PROCEDURAL OPTIMIZATION AND MEASUREMENT OF PASSIVE ACOUSTIC SENSOR NETWORKS FOR ANIMAL OBSERVATION IN MARINE ENVIRONMENTS

A THESIS SUBMITTED TO THE GRADUATE DIVISION OF THE
UNIVERSITY OF HAWAIʻI AT MĀNOA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

IN

COMPUTER SCIENCE

MAY 2016

By

Gregory L P Burgess

Thesis Committee:

Philip M Johnson, Chairperson
Kevin Weng

Keywords: Acoustic Tracking, Acoustic Network, Animal Telemetry, Network Design, Network Metrics, Network Optimization

To myself,

Perry H. Disdainful,

the only person worthy of my company.

ACKNOWLEDGMENTS

I want to “thank” my committee, without whose ridiculous demands, I would have graduated so, so, very much faster.

ABSTRACT

Static Observation Networks (SONs) are often used in the biological sciences to study animal migration and habitat. These networks are comprised of self-contained, stationary sensors that continuously listen for acoustic transmissions released by sonic tags carried by individual animals. The transmissions released by these tags carry serial identification numbers that can be used to verify that a particular individual was near a given sensor. Sensors in these networks are stationary; therefore, sensor placement is critical to maximizing data recovery. Currently, no open-source automated mechanism exists to facilitate the design of optimal sensor networks. SON design is often governed by loose ”rules of thumb” and ”by eye” readings of low resolution bathymetric maps. Moreover, there is no standardized method for evaluating the efficacy of a SON. In this paper, we present a system which takes advantage of high-resolution bathymetric data and advanced animal modeling to provide optimal network designs. Our system also allows for statistical analysis of existing network configurations in order to create efficacy-metrics that can be used to evaluate arbitrary network configurations.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

Static Acoustic Observation Networks (SAONs) are often used in the biological sciences to study aquatic animal migration and habitat. These networks are comprised of self-contained, stationary sensors (hydrophones) that continuously listen for acoustic transmissions released by sonic tags carried by individual animals. The transmissions released by these tags carry serial identification numbers that can be used to verify that a particular individual was within detection range of a specific sensor at a given time. Acoustic networks are relatively inexpensive (compared to GPS/VHF Radio/Satellite tags). The primary goal of any tracking study is to obtain a high number of high quality data points (relating individual animals to space and time) in order to gain some insight into animal behavior. SAONs provide a way to generate a large volume of data points at low cost, resulting in cost-efficient data points. However, unless these data points are captured, the cost efficiency of SAONs is lost. Within SAONs, data capture rates are highly dependent upon the chosen locations for sensors within the study area. The malplacement of sensors (in locations that interfere with the reception of data or where no tagged individuals are present) leads to low data returns, wasted resources, and diminished cost-efficiency. We present an application that takes advantage of high resolution bathymetry, flexible behavioral modeling, and simplified acoustic propagation models to maximize the data recovery of a SAON. Our application provides a reproducible, customizable, and distributable method for generating optimal sensor placements and analytical network metrics.

1.1 Static Acoustic Observation Networks

1.1.1 Sensor Assembly

[Diagram of rigging] SAONs are composed of stationary rigs that are responsible for maintaining the chosen location for a sensor. Because positional data is interpolated from the position of nearby sensors, it is important that sensors are deployed accurately and maintain their position throughout the entire experiment[11]. This is best accomplished by attaching sensors to permanent emplacements (such as a rigid metal frame driven into a rocky substrate) that will resist substantial amounts of force (such as strong currents and curious animals). However, when it is not always possible to create such permanent emplacements (perhaps due to regulation or extreme depth), more creative approaches are called for. A popular rigging consists of an acoustic sensor attached to a length of wire/rope with a strong float on one end, and a substantial ballast with an acoustic quick release on the other[11]. Such a rig can be dropped in the ocean and allowed to sink to its desired location. Obviously, various situations will require different rig designs and may contribute significantly to network costs (acoustic releases cost approximately $2700 per piece).

