculate estimates for real and false target densities with each update.

5.2 Model Comparison

We compare search plans using estimates as developed in Chapter 3 to other methods from the literature. We simulate the following search plans:

- ONAP. This uses the model of Chapter 4 but only conducts a single sortie for the entire search time available. Executing a single sortie, it does not update allocations of search effort, making it a non-adaptive plan.
- ONAP with ICCI. Rather than permitting inconclusive contact investigation with breathers, this non-adaptive model – of a single sortie – requires all contact investigations to be immediate and conclusive. As discussed in section 4.4.2, this is equivalent to setting the contingent contact ID effort allocation $\omega = \infty$.
- OSAP. This uses the model of Chapter 4 and conducts multiple sorties. With each sortie, it updates its search plan as in Chapter 4, but does not estimate the target density as in Chapters 2 and 3.
- OSAP with ICCI. This is semi-adaptive search requiring immediate and conclusive contact investigation, rather than permitting inconclusive investigation and breathers.
- Optimal Non-Adaptive Plan with Estimates (ONAP-E). This uses the model of Chapter 4 and conducts a single sortie. It starts by updating the target distributions f and δ as described in Chapters 2 and 3, and then calculates a search plan as in Chapter 4. In order to calculate an estimate, an ONAP-E can only be used for a search already in progress.
- Optimal Semi-Adaptive Plan with Estimates (OSAP-E). This uses the model of Chapter 4 and conduct multiple sorties. With each sortie, it will first update the target distributions f and δ as described in Chapters 2 and 3, and then update its search plan as in Chapter 4. This model is the focus and primary contribution of this study.
- OSAP-E with ICCI (OSAP-ICCI-E). We use this method to test whether our density estimation provides an advantage when ICCI is used.
- Optimal Semi-Adaptive Plan with Accepted Estimates (OSAP-AE). This plan dis-

cards the prior density and intensity of real and false targets once it is able to estimate from data, and fully accepts the estimates to use in search optimization. It does not use the heuristic from Section 3.5.2.

5.3 Estimating Search Success

We first use simulation to evaluate our non-adaptive model from Chapter 4, to confirm the formulas for expected search outcomes of Section 4.7. To do so, we randomly generate 2000 search regions using an NHPP constructed as in Equation (2.4). We then simulate the application of broad search effort uniformly over R sufficient to detect 50% of all targets. This represents the state of a search in the middle of an operation. We compare the model's estimated probability of search success to simulated results of running it in terms of probability of success, and plot how the estimate compares to the simulated results. We use both our ONAP-E model which first builds an updated estimate of the real and false target densities and then finds the optimal allocation of search effort, and an ONAP search that calculates the allocation of effort based on the prior real and false target densities.[3] As Figure 5.1 shows, these estimates for the probability of success are quite accurate, and estimating the intensities helps improve them substantially.

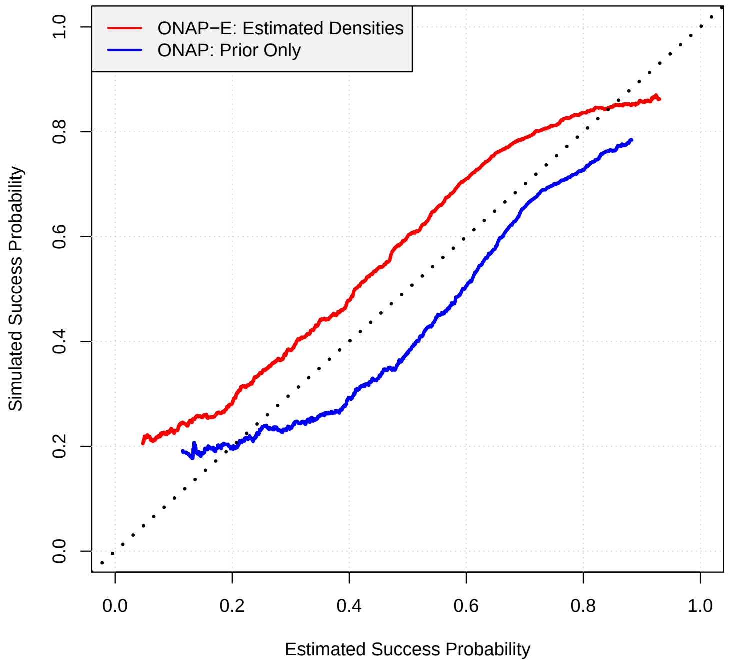
5.4 Simulation Study

We next use simulation to evaluate the performance of the search models from Section 5.2 under various conditions. Within the scope of this study, we do not execute a full experimental design to attempt to build a response surface for the performance of these model under various combinations of factors, with a focus on OSAP-E. We do, however, make some preliminary investigations to show the relative performance of this approach and its potential for improved results in terms of search success when searching in the presence of false targets. This adds significantly to the simulation studies of false target optimal search in the literature. Finally, we identify some strengths and weaknesses that may illuminate opportunities for future investigation. Our specific goals are to:

- Investigate the effect of the number of sorties in a search operation

[3] Note that we do not have a means to directly calculate the probability of success for a semi-adaptive search, as semi-adaptive searches are built as a sequence of non-adaptive searches.

- Investigate the effect of the false target intensity on background intensity and feature intensity
- Investigate the effect of the time to identify targets
- Investigate the effect of time available in the operation.



We compare simulation results to predicted search success from our optimization over 2000 trials. Both closely track the a diagonal line of slope 1.0 that represents a match between estimated and simulated probability of success. The ONAP-E model that first calculates the MLE estimates for real and false target intensities, provides more conservative estimates of the probability of success, in contrast to the ONAP model that does not estimate the densities. ONAP is below the line because its prior real and false target distributions, which are centered at the origin, are not identical to the actual ones, which have an elliptical "feature" centered elsewhere in any particular simulated search.

Figure 5.1. Comparison of Predicted vs Simulated Success Rate

5.4.1 Methodology

We conduct ten replications for each randomly generated search region with its realized distribution of false targets and the real target location. We generate a sufficient number of regions in order to achieve significance in our key results. We calculate confidence intervals for success probability in the normal way using the t-distribution and an $\alpha = 0.95$ confidence level.

We use the factor settings in Table 5.1 as our baseline:

Table 5.1. Baseline Factor Settings for Simulation

Factor	Baseline Setting
Number of Sorties (n)	16
Background Intensity (ρ)	0.10 ($\sim$ 10 false targets in R)
Feature Intensity (D)	4.0 ($\sim$ 7 false targets in feature)
Identification Time - real target ($E[a]$)	12
Identification Time - false target ($E[A]$)	16
Total Time Available in Operation	200

We report the probability of success for a given search, rather than mean time to success. This is because the mean time to success assumes that a search will last as long as it needs to be successful. A planner – and decision maker – is interested in how likely a search is to be successful given the time available; or how much time must be planned for a search operation to have an acceptable probability of success. Further, in the sortie construct of real-world search operations, the time of success is dependent on the path chosen and sortie-level details that would require a level of resolution greater than considered in this study. A success curve similar to that in Figure 4.1 can be calculated at any point in the search, and will provide information on the relationship between success probabilities and time that planners need to consider.

5.4.2 Implementation Details

We implement these simulations in R, using code developed for the methods of Chapters 2 through 4. We include the ability to track the status of a search over time and multiple sorties. Typical run times for a simulation of a single operation of 16 sorties were on the order of 15 minutes if the search was unsuccessful. More computation time is required for the search optimization of Chapter 4 than the intensity estimation of Chapter 2. Parallel processing on 36-core Dell Precision 7910 workstations enabled a sufficient number of replications for this study.

We ensure that simulated searches do not exceed the total time allocated. This can lead to some issues if the estimate of false target density is off. For example, if we plan for a 30 hour search and we need to terminate the search at 30 hours of actual searching, we may have planned an expected value of 10 hours of contingent contact identification. If the false targets are denser than expected, we could need to spend 20 hours identifying them, but we have to end the search at 30 total hours after only half of the false targets have been identified per the plan. In this case we assume we search the false targets in order, from highest to lowest probability, until the target is identified identified, allocated time is used up, or the sortie time expires. Alternatively, we may have fewer false targets than expected and have leftover search time, that is unused for lack of targets to identify. We recycle this time, making the next sortie slightly longer by the amount of time not used in the previous sortie. This is to ensure fair comparisons, but is not far from operational practice.

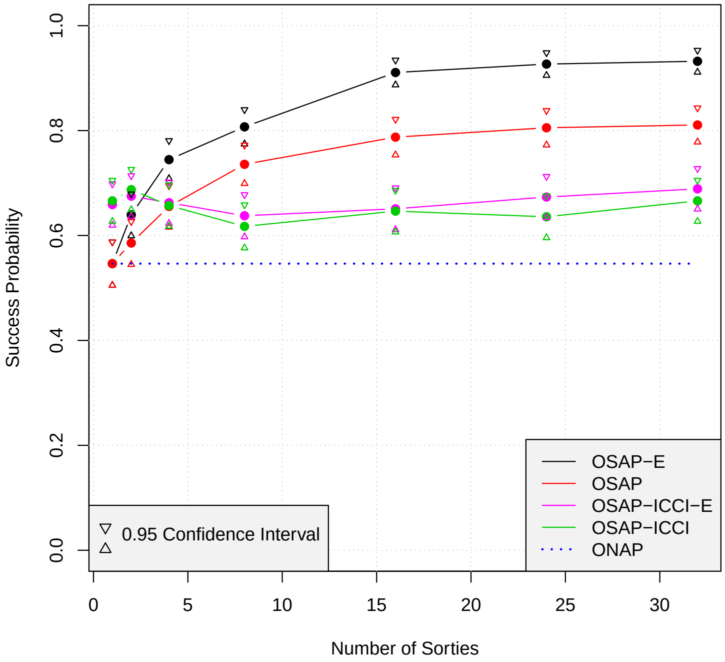
5.5 Results

We report the results of simulation experiments using over 80,000 simulated searches. A table of detailed results is included in Appendix C.

5.5.1 Number of Sorties

We vary the number of sorties in a search operation to change the number of times a semi-adaptive search updates and recalculates allocations of effort and estimates of intensity. In Figure 5.2, we show that OSAP-E significantly outperforms both OSAP and ICCI when it divides an operation in eight or more sorties (and thus eight or more discrete updates). It appears that 16 sorties are a sufficient number of updates: we do not see a significant change in outcomes as we increase the number of sorties for a fixed total search time.

We can see that the ICCI and the Immediate and Conclusive Contact Identification with Estimates (ICCI-E) plans appear to have indistinguishable results. They outperform the ONAP methods that allow inconclusive contact investigation when there is only one sortie, representing a non-adaptive plan. However, as the plans with split allocation execute more sorties, becoming true semi-adaptive plans, they rapidly outperform the ICCI plans. This is expected, as this search meets the conditions (4.1) for split allocation to be optimal as in [10].

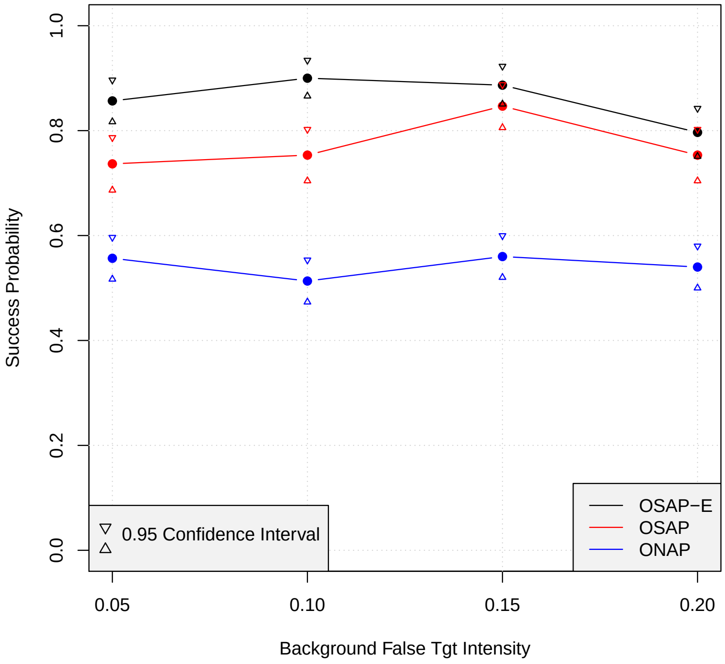


This plots the performance of our models from Section 5.2 by number of sorties. We can see the advantage of attempting to update the prior probability distributions as the number of sorties increases, increasing the opportunity to take advantage of this additional information. There are minimal returns beyond 16 sorties, so we use 16 sorties in semi-adaptive searches for the remainder of this study. Note that the ONAP curve is plotted for comparison, but only executes a single sortie.

Figure 5.2. Comparison of Search Methods by Number of Sorties

5.5.2 Background Intensity

We simulate various levels of background intensity ρ and plot the results in Figure 5.3. Over a total search area of 100 notional units, the range of this graph is from an average of 5 to an average of 20 false targets in the background clutter, compared to an average of 7.5 false targets in the feature. Again we see that OSAP-E significantly outperforms OSAP, showing the potential benefit of our intensity-estimating approach. While we would expect that more background clutter would make the search more difficult, this does not appear to always be the case. OSAP does not decline with ρ over this relatively broad range. This appears to be an inherent advantage of semi-adaptive search plans under these conditions.

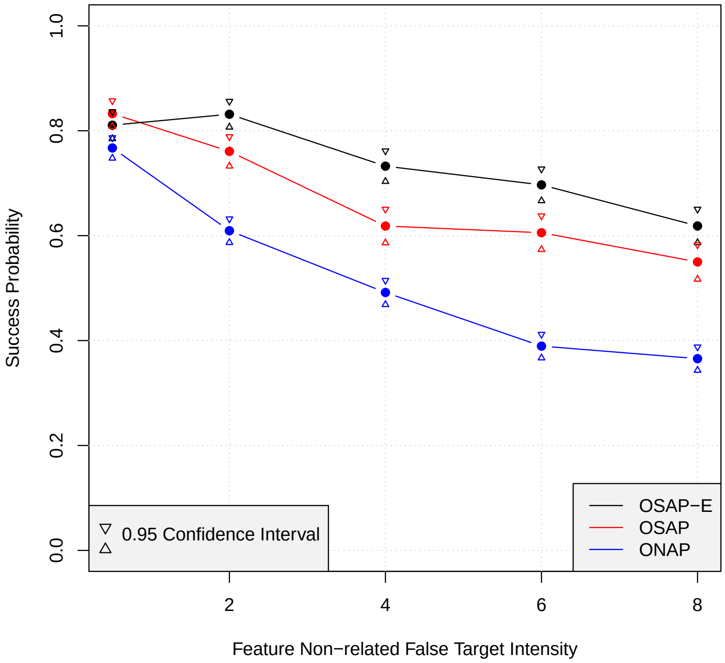


Success probability is relatively constant with the background level of false target clutter. Unexpectedly, both semi-adaptive methods may perform better with more background targets.

Figure 5.3. Effect of Background False Target Intensity ρ on Success Rate

5.5.3 Feature Intensity

In Figure 5.4, we plot the effect on success rate of varying feature intensity D. We see that OSAP-E may perform worse than OSAP when the feature is weak – at the left end of the graph, less than 1 false target in the feature on average. It is possible the MLE estimation is finding features that do not exist and actually harming the search by diverting effort, rather than adding value by improving the intensity estimates. However, this difference is not significant, $p = 0.22$ in a two-sided t-test of their simulated success probabilities at $D = 0.5$. For any more intense features with at least 3.5 expected false targets ($D \geq 2$), OSAP-E significantly outperforms OSAP by taking advantage of the additional information in the MLE estimates. As the expected number of feature false targets grows large, the two methods may begin to converge. It is possible they become time-limited by the number of false targets that must be investigated.

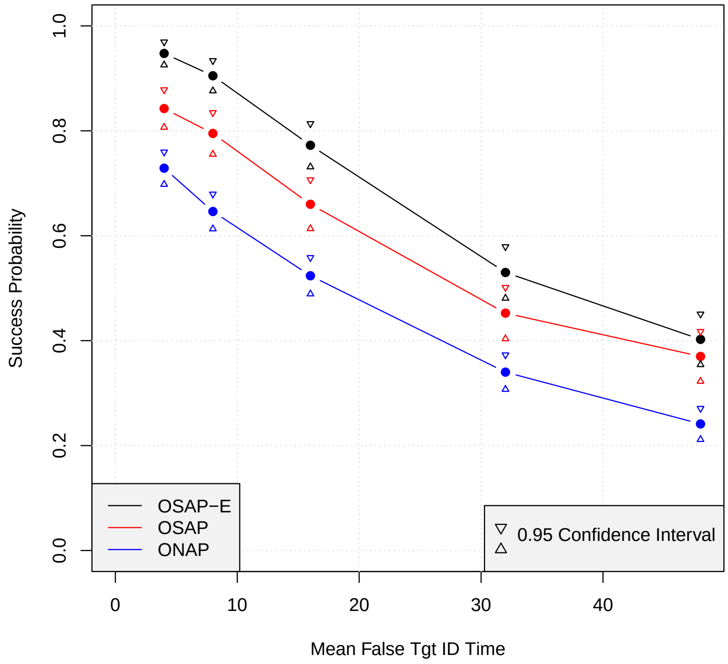


Here we show how effectiveness varies when the peak feature intensity changes from low to high. OSAP-E appears to benefit from at least some false targets in the feature, allowing it to better estimate the location of the target.

Figure 5.4. Effect of Feature Peak Intensity D on Success Rate

5.5.4 Identification Time

We show performance of OSAP and OSAP-E over a range of expected contact identification times in Figure 5.5. OSAP-E appears to consistently outperforms OSAP over the entire range considered, but this advantage is not statistically significant. Average simulation success probabilities decline steadily with increased time to identify false targets for all methods. Unlike Stone in [16], we do not see a significantly greater advantage to the semi-adaptive methods as identification time increases.



Success rate declines as contact identification time increases for all methods.

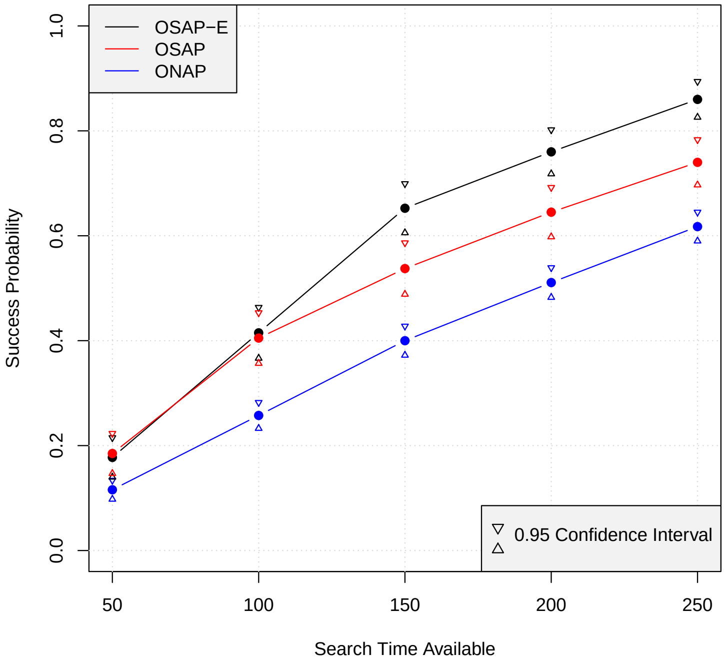
Figure 5.5. Effect of Contact Identification Time $E[A]$ on Average Simulation Success Probability

5.5.5 Total Operation Time

We anticipate that for low search times, OSAP-E might not have much if any advantage over OSAP. This is because of the thresholds used in the heuristic, which require a significant fraction of search to be completed before accepting the estimates for use in the search optimization over the priors. Figure 5.6 confirms that the OSAP-E does converge to the OSAP for relatively low total search times. As the heuristic is written - in terms of the coverage fraction of the search region - it is dependent on the size of the search region, not just the prior. This might depend too heavily on the region rather than the prior, suggesting

$\phi^{(1)}_{\text{cov}}$ as one potential improvement for future experimentation:

$$\phi^{(1)}_{\text{cov}} = \int_R f_{\text{est}}(\boldsymbol{x}) b(\boldsymbol{x}, Z(\boldsymbol{x}))\, \mathrm{d}\boldsymbol{x}. \tag{5.1}$$