1.1.2 Sensor Deployment & Recovery

The labor required for sensor deployment and recovery depends on the design of the sensor assembly. Creating a permanent, rigid emplacement for a sensor can require multiple divers, special equipment, and hours of underwater elbow grease. Recovery of a permanent, rigid emplacement will most likely require a diver to physically remove the sensor from its emplacement. Deployment of ballast/float assemblies can be as simple as dropping the assembly overboard. Recovering a ballast/float assembly simply requires signaling the acoustic release with a hydrophone and allowing the buoy to carry the sensor assembly to the surface.

1.1.3 Tag Deployment

The most challenging and time consuming task in animal tracking is the physical deposition/implantation of tags on/into the individuals to be tracked. In the case of marine tracking, this can be particularly challenging as animals must be located, captured, tagged, and released relatively quickly to avoid over-stressing the animals. Improper handling/release of an animal can result in its death and the loss of a tag. All telemetry technologies will eventually require interaction with the individual to be tracked, and acoustic tracking is no different.

1.1.4 Comparison of Technologies

1.1.4.1 Very High Frequency Radio

Very High Frequency radio (VHF) tracking involves attaching a VHF transmitter to an animal, and then using a VHF antenna and receiver to receive transmissions. VHF transmissions have effective ranges on the order of tens of kilometers. Transmissions from VHS devices do not generally contain positional data, but instead serve as a means to estimate the distance and direction of a VHS device. Positional data is derived by noting the direction and strength of a signal from several different observational positions, and estimating the transmitter's position by triangulation[8]. In a marine setting, VHF observation is generally performed from a plane or boat[1].

1.1.4.1.0.1 Sateallite/GPS Satellite and GPS tracking are distinct but related technologies that rely on a network of satellites (either ARGOS or GPS, respectively) to compute the positional data of a tag. GPS tags rely on the GPS network of satellites to triangulate a tag's three dimensional position. GPS telemetry may be stored on-board a tag (requiring later retrieval), or transmitted via satellite to a remote server[8]. Satellite tags operate by transmitting messages to the ARGOS satellite system, which computes a tag's position by observing the Doppler effect on a tag's transmission[2]. Because the telemetry from satellite tags is transmitted back to remote servers, data recovery is automatic. Both technologies have fairly poor penetration into the ocean, and so GPS/Satellite transmissions generally occur only when an animal is near the surface of the ocean. This can lead to data sets with large spatial/temporal gaps between detections. Additionally, neither technology is desirable for observing animals that reside at significant depths. Due to the high cost of Satellite/GPS technology, studies using this technology generally have very small sample sizes.

1.1.5 Advantages of Acoustic Networks

After initial deployment, SAONs require no maintenance and incur no operating costs (unlike satellite and VHF radio technologies). This means that SAONs can operate around the clock, and in conditions that would otherwise make it unsafe/impossible for field researchers to track animals (e.g. in a storm)[11]. However, it is necessary to retrieve the acoustic sensors at the end of the study in order to recover data[11]. Finally, SAONs allow for passive animal monitoring, removing the potential disruption of natural behavior caused by active tracking (e.g. aircraft/boat noise/shadow scaring animals)[11]. SAONs also function at greater depths than satellite/VHF-based systems. Because the reception of acoustic transmissions (by acoustic sensors) occurs at the resident depth of the target species, an acoustic tag's transmission need not penetrate to the surface to be detected (unlike Satellite and GPS based systems).

1.2 The Cost of Data

1.2.1 Cost of Alternative Technologies

SAONs are relatively cheap, with acoustic sensors costing $1300, and acoustic tags costing $330 each. Moorings for acoustic receivers can be significantly more expensive, with acoustic releases costing slightly more than twice the cost of a receiver. However, these costs are still significantly more affordable than satellite-based tags and collars, which cost upwards of $5000 each[5]. Additionally, recurring service fees and per-transmission charges may apply to data transferred over the satellite network. VHS radio tags are a seemingly cheaper alternative at $223 per unit, but require active monitoring to obtain each data point. The cost of paying for vehicles(boats/planes) and crews to periodically collect telemetry from these tags will significantly outweigh any initial cost savings.