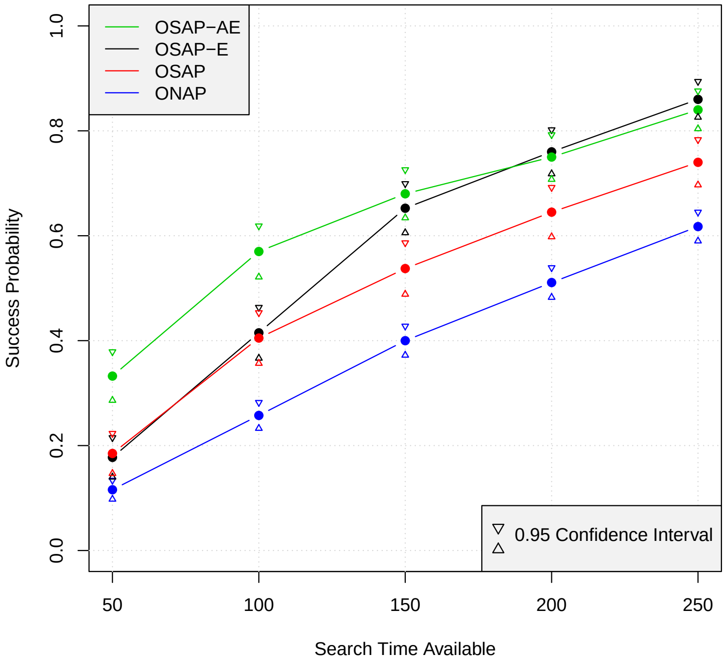


Success rates for OSAP and ICCI methods scale linearly, but OSAP-E does better past a threshold and begins to outperform the other methods. This may be related to thresholds programmed into the heuristic.

Figure 5.6. Effect of Search Time Available t on Success Rate

5.5.6 Heuristic

We omit a detailed study of the heuristic. Its intention is to ensure we retain an appropriate weighting of the prior distribution for the false target and do not accept a weak estimate built on a small sample of data. In this case, however, it would be better to discard all prior data and completely accept the estimate for optimizing search effort. In Figure 5.7 we show that accepting the estimate allows the OSAP-AE to retain a significant advantage over OSAP at low operational search times and chances of success. Other target distributions – such as weak features, or when the prior does not exactly match the distribution used to generate the targets in R – could reveal some advantage to the heuristic's inherent conservatism.



If we simply accept the estimate of false target intensity, rather than scaling it with the prior as in the heuristic, OSAP-AE outperforms OSAP-E at low total search times. This suggests further work evaluating the heuristic and identifying situations when it performs well.

Figure 5.7. Comparison of Accepting Estimates with Heuristic Blending

CHAPTER 6: Conclusion

6.1 Significance of Results

This study demonstrates previous theoretical work on non-adaptive search plans, and those with immediate and conclusive contact investigation, in a flexible simulation study over two-dimensional space. We are able to compare empirical success rates for false-target searches and the effectiveness of semi-adaptive search plans in a more realistic simulation than the small number of boxes used in past work. Additionally, in certain situations such as those studied here, the use of estimates of target density is shown to improve search performance: it is possible to take advantage of information on the relative locations of targets to improve the probability of search success.

6.2 Operational Utility

The OSAP-E search planning method may not be different from what an experienced analyst could do with appropriate time and tools available. However, there is an advantage to automating the process for consistency and as a supplement for experience. It is also able to minimize the opportunity for human biases and errors, including confirmation bias and over-optimism, enabling impartial detection of subtle features. A tool using these techniques – implemented on a laptop computer and deployable to the site of any search operation worldwide – could be a valuable addition for an experienced analyst, just as a weather model assists a knowledgeable weatherman in making better predictions. Additionally, an automated tool for search planning based on these technique could be applied by autonomous vehicles with long mission durations, such as mine hunting Unmanned Surface Vehicles (USVs) and UUVs.

The ability to employ the tools developed by Stone and others to estimate the probability of success given remaining time available as in Figure 4.1– based on all data collected up to the point of decision – can have use in risk assessments and commander's decisions on whether and for how long to proceed with a search operation.

Not all searches will meet the assumptions used here to improve results with estimation of intensity. In these cases – when there are not expected to be any false targets related to the real target – semi-adaptive search without estimated intensity is a good option that outperforms non-adaptive search. Improved heuristics may point towards a single solution that can balance these cases better and not require a selection of method ahead of time.

Further, the case where the time to identification for real targets may be longer than for false targets deserves further study. Although we did not see any clear advantage to the use of estimates in ICCI searches, there may be some, particularly in the domain of (4.2) from [10] when ICCI searches are optimal over split effort searches. If not, that may provide insight into the strengths of this method.

6.2.1 Success Criteria

This study focuses on finding the actual black box itself as the goal of the search. It did not consider any value to finding a related false target that might be a piece of debris from the aircraft. However, in an actual search, often the most difficult task is to find the site of the wreckage itself. Any follow-on investigation of the wreckage site will take time but be certain to produce results, and will be fully resourced as needed. Therefore, another search objective could be to identify one or more related false targets as sufficient to locate the crash site for further recovery efforts.

6.3 Further Study

The heuristic of Section 3.5.2 can be better studied and improved. We are not able to make recommendations on blending of prior information with the estimates developed from search results, as we have not fully explored the region where accepting the estimates and discarding the prior is preferred.

Often there may be two or more scenarios developed for the target location, and each could have different characteristics that suggested different search optimization methods would have the best performance. The heuristic in Equation (3.18) may protect against overfitting in some possible scenarios, when the prior is in error and there is no detectable feature, but prevent realization of the full potential of intensity estimates in others, when there is a strong feature. An example could be if it is unknown whether an aircraft exploded in flight to create a debris field, or crash-landed intact and can be found intact on the sea floor. The selection of a method then would be a decision problem for the searcher before beginning operations.

This method assumes that the prior contains a 100% coverage of the target. That is, the searcher is certain that the target is in this search region. That is not always realistic. Despite this, the searcher can use this method as is, and the searcher will have to decide – based on the probability of success this method can provide, and predictions of how long it might take to improve that success – when to stop searching and switch to another region. See [8] Section 2.4 for a discussion of defective distributions.

Alternately, this could be seen as an incomplete prior distribution. This method could be used with a prior that is a larger search region containing two or more modes of probability, each representing a scenario, all appropriately normalized based on prior beliefs to sum to 100% total probability.

There is room for more simulation experimentation and improvement. More simulation results would help better understand the applicability of this model and its robustness to a variety of possible scenarios – when it is better than other methods, and when it breaks down and fails to improve results.

There may be an opportunity for a “learning” information-based search algorithm that can plan ahead to search regions that it has not yet searched in order to confirm or deny the presence of false targets. This would allow it to seek out more information to improve its estimates before spending time on contact investigation - which could be very expensive. This could lead towards an optimal adaptive search plan.

Finally, we did not include real-world data, but it would be informative to study real-world searches using this method.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A:
Bootstrapped Estimates of Uncertainty

A.1 Problem Statement

In this appendix, we use the bootstrap to estimate uncertainty in the values of $\lambda(\mathbf{x})$ through the parameters defining the intensity function. We compare several different non-parametric and parametric bootstrap methods using different types of intervals. Finally, we attempt to estimate the uncertainty in the intensity $\lambda(\mathbf{x})$ directly without resort to the bootstrap, to see if we can save a significant amount of computation time.

A.1.1 Contact Intensity Function

We expect the intensity of search contacts $\lambda(\mathbf{x})$ to be an elliptical feature on top of a constant background level. We model it with the following equation:

$$\xi(\boldsymbol{x}) = \left(\frac{(x_1 - C_1)\cos\alpha + (x_2 - C_2)\sin\alpha}{\eta_1}\right)^{2\beta} + \left(\frac{(x_1 - C_1)\sin\alpha - (x_2 - C_2)\cos\alpha}{\eta_2}\right)^{2\beta}$$

$$\lambda(\mathbf{x}) = B + \sum_{j=1}^{\kappa} De^{-\xi(\mathbf{x})} \tag{A.1}$$

where

C_1 = x_1 coordinate of center of ellipse

C_1 = x_2 coordinate of center of ellipse

η_1 = length of ellipse

η_2 = width of ellipse

B = background density that is constant over the entire region

D = peak intensity of ellipse - additive with background intensity

α = angle of rotation of the ellipse

β = steepness of edges of ellipse feature. We fix $\beta = 1$ in this study.

$\theta = (C_1, C_2, \eta_1, \eta_2, B, D, A)^T$

We consider a square search region A defined over $x_1 \in [-5, 5], x_2 \in [-5, 5]$.

A.1.2 Estimate of Intensity

We use the MLE intensity estimates from Chapter 2 in this Appendix. For this study, we use one source distribution: $\mathbf{X} \sim \text{Pois}(\lambda(\boldsymbol{x}, \boldsymbol{\theta}_{\text{SRC}}))$, where $\boldsymbol{\theta}_{\text{SRC}} = (C_1 = 2.0, C_2 = -2.0, \eta_1 = 2.5, \eta_2 = 1.25, B = 0.1, D = 3.5, A = 1.0, \beta = 1.0))$. This source distribution is plotted in Figure 2.4.

As noted in Chapter 2, the optimization does not provide any measure of uncertainty in this $\boldsymbol{\theta}$. An estimate of the uncertainty is important, for example, in attempting to optimize allocation of search effort to identify a given target.

Our estimate of uncertainty has two sources of error - one from the estimate itself, and one of sampling error from the (single) observation of the source distribution we have to work with.

We can sample from this distribution 1000 times, and calculate the MLE optimal $\hat{\boldsymbol{\theta}}$ for each. We plot the results in histograms in Figure A.1, including the resulting log-likelihood as well. We can see that there may be some bias in our estimator for some of the parameters; in particular it has difficulty capturing the correct dimensions of the ellipse η_1 and η_2, and the rotation angle of the ellipse A. These are not independent biases but result from the structure of our intensity function and $\boldsymbol{\theta}_{\text{SRC}}$.

We can see the samples mostly have error that appears normal. However, this is error we cannot reduce or eliminate by choice of an estimation method - it is a lower bound on the error in our estimation.

A.2 Approaches

The literature contains some discussion of bootstrapping to estimate the intensity of a non-homogeneous spatial process. Snethlage evaluates bootstrapping a spatial Poisson process to determine statistics describing the point process [20]. Section 2 of [20] discusses a traditional bootstrap of the data points observed. One drawback pointed out is that the resulting bootstrap samples - n points sampled with replacement from the n observed data points - will have some points repeated multiple times, which is not the case in the original data and could lead to differences in behavior of the estimator. Section 3 of [20] discusses another bootstrap sampling method, using $n^* \sim \text{Poisson}(n)$ points. In Section 3, [20] recommends use of confidence regions for the Poisson parameter $\lambda(\mathbf{x})$, rather than bootstrapping. Patil and Kulkarni compare 19 alternate ways to generate such intervals for the Poisson parameter, and recommend applicability in different situations [22]. Cowling et al. use these two bootstrap sampling methods as well as a parametric method, with a kernel estimator for the intensity $\lambda(\boldsymbol{x})$ [23] .

Our comparative approach over three bootstrap sampling methods is similar to that in [23], but over two dimensions for a spatial process rather than one-dimensional time series data on coal mining accidents. See also Barker [24].

A.3 Methods for Estimating Uncertainty

We compare three ways of generating bootstrap samples, two types of bootstrap-derived confidence intervals used to estimate uncertainty, and three confidence intervals for the intensity that rely on simple calculations. Our goal is to find the best method to use for estimates of uncertainty, that is the most accurate, useful, and computationally efficient.

A.3.1 Bootstrap Methods

We study the following three bootstrap sampling methods:

1. Constant-n. This is a non-parametric bootstrap with a constant number of points. Our bootstrap samples are exactly n points chosen from $\mathbf{X}$, where $\mathbf{X} = \{\mathbf{x^1}, \mathbf{x^2}, \ldots, \mathbf{x^n})\}$.
2. Poisson-n. This is a non-parametric bootstrap with a Poisson number of points. We sample with replacement $n \sim \text{Pois}(n)$ points from $\mathbf{X}$.

3. Parametric. We draw a sample of points from the non-homogeneous spatial Poisson process defined by the estimated intensity function $\lambda(\mathbf{x}, \hat{\boldsymbol{\theta}})$ as its intensity function.

We use the following two bootstrap intervals to quantify the uncertainty in the Poisson intensity over space:

1. Standard error estimate. [25] Chapter 6 discusses use of the bootstrap and the assumption of normality to calculate the standard error of an estimator. The authors suggest that $B = 50$ is typically enough bootstrap samples to calculate the bootstrap standard error:

   $$100(1-\alpha)\%\text{CI} = \left(\hat{\boldsymbol{\theta}} + Z_{\alpha_1}\hat{\text{se}}_B, \hat{\boldsymbol{\theta}} + Z_{\alpha_2}\hat{\text{se}}_B\right), \tag{A.2}$$

   where

   $$\begin{aligned} Z_\alpha &= \text{normal critical value} \\ \alpha_1 &= (1-\alpha)/2; \text{ lower bound of interval for confidence level } \alpha \\ \alpha_2 &= 1-(1-\alpha)/2; \text{ upper bound of interval for confidence level } \alpha \\ \hat{\boldsymbol{\theta}} &= \text{Estimate of } \boldsymbol{\theta} \text{ from data observation} \\ \hat{\text{se}}_B &= \text{Bootstrap standard error} \\ &= \text{sd}(\hat{\boldsymbol{\theta}}^*). \end{aligned}$$

2. Bootstrap percent interval. [25] Chapter 13 discusses an alternative, bootstrap percentile intervals. These do not rely on an assumption of normality but require many more bootstrap samples. After [25], we use $B = 1000$ bootstrap samples. Although we have more computing power available, solving a non-convex optimization problem for each bootstrap limits our appetite for larger sample sizes, and would be (even more) impractical for applied use. From [25](13.3), the bootstrap percentile interval is

   $$100(1-\alpha)\%\text{CI} = \left(\hat{\boldsymbol{\theta}}_B^{(\alpha)}, \hat{\boldsymbol{\theta}}_B^{(1-\alpha)}\right) \tag{A.3}$$

   where

   $$\hat{\boldsymbol{\theta}}^{(\alpha)} = \text{the } (100 \cdot \alpha)\text{th empirical percentile of } \hat{\boldsymbol{\theta}}$$

We do not experiment with Approximate Bootstrap Confidence (ABC) or Bias-Corrected and Accelerated (BCA) interval methods, or other intervals from [23].

A.3.2 Non-bootstrap methods

Non-bootstrap methods generate confidence intervals for $\lambda(\mathbf{x})$ based only on the value of $\lambda(\mathbf{x})$ itself. Thus, they rely on only a single solve of the MLE optimization problem of Equation (2.2) to find the optimal value of $\lambda(\mathbf{x})$ suggested by the data $\boldsymbol{X}$. Compared to bootstrap methods, which require B solves of the MLE (2.2), this is much faster to compute, as recommended by [20].

Confidence interval for intensity $\lambda(\mathbf{x})$ methods:

1. Modified Wald interval λ. Instead of a bootstrap, we use the bounds on $\lambda(\mathbf{x})$ given by the modified Wald interval. Patil and Kulkarni recommend the Modified Wald interval for low anticipated values of the Poisson mean, such as we expect in our search [22]. We use the following, using 0 as a floor in place of any negative values:

$$100(1-\alpha)\%\text{CI} = \left(\lambda(\boldsymbol{x}) + Z_{\alpha_1}\sqrt{\lambda(\boldsymbol{x})}, \lambda(\boldsymbol{x}) + Z_{\alpha_2}\sqrt{\lambda(\boldsymbol{x})}\right). \tag{A.4}$$

2. Standard Poisson interval. This is given by finding the nearest integer value for the 5th and 95th percentiles using the inverse quantile function in R: `qpois(c(0.05, 0.95), lambda)`. Patil and Kulkarni observe that the Confidence Interval (CI) with endpoints rounded (outwards) to the nearest integer performed much better in term of accuracy with a minimal increase in length [22].
3. Exact Poisson interval. This is given by the formula for the exact Poisson interval from Garwood [26] and as listed in [22]. It is shown to always have at least the desired level of accuracy, but at some cost in increased width.

$$100(1-\alpha)\%\text{CI} = \left(\frac{\chi^2_{2\lambda(\boldsymbol{x}),\alpha_1}}{2}, \frac{\chi^2_{2(\lambda(\boldsymbol{x})+1),\alpha_2}}{2}\right)$$

A.3.3 Comparison

We evaluate the in terms of accuracy - whether the $100(1-\alpha)\%$ (here, 90%) intervals they provide actually cover 90% of the error; and efficiency, by the size or width of the 90% interval. We evaluate the three bootstrap sampling methods, using both bootstrap confidence intervals, a total of six methods.

We also compare the size of the confidence intervals for $\hat{\lambda}(\mathbf{x})$. We can use the three non-bootstrap methods, and also the six bootstrap methods - calculating $\hat{\lambda} = \lambda(\boldsymbol{x}, \hat{\boldsymbol{\theta}}_{\boldsymbol{B}})$ for each. We do this over a grid of points covering the entire search region.

Finally, we consider the computational efficiency of each method, in terms of how many bootstrap replications they require. Applications of search optimization include austere underwater environments where power, computer resources, and time are often limited, and optimization routines will need to use many estimates of the contact intensity and its uncertainty, so this is an equally important consideration.

A.4 Results

We use a single actual value of $\boldsymbol{\theta}_{\mathrm{SRC}}$ as a source distribution we draw samples from, $\lambda(\mathbf{x}) = \lambda(\mathbf{x}, \boldsymbol{\theta}_{\mathrm{SRC}})$. For each simulated observation, we draw one sample $\mathbf{X}$ and build confidence intervals using each of our sampling and interval methods, with up to 1000 bootstrap replications. We simulate 500 observations. We plot the first 50 ellipses - as proxies for $\hat{\boldsymbol{\theta}}_{\boldsymbol{B}}$ - in Figure A.2.

A.4.1 Parameter Values

Figure A.3 shows our results for values of $\hat{\boldsymbol{\theta}}$. We can see that Constant-n sampling (a) is quite different from Poisson-n (b) and Parametric (c) sampling and gives markedly different results. Poisson-n and Parametric sampling appear almost equivalent, with the blue and green matching each other very closely both in Figure A.2 and in Figure A.3. The green and blue give tighter groups. However, the red - fixed N bootstrap - seems to better capture the source distribution, and its estimates fall better within the bounds of the observed distribution - the black lines in Figure A.3.

Of note, the Poisson-n and Parametric bootstrap sampling methods are quite inaccurate at estimating the location of the intensity feature (ellipse) - that is, values of C_1 and C_2. Both are pulling to the center of the search area, towards the origin at $(0, 0)$. Bootstrap samples of a fixed number of points do not seem to have this bias - the red points are nicely centered on the actual values of C_1 and C_2 in the panel (a) of Figure A.3. The Poisson-n and Parametric methods have the advantage of tighter groups for the other parameters.

A.4.2 Accuracy of Confidence Intervals

We next simulate a large number of samples to evaluate the coverage of the confidence intervals for $\boldsymbol{\theta}$ given by each method. We ran 500 simulations of observations drawn from the source distribution, of 1000 bootstrap samples for each sampling method, for percentile intervals. We ran 1000 simulations with 50 bootstrap samples for standard error intervals. Accuracy for the bootstrap methods is plotted in Figure A.4, compared to the parameters used in the source distribution. We also plot the accuracy of estimates for the intensity of the spatial NHPP over space in Figure A.5, compared to the intensity of the source distribution.

In these results, we see that the Constant-n bootstrap sampling seem to outperform the other two sampling techniques. The red bars achieve 90% accuracy over almost all parameters in $\boldsymbol{\theta}$. In contrast, the Poisson-n and Parametric sampled estimates fail to come close to the prescribed 90% almost half of the time.

Further, we can see that the standard error intervals are more accurate than the percentage intervals, in particular for Poisson and parametric sampling. As these plots are made with only 50 bootstrap samples - as recommended for standard error intervals, and well below recommendations for percentage intervals - this is not surprising. Further, we have seen in Figure A.1, our data is very close to normally distributed.

Plotted over space, we can see similar trends. Constant-n sampling outperforms other methods in accuracy, and standard error intervals outperform percentage. All models do well over the background noise along the left and top of the region; those sampling with random n have the most trouble in the center. This reflects how those methods tend to "pull" C_1 and C_2 towards the center of the region in Figures (A.2) and (A.3).