Technology	Tag Cost	Receiver Cost	Operating Cost
VHF Radio Tag	$223[4]	$2940[3]	Agents & Transport
Satellite Tag	$3000-$5000[5]	$0	Service fees
Acoustic Network	$330	$1300	$0

Table 1.1: Cost Summary of Alternative Technologies

1.2.2 Operating Costs

While SAONs require no maintenance to operate, both Satellite and VHF based systems can incur operating costs while deployed. Satellite tags will require very little maintenance (precluding animal mortality), but satellite network operators may charge for access to and transmission over their network[5]. VHF systems require little maintenance, but do require active field work in order to obtain positional information. Because a tag's location is interpolated from observations of its VHS signal from multiple locations, it is necessary for a field agent to routinely collect these observations to obtain telemetry data. VHF networks have perhaps the highest operating cost, requiring a salary for one or more field agent(s) and transportation costs (renting a boat/plane). The operating costs for each technology should be included in the total cost of data collection, and subsequently the cost effectiveness of each solution.

[http://www.wildlifetracking.org/faq.shtml] [http://www.lionconservation.org/lion-collars.html] [http://www.africat.org/projects/radio-collars-for-lions]

VR2's cost $1.5k each, tags cost $350 each. http://www.gulfcounty-fl.gov/pdf/882532513025603.pdf

1.2.3 Cost Efficciency

Technology	Tag Model	Transmit Period	Expected Lifespan	Expected Transmissions
VHF Radio Tag	Telonics FIS-040	1s	0.7 days	60,480
Satellite Tag	Telonics ST-18	60s	117 days	168,480
Acoustic Tag	Vemco VR-13	90s	1,135 days	1,089,600

Table 1.2: Lifespan & Total Expected Transmissions

Technology	Tag Model	Tag Price	Expected Transmissions	Price Per Transmission
VHF Radio Tag	Telonics FIS-040	$199	60,480	$\$3.29e^{-3}$
Satellite Tag	Telonics ST-18	$3000	168,480	$\$1.78e^{-2}$
Acoustic Tag	Vemco VR-13	$330	1,089,600	$\$3.03e^{-4}$

Table 1.3: Price per Transmissions

http://vemco.com/products/v7-to-v16-69khz/ http://www.wildlifetracking.org/faq.shtml http://www.telonics 18/ http://www.mrcmekong.org/assets/Publications/Catch-and-Culture/catchmar02vol7.3.pdf

1.3 State of the Art

1.3.1 Rules of Thumb for Sensor Placement

While animal tracking studies draw upon hard data to draw conclusions, the methodology for collecting this data is based upon loose rules of thumb driven by anecdotal evidence. Heupel et al's highly cited 2006 paper distills their prior experience with animal tracking into ”rules of thumb” for designing a SAON. These rules focus on generalized considerations such as avoiding areas of high noise, bathymetric obstruction, and acoustic echoing[11].

While Heupel et al's publication gives sound advice on design issues that warrant warrant consideration, the discussion of analytical methods for measuring these issues falls out of the publication's scope.

1.3.2 Metrics

1.3.2.1 Data Recovery Rate

The most common metric used in analyzing the success of animal tracking studies is the Data Recovery Rate (DRR): $(\frac{pings\_emitted}{pings\_recovered})$. While it may seem intuitive to understand data recovery rates as an indicator of the quality of the dataset, one must bear in mind the objective of the study. As illustrated in section 1.3.3, the objective of the study defines how useful a particular dataset is in addressing a research question. Therefore, Data Recovery Rate should be treated simply as a measure of how complete a particular dataset is, and how strongly it can support a claim.

1.3.2.1.1 Unique & Absoloute Data Recovery Rates When discussing recovery rates, the Absoloute Data Recovery Rate (ADRR) and Unique Data Recovery Rate (UDRR) can give insight into the qualities of a particular network design. ADRR is computed as the total number of pings that were received by all sensors in the network, divided by the total number of emitted pings. This means, that data recovery rates of more than 100% are possible.

1.3.2.2 Network Sparsity

Network Sparsity (δ) is a unit-less measure of the observational qualities of an acoustic network. This metric is useful in quickly expressing the density and intent of an acoustic network. For a list of n receivers within a SAON r_1, r_2, r_3, ... r_n, let a_i $(1 \leq i \leq n)$ be the distance to receiver r_i's nearest neighbor. Then, a is the median over all a_i. d_r is given as the detection range of a sensor. Network sparsity is defined as $\frac{a}{2d_r}$.

A δ of 0 describes an array of receivers that are virtually stacked on top each other. A *delta* between 0 and 1 indicates that receivers are placed such that their detection ranges overlap (a smaller δ indicates more overlap). A δ of 1 signifies that receivers in the array are positioned such that their detection ranges are just touching but not overlapping. A δ greater than 1 indicates that the receivers are farther apart, and that there are gaps between receiver coverage areas. With this definition, it becomes obvious that Network Sparsity is a positive indicator for data fusion (section 1.3.3.1), and data resolution(section 1.3.3.1).

1.3.2.3 Sample Size

Another important factor to consider is the number of tagged individuals within a dataset. A dataset for a single tagged individual, no matter how complete, will not offer very much support to any conclusions drawn. At the same time, a dataset for a large number of individuals, with very low data recovery rates may not provide enough individual telemetry to draw a conclusion at all.

1.3.3 Data Quality

1.3.3.1 Data Resolution

Acoustic receivers like the Vemco VR2 log detections of acoustic transmissions as a tuple of time, tag number, and transmission strength. The strength of the received transmission can be used to approximate the distance between the tag and receiver. Data from a single receiver has a fairly low certainty of the exact position (low resolution) of the transmitting tag because only a single distance can be observed. If multiple receivers are in close enough proximity to receive the same transmission, the strength of the transmission observed from several different known, fixed positions can be used to triangulate a more precise position (higher resolution). This process of combining multiple observations into a more accurate observation, known as **Data Fusion**, is useful for increasing the resolution of tagged individuals and allows for the tracking of fine movements within a three-dimensional space. This increase in resolution does however have a price. Detection of a tag by more than one receiver requires that those receivers have overlapping detection ranges. Assuming a fixed number of receivers, placing receivers closer together reduces the actual coverage area of the array. Thus, the coverage area of the array is inversely correlated with the resolution of the array. Alternatively, purchasing more receivers will achieve a higher resolution, but increases the cost of the array.

1.3.3.2 Meaningful Data

Obviously the number of sensors and the size of the research area determine Data Resolution. However, high-resolution data, while desirable, is not always critical to the study. First consider a number of receivers placed in a tight cluster.

If the target species were highly sedentary, and the receiver cluster was placed around the area where a large number of individuals were captured and tagged, then the study would very likely yield a high Data Recovery Rate, but that dataset would be of little use in determining the spatial distribution of that species, as the data would be limited to the small area in which the receiver cluster was placed. On the other hand, this dataset would be highly useful in confirming the sedentary nature of the species and defining a small home range. Additionally, data from multiple receivers could be combined to provide high-resolution telemetry for the location of an animal over time (see section 1.3.3.1). This high-resolution telemetry could be used to infer fine-scale co-location of two individuals, giving insight into social movement behaviors such as schooling and mating.

If the target species tended to roam over a large home range, then it is likely the cluster would receive only a few pings. In this case, little data will be gathered in regards to the extent of the specie's home range, but high resolution telemetry can be gathered for a short time if the animal passes through the cluster. This might be useful in identifying particular corridors that individuals prefer.

Now consider a number of receivers placed in an array very far apart from each other over a very large spatial area. If the target species were highly sedentary, then individuals might show up on one or two receivers. If many individuals were tagged, then the dataset could describe the spatial distribution of the species over a large area. The data on particular individuals would likely be very low resolution, as detection by a single receiver only tracks proximity.

If the target species tended to roam over a large home range, then it is likely the receiver array would pick up an individual animal over a number of distant sensors. This data could be used to detect potential corridors for animal movement, establish individual home ranges, and to identify the spatial distribution of the species over a large area. Again, the telemetry for individual animals would be very low, but the detection of many individuals by a single receiver could indicate areas of interest for future research.

1.3.4 Scale of Experiments

1.3.5 Oversights

CHAPTER 2
RELATED WORK

2.1 Yuan et al - Fast Sensor Placement Algorithms for Fusion-based Target Detection

Using data fusion for enhanced range and accuracy Constrained Simulated Annealing and Optimal Placement

2.2 Poduri et al Constrained Coverage for Mobile Sensor Networks Constrained Coverage (K-Neighbor Networks vs Maximum Coverage)

Density of Deployment Influencing global network properties via local restrictions Force dispersion algorithm

2.3 Pedersen and Weng - Estimating Individual Animal Movement from Observation Networks

Movement models Observation models Network Sparsity Home Range Investigation Assumptions when simulating fish movement in state-space models Fish speed and sensor area Observable space and total study area ?Environmental factors affecting fish behavior?