Also on Figure A.5, in the third row, we see the accuracy plotted for the Poisson intervals that do not use bootstrap sampling. In general these have much higher average accuracy.

Although constructed at the 90% confidence level, the modified Wald and exact intervals exceed 99% accuracy. However, the “standard” Poisson interval - the integer bounds above and below - performs very close to the 90% level.

A.4.3 Size of Confidence Intervals

We now can judge not just how accurate, but how useful these intervals are. The smaller the interval, the better. We plot the length of confidence intervals in Figure A.6 and over space in Figure A.7.

We can see that in general, the bootstrap confidence intervals are the smallest, with no clear advantage from the percentile intervals using 1000 bootstrap samples. The standard error intervals are slightly wider than the percentile intervals, but not significantly so for most applications. The standard (nearest-integer) intervals are in fact both longer and less accurate on average than the modified Wald intervals. The modified Wald and standard Poisson intervals are two times wider - and the exact Poisson interval was five times wider - than the bootstrap intervals. This is the cost of not performing any bootstrap sampling.

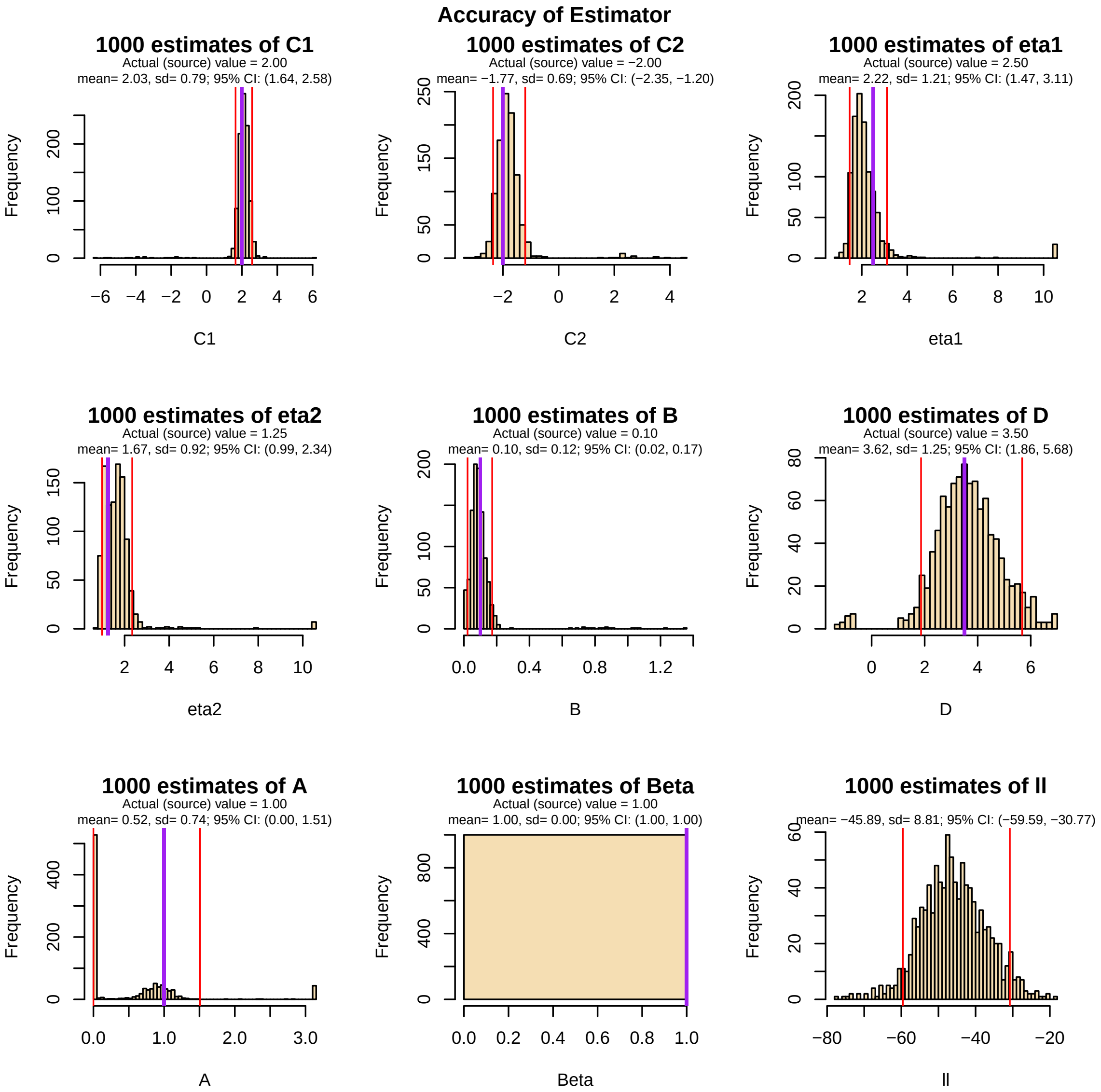
A.5 Implications

Based on these results, we can recommend the following:

1. Bootstrapping can help understand uncertainty in estimates, but it is not the only method. It is computationally intensive, although parallel processing can reduce the time required significantly if computing resources are available.
2. 50 bootstrap samples are sufficient for this problem, where the parameters have a normal distribution. The Standard Error intervals take advantage of this, and perform better, with fewer bootstraps required, than the percentile intervals.
3. Bootstrap intervals can achieve advertised accuracy. Bootstrap samples are best made non-parametrically using the same number of points that was observed to reduce bias (Constant-n). The other sampling methods, using a random number of points n, underperform significantly.
4. Without bootstrap, it is possible to generate very accurate intervals for intensity value at any given point. The Modified Wald intervals exceed the desired accuracy and are

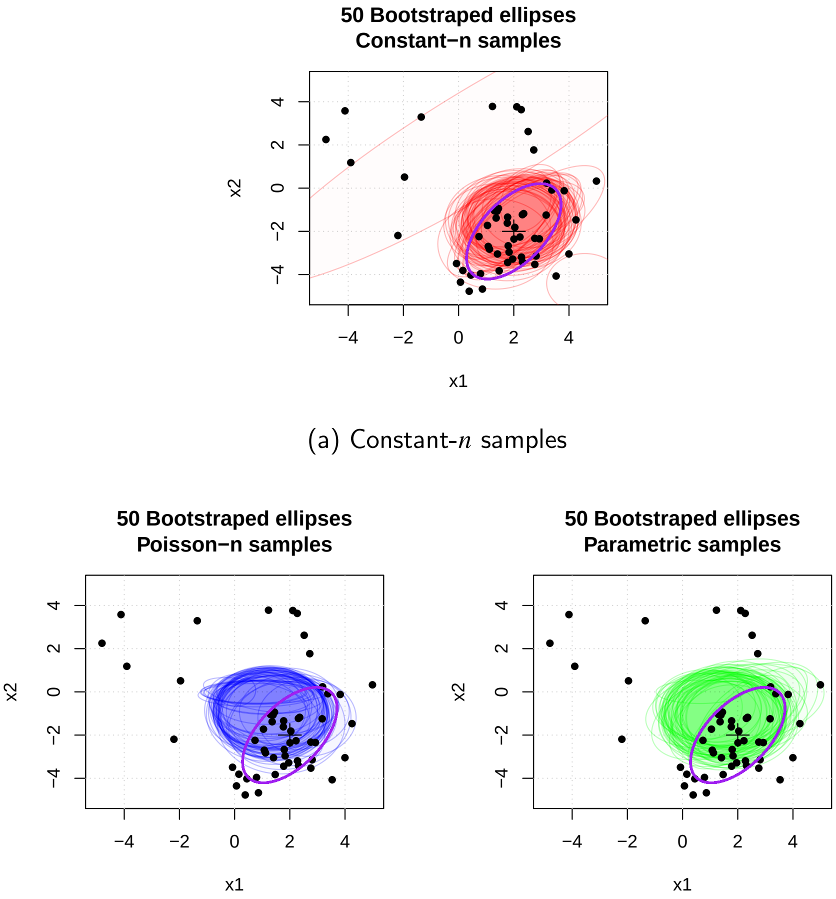
the most efficient, or smallest, of the non-bootstrap intervals. Although twice as large as a bootstrap interval, it is more accurate and much faster to generate.

5. Further study of more distributions, with different values of $\boldsymbol{\theta}_{\text{SRC}}$ would be helpful in understanding the limitations of both bootstrap and CI estimation techniques. One area for particular study is changing relative intensity of the feature and background noise, that is D and B, to see how that affects the MLE estimation algorithm and estimates of uncertainty. This would be a computationally intensive study.
6. Further study of a binomial sample from a single point pattern, rather than generating new point patterns from a distribution, would be of interest. Actual search results are sampled binomially from a single, fixed set of objects on the sea floor, which are either detected or not detected by a sensor. Any given search will either detect, or not detect, each object. This problem setup - instead of sampling from a continuous distribution of all possible debris patterns - also could affect future results in applications.



Histogram of $\hat{\boldsymbol{\theta}}$ over 1000 samples. Purple line is the actual parameter value from $\boldsymbol{\theta}_{\mathsf{SRC}}$.

Figure A.1. Maximum Likelihood Estimates of Parameters



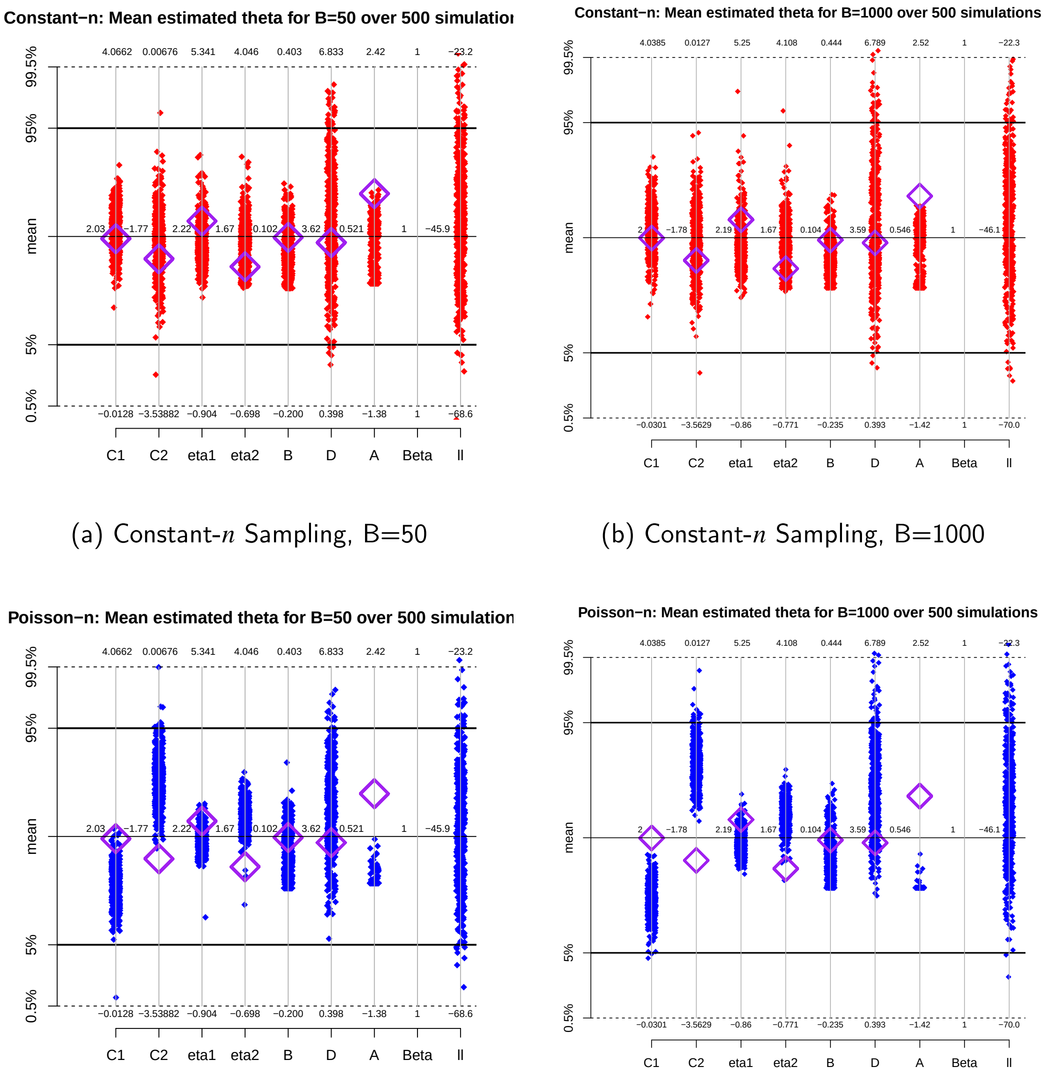
(a) Constant-n samples

(b) Poisson-n samples

(c) Parametric samples

Bootstrap ellipses of $\hat{\boldsymbol{\theta}}$ plotted for three methods of bootstrap, $B = 50$. All share a single observed set of points $\mathbf{X}$. Purple ellipse shows actual parameters of source distribution $\boldsymbol{\theta}_{\mathsf{SRC}}$.

Figure A.2. Ellipses for Three Bootstrap Methods



(a) Constant-n Sampling, B=50

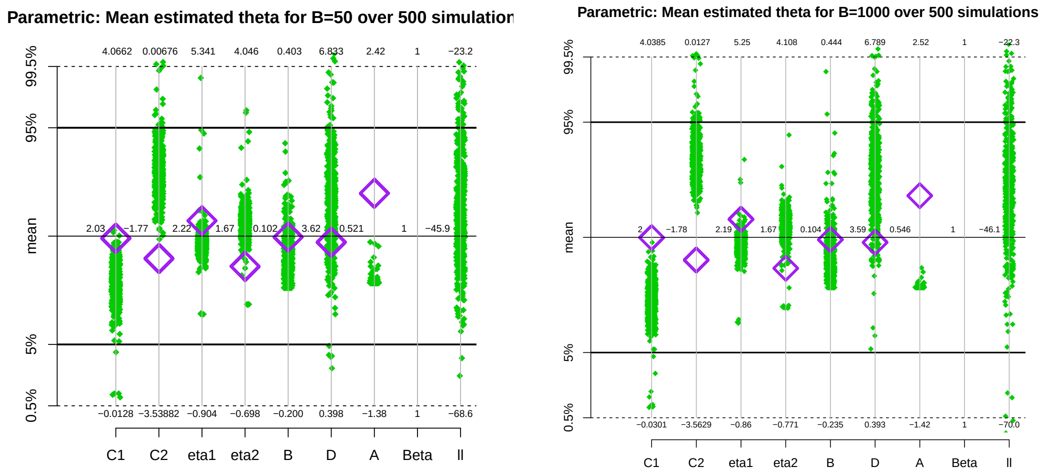
(b) Constant-n Sampling, B=1000

(c) Poisson-n Sampling, B=50

(d) Poisson-n Sampling, B=1000

Estimated parameter values $\hat{\boldsymbol{\theta}}$ for three bootstrap sampling methods, over 500 and 1000 simulated sets of points $\boldsymbol{X}$. Black lines are sample mean and quantiles estimated directly from 1000 simulations of $\boldsymbol{X}$ (see Figure A.1). Purple diamonds bracket the actual values from the source distribution $\boldsymbol{\theta}_{\mathsf{SRC}}$.

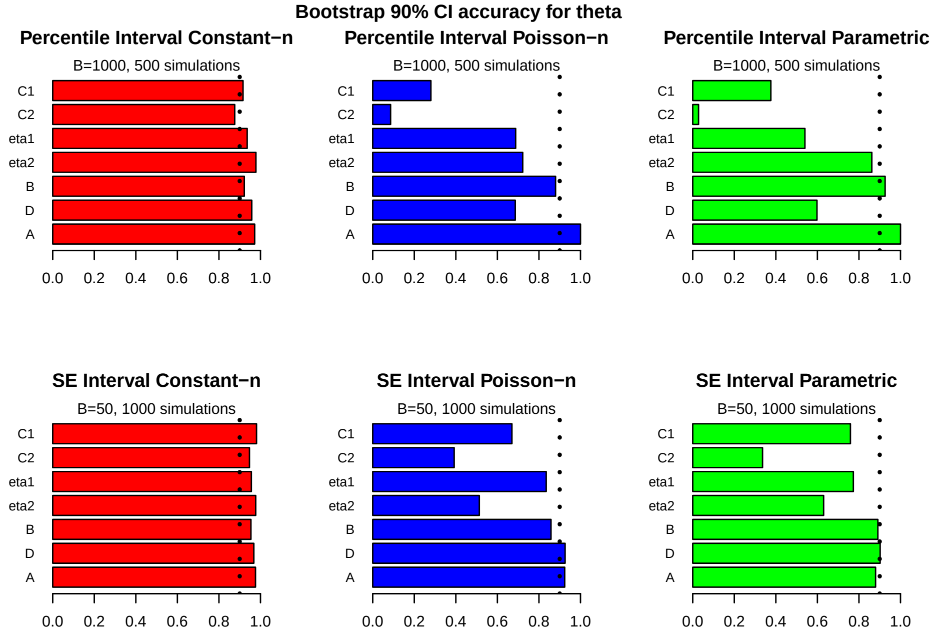
Figure A.3. Bootstrap Estimated Parameter Values



(e) Parametric Sampling, B=50

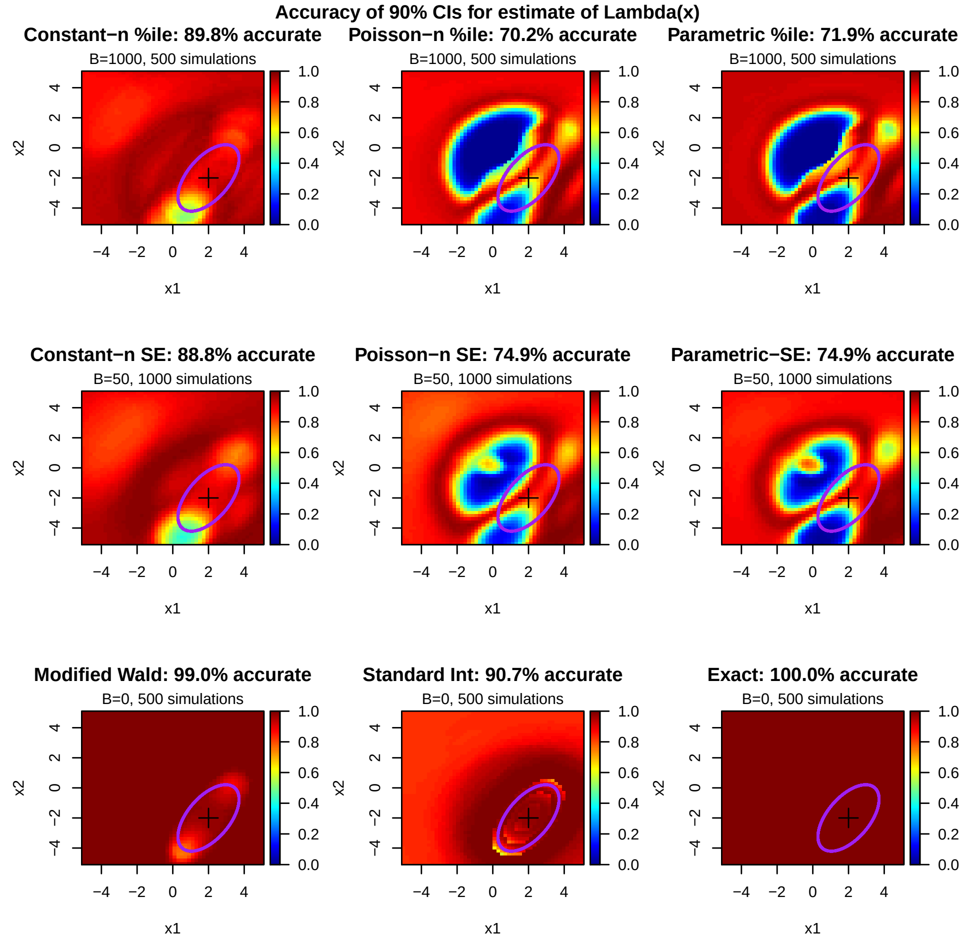
(f) Parametric Sampling, B=1000

Figure A.3. Bootstrap Estimated Parameter Values (continued)



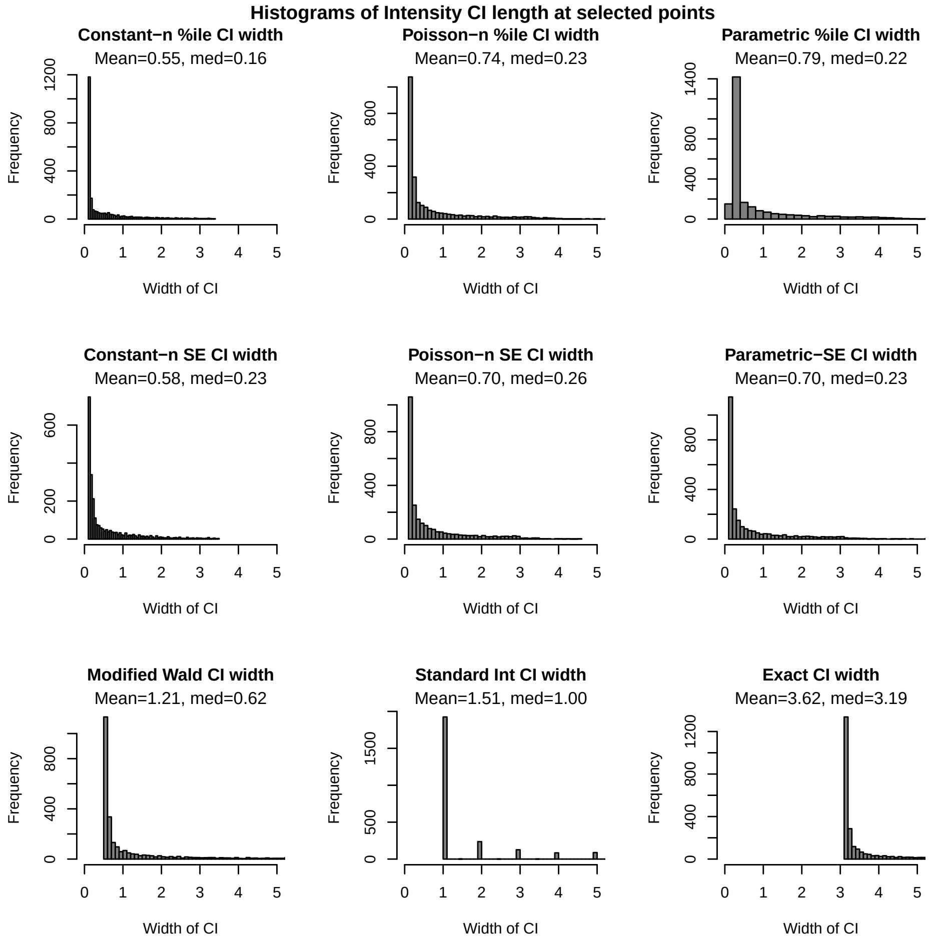
These plots show that the Constant-n sampling method (in red) is best able to meet expected accuracy of 90% for all estimated parameters, using either interval method. Non-bootstrap methods do not permit estimation of parameters $\hat{\boldsymbol{\theta}}$.

Figure A.4. Accuracy of Confidence Intervals



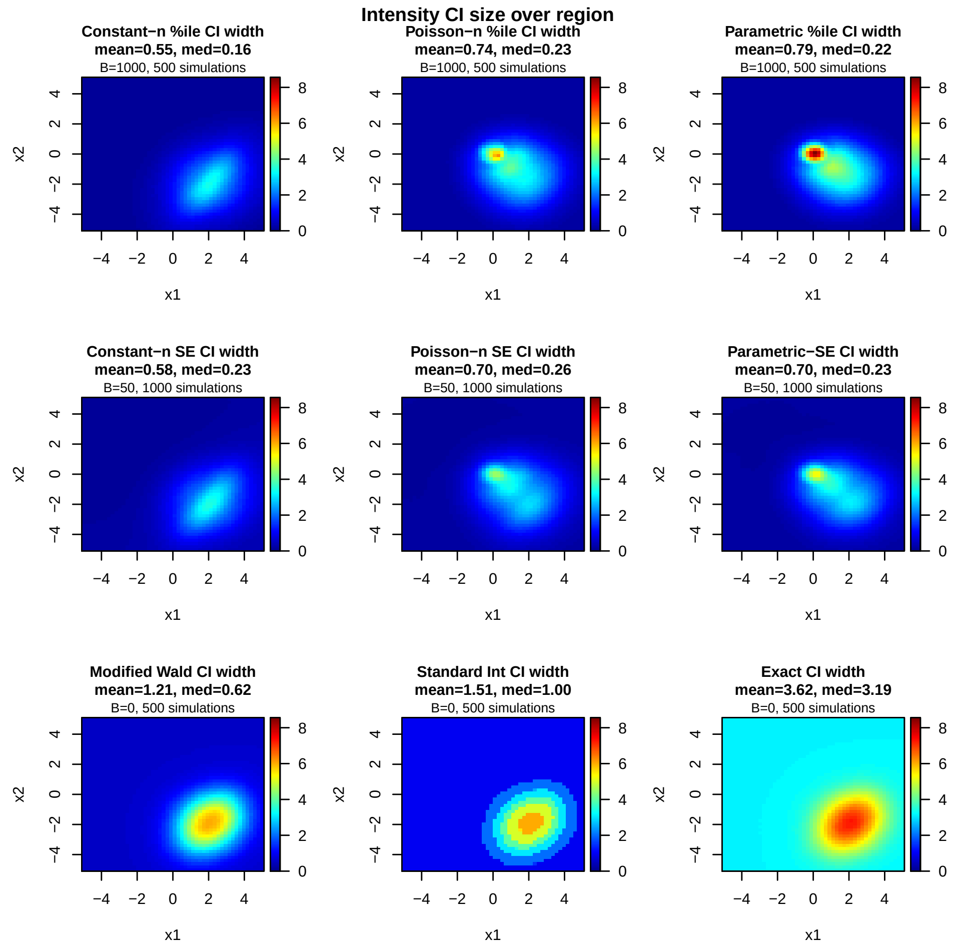
These plots show the realized accuracy of the respective 90% CIs evaluated on a 50x50 grid over the search region R. Red color represents (desirable) high accuracy; blue colors represent low accuracy. The plots show that the Constant-n bootstrap sampling method has the best spatial performance of bootstrap methods, but all non-bootstrap methods outperform it.

Figure A.5. Accuracy of Confidence Intervals over R



These histograms show the size of confidence intervals for each method. B and simulations as in Figure A.5. We can see that the bootstrap methods develop much tighter CIs than non-bootstrap methods, which is desirable.

Figure A.6. Width of Confidence Intervals



These plots show the size of the confidence interval evaluated on a 25x25 grid. Blue color represents (desirable) small, efficient intervals; red represents wide intervals. Again bootstrap methods have the tightest intervals, with Constant-n outperforming the other sampling methods.

Figure A.7. Width of Confidence Intervals over R

APPENDIX B:
Optimal Search

B.1 Optimal Search

In this appendix, we include an algorithm for optimal search without false targets over continuous space, with continuous application of search effort, (2.71) and (2.72) from [8]. This formulation assumes an input probability distribution function for the target $f_{\text{pri}}(x)$ which does not change over time as the search progresses. We have updated the notation from the reference in an attempt at consistency and to minimize confusion. We first define the following terms:

R: continuous search region in 2-dimensional space
x: a location in R; $x = (x_1, x_2)^T$
$m(x)$: a broad area search plan, as a function allocating search effort for any location x
$b(x, z)$: probability of detecting the target if it is at x given the applied broad search effort z
z: density of broad area search effort applied at a location
M: upper bound on the search effort that can be applied
$\rho(x, z)$: rate of return function. Note this differs from ρ as defined in Section 2.3.
$\boldsymbol{K}$: Total cost of the search plan (i.e. total search time)
$f_{\text{pri}}(x)$: prior PDF for target location
$c(x) > 0$: cost to apply search effort; here we use time
λ: the optimal rate of return of search effort and Lagrange multiplier.[4]

We then can calculate the following values to find the optimal allocation of broad area search effort m_λ:

$$\rho(x, z) = \frac{b'(x, z) f_{\text{pri}}(x)}{c(x)} \quad \text{for } z \geq 0 \text{ and } x \in S \tag{B.1}$$

$$\rho_M^{-1}(x, \lambda) = \min\{M, \rho^{-1}(x, \lambda)\} \tag{B.2}$$

[4] Note this differs from λ as used in 2.3 for NHPP intensity; we re-use it in this section as it is the standard notation for both NHPP intensity and Lagrange multipliers, and used in [8].

$$\boldsymbol{K}(\lambda) = \int\limits_R c(x)\rho_M^{-1}(x,\lambda)\,\mathrm{d}x \text{ for } \lambda > 0 \tag{B.3}$$

$$\lambda = \boldsymbol{K}^{-1}(K) \tag{B.4}$$

$$m_\lambda(x) = \rho_M^{-1}(x,\lambda) \text{ for } x \in R \tag{B.5}$$

Here $m_\lambda(x)$ is the optimal search plan, or allocation of broad search effort, for the total time (cost) K. This formulation uses Lagrange multipliers to find the optimal plan. λ can be found for any value of K by a one-dimension linear search, which is straightforward using numerical methods on a computer [8].

APPENDIX C: Data Table

Table C.1. Simulation Data

Total Time	# Sorties	ID Times[1]	Background	Feature	OSAP-E[2]	OSAP-ICCI[2]	OSAP-ICCI-E[2]	OSAP[2]	OSAP-E-AE[2]
50	1	a12A16	0.1	4	0.11 (0.022)	-	-	0.11 (0.022)	0.13 (0.024)
50	16	a12A16	0.1	4	0.18 (0.027)	-	-	0.18 (0.027)	0.33 (0.033)
100	1	a12A16	0.1	4	0.23 (0.029)	-	-	0.28 (0.031)	0.26 (0.031)
100	16	a12A16	0.1	4	0.41 (0.034)	-	-	0.40 (0.034)	0.57 (0.034)
150	1	a12A16	0.1	4	0.40 (0.034)	-	-	0.40 (0.034)	0.39 (0.034)
150	16	a12A16	0.1	4	0.65 (0.033)	-	-	0.54 (0.035)	0.68 (0.032)
200	1	a12A16	0.05	0.5	0.71 (0.063)	0.86 (0.049)	-	0.75 (0.061)	0.70 (0.064)
200	1	a12A16	0.05	4	0.56 (0.056)	-	-	0.55 (0.057)	-
200	1	a12A16	0.05	8	0.34 (0.066)	0.52 (0.070)	-	0.29 (0.063)	0.29 (0.063)
200	1	a12A16	0.1	0.5	0.77 (0.027)	-	-	0.76 (0.028)	-
200	1	a12A16	0.1	2	0.61 (0.032)	-	-	0.61 (0.032)	-
200	1	a12A16	0.1	4	0.51 (0.018)	0.65 (0.030)	0.64 (0.030)	0.52 (0.018)	0.52 (0.035)
200	1	a12A16	0.1	6	0.39 (0.032)	-	-	0.39 (0.032)	-
200	1	a12A16	0.1	8	0.38 (0.032)	-	-	0.35 (0.031)	-
200	1	a12A16	0.15	4	0.56 (0.056)	-	-	0.56 (0.057)	-
200	1	a12A16	0.2	0.5	0.54 (0.070)	0.60 (0.068)	-	0.57 (0.069)	0.57 (0.069)
200	1	a12A16	0.2	4	0.54 (0.057)	-	-	0.54 (0.057)	-
200	1	a12A16	0.2	8	0.32 (0.065)	0.40 (0.069)	-	0.34 (0.066)	0.32 (0.065)
200	1	a24A32	0.1	4	0.32 (0.046)	0.44 (0.049)	0.44 (0.049)	0.36 (0.047)	-
200	1	a36A48	0.1	4	0.25 (0.042)	0.27 (0.044)	0.26 (0.043)	0.23 (0.042)	-
200	1	a3A4	0.1	4	0.71 (0.045)	0.86 (0.034)	0.86 (0.034)	0.74 (0.043)	-
200	1	a6A8	0.1	4	0.64 (0.047)	0.78 (0.041)	0.81 (0.039)	0.65 (0.047)	-
200	2	a12A16	0.1	4	0.64 (0.040)	0.69 (0.039)	0.68 (0.039)	0.59 (0.041)	-
200	4	a12A16	0.1	4	0.74 (0.036)	0.66 (0.040)	0.66 (0.039)	0.66 (0.039)	-
200	8	a12A16	0.1	4	0.81 (0.033)	0.62 (0.041)	0.64 (0.040)	0.74 (0.037)	-
200	16	a12A16	0.05	0.5	0.88 (0.045)	-	-	0.86 (0.049)	0.83 (0.053)
200	16	a12A16	0.05	4	0.86 (0.040)	-	-	0.74 (0.050)	-
200	16	a12A16	0.05	8	0.70 (0.064)	-	-	0.39 (0.068)	0.74 (0.061)
200	16	a12A16	0.1	0.5	0.81 (0.026)	-	-	0.83 (0.025)	-
200	16	a12A16	0.1	2	0.83 (0.025)	-	-	0.76 (0.028)	-
200	16	a12A16	0.1	4	0.80 (0.015)	0.65 (0.040)	0.65 (0.030)	0.68 (0.017)	0.75 (0.030)
200	16	a12A16	0.1	6	0.70 (0.030)	-	-	0.61 (0.032)	-
200	16	a12A16	0.1	8	0.62 (0.032)	-	-	0.55 (0.033)	-
200	16	a12A16	0.15	4	0.89 (0.036)	-	-	0.85 (0.041)	-
200	16	a12A16	0.2	0.5	0.62 (0.068)	-	-	0.56 (0.069)	0.54 (0.070)
200	16	a12A16	0.2	4	0.80 (0.046)	-	-	0.75 (0.049)	-
200	16	a12A16	0.2	8	0.64 (0.067)	-	-	0.40 (0.069)	0.83 (0.052)
200	16	a24A32	0.1	4	0.53 (0.049)	-	0.36 (0.048)	0.45 (0.049)	-
200	16	a36A48	0.1	4	0.40 (0.048)	-	0.27 (0.044)	0.37 (0.048)	-
200	16	a3A4	0.1	4	0.95 (0.022)	-	0.93 (0.025)	0.84 (0.036)	-
200	16	a6A8	0.1	4	0.90 (0.029)	-	0.89 (0.031)	0.80 (0.040)	-
200	24	a12A16	0.1	4	0.93 (0.022)	0.64 (0.040)	0.67 (0.039)	0.81 (0.033)	-
200	32	a12A16	0.1	4	0.93 (0.021)	0.67 (0.039)	0.69 (0.039)	0.81 (0.033)	-
250	1	a12A16	0.1	4	0.63 (0.034)	-	-	0.61 (0.034)	0.61 (0.034)
250	16	a12A16	0.1	4	0.86 (0.024)	-	-	0.74 (0.030)	0.84 (0.025)

In all cases we used $b(\boldsymbol{x}, z) = B(\boldsymbol{x}, z) = 1 - \exp(-z)$, and $U = \Lambda = 1$.

[1] "a12" means $a(w) = 1 - \exp(-w/12)$. "A16" means $A(w) = 1 - \exp(-w/16)$, etc.

[2] Data is given as mean percentage of successful searches, with half-width of 95% confidence interval in (). Empty field means that combination of input factor was not simulated.

THIS PAGE INTENTIONALLY LEFT BLANK

List of References

[1] H. R. Richardson and L. D. Stone, “Operations analysis during the underwater search for Scorpion,” *Naval Research Logistics Quarterly*, vol. 18, no. 2, pp. 141–157, June 1971.

[2] L. D. Stone, “Revisiting the SS Central America search,” in *2010 13th International Conference on Information Fusion*. IEEE Publishing, July 2010, pp. 1–8.

[3] S. Davey, N. Gordon, I. Holland, M. Rutten, and J. Williams, *Bayesian Methods in the Search for MH370*. Singapore: Springer, 2016.

[4] Australian Transport Safety Bureau, “The operational search for MH370,” Canberra, Australian Capital Territory, Commonwealth of Australia, Tech. Rep. AE-2014-054, 2017. Available: https://www.atsb.gov.au/media/5773565/operational-search-for-mh370_final_3oct2017.pdf

[5] L. D. Stone, “Incremental approximation of optimal allocations,” *Naval Research Logistics Quarterly*, vol. 19, pp. 111–122, 1972.

[6] L. D. Stone, *Theory of Optimal Search*, 1st ed. New York, NY: Academic Press, 1975.

[7] S. J. Benkoski, M. G. Monticino, and J. R. Weisinger, “A survey of the search theory literature,” *Naval Research Logistics*, vol. 38, pp. 469–494, 1991.

[8] L. D. Stone, J. O. Royset, and A. R. Washburn, *Optimal Search for Moving Targets* (International Series in Operations Research and Management Science 237). Cham, Switzerland: Springer, 2016.

[9] L. D. Stone and J. A. Stanshine, “Optimal search using uninterrupted contact investigation,” *SIAM Journal on Applied Mathematics*, vol. 20, no. 2, pp. 241–263, Mar. 1971.

[10] L. D. Stone, J. A. Stanshine, and C. A. Persinger, “Optimal search in the presence of Poisson-distributed false targets,” *SIAM Journal on Applied Mathematics*, vol. 23, no. 1, pp. 6–27, July 1972.

[11] T. Kisi, “Optimal stopping of the investigation search,” in *Search Theory and Applications*, K. B. Haley and L. D. Stone, Eds. New York, NY: Plenum, 1979, pp. 255–260.

[12] K. Iida, *Studies on the Optimal Search Plan* (Lecture Notes in Statistics 70). New York, NY: Springer-Verlag, 1992, ch. 4, pp. 84–102.

[13] D. V. Kalbaugh, “Optimal search among false contacts,” *SIAM Journal on Applied Mathematics*, vol. 52, no. 6, pp. 1722–1750, Dec. 1992.

[14] B. Conolly and J. G. Pierce, *Information Mechanics*. West Sussex, England: Ellis Horwood Limited, 1988.

[15] M. Kress, K. Y. Lin, and R. Szechtman, “Optimal discrete search with imperfect specificity,” *Math Meth Oper Res*, vol. 68, pp. 539–549, 2008.

[16] L. D. Stone, “Semi-adaptive search plans,” Daniel H. Wagner, Associates, Tech. Rep. AD785295, Sep. 1973.

[17] J. M. Dobbie, “Some search problems with false contacts,” *Operations Research*, vol. 21, no. 4, pp. 907–925, 1973.

[18] P. J. Diggle, *Statistical Analysis of Spatial and Spatio-Temporal Point Patterns*, 3rd ed. (Monographs on Statistics and Applied Probability 128). Boca Raton, FL: CRC Press, 2014.

[19] S. Dirkse, M. Ferris, and R. Jain, *gdxrrw: An Interface Between “GAMS” and R*, 2016, R package version 1.0.2. Available: http://www.gams.com

[20] M. Snethlage, “Is bootstrap really helpful in point process statistics?” *Metrika*, vol. 49, pp. 245–255, 2000.

[21] S. Banerjee, B. P. Carlin, and A. E. Gelfand, *Hierarchical Modeling and Analysis for Spatial Data*, 2nd ed. (Monographs on Statistics and Applied Probability 135). Boca Raton, FL: CRC Press, 2015.

[22] V. Patil and H. Kulkarni, “Comparison of confidence intervals for the Poisson mean: Some new aspects,” *REVSTAT - Statistical Journal*, vol. 10, no. 2, pp. 211–227, June 2012.

[23] A. Cowling, P. Hall, and M. J. Phillips, “Bootstrap confidence regions for the intensity of a Poisson point process,” *Journal of the American Statistical Association*, vol. 91, no. 436, pp. 1516–1524, 1996. Available: http://www.jstor.org/stable/2291577

[24] L. Barker, “A comparison of nine confidence intervals for a Poisson parameter when the expected number of events is < = 5,” *The American Statistician*, vol. 56, no. 2, pp. 85–89, 2002.

[25] B. Efron and R. J. Tibshirani, *An Introduction to the Bootstrap* (Monographs on Statistics and Applied Probabiilty 57). Boca Raton, FL: Chapman & Hall/CRC, 1994.

[26] F. Garwood, “Fiducial limits for the Poisson distribution,” *Biometrika*, vol. 28, pp. 437–442, 1936.

THIS PAGE INTENTIONALLY LEFT BLANK

Initial Distribution List

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California