2.4 Howard et al - Mobile Sensor Network Deployment using Potential Fields Potential Field Algorithm

Static Equilibrium: Optimal placement vs Run time Runtime and Results

2.5 Akbarzadeh et al - Probabilistic Sensing Model for Sensor Placement Optimization Based Signal Simulation and Attenuation (Omni Directional Sensors)

Line of Sight modeling Weighted Coverage L-BFGS, Simulated Annealing, and Covariance Matrix algorithms Here is where you discuss the related work. Use BibTex to reference related work.

CHAPTER 3
DESIGN

3.1 Program Requirements

3.1.1 Motivation

While the detriments to SAON technologies are well-known [6], [11], [12], [13], [19] there few tools/services to analytically design SAONs around them. Further, none of these tools/services are free and open-source.

3.1.1.1 Cost Efficiency

In section **??**, we discuss the costs of marine telemetry systems, noting that acoustic telemetry systems produce data at a significantly lower (≥10x cheaper) cost than VHF or GPS/Satellite based technologies. In order to maintain the cost-efficiency of acoustic technology, at least 10% of the produced transmissions must be captured by the SAON's receiver array. Given the numerous (but avoidable) impediments to reception of these acoustic signals (1.3.1), the array-design process becomes critical to maintaining the cost-efficiency of SAON technologies. A free network design tool would help to maintain the cost-efficiency of SAONs by eliminating costs surrounding their design and evaluation.

3.1.1.2 Metrics

The computation of network metrics (Absolutes Recovery Rate, Unique Recovery Rate, Network Sparsity) is very labor intensive at large scale. Additionally, the process of computation may vary from experiment to experiment. An automated tool would solve both issues by providing a fast, simple, repeatable, and well-documented method for computation. Metrics from such a tool would be useful in directly comparing different network deigns.

3.1.1.3 Transparency

An open-sourced tool/service would make the design process more transparent, permitting peer-review and modification. This would provide increased confidence in the process, and increased adoption of the tool. Increased adoption would result in a larger number of efficient SAONs, leading to higher data recovery rates, better data quality, increased return-on-investment, and the ability to better address scientific-research questions.

3.1.2 Supported Workflows

3.1.2.1 Static Analysis

As mentioned in section 3.1.1.2, a primary motive for this tool was the ability to create a repeatable means of measuring the performance of a SAON. To this end, the ability to measure an existing network design is important. Users should be presented with network metrics after specifying bathymetry, receiver locations, network properties, and an animal model for a given study site.

3.1.2.2 Optimal Design

The primary motive for this tool is the ability to design optimal SAONs. Users should be presented with a network design (optimal receiver locations), and network metrics after specifying bathymetry, the number of receivers in the network, network properties, and an animal model for a given study site.

3.1.2.3 Optimal Addition

Similar to the problem of optimal design, is the problem of optimal addition: the augmentation of an already existing SAON. Users should be presented with a network design (optimal augmenting receiver locations), and network metrics after specifying bathymetry, the number of receivers to add to the network, network properties, existing receiver locations, and an animal model for a given study site.

3.2 Conceptual Model

3.2.1 Time/Space Modeling

3.2.1.1 Spatial Modeling

To model a 4-dimensional underwater environment (a 3-dimensional spatial grid of various attributes), we use a two-dimensional grid of cells (in the x and y dimensions) containing numerical values. Numerical values in those cells, combined with other user-defined values, can then be used in various shape functions to generate a third dimension (z) of values for that cell. In this way, we save significant amounts of memory by computing values for a specific three-dimensional cell on the fly, instead of storing an additional dimension of values.

3.2.1.2 Temporal Modeling

With respect to the passage of time, our model assumes that receivers are stationary throughout the entire experiment. Furthermore, the animal model does not represent animal movement over time, but the percentage of time an animal would spend in a particular cell over the entire study period. The animal model therefore represents the tendency of an animal's movements over the expected study period rather than its particular movements in a small time period. As a result, we need not consider temporally-related phenomena.

3.2.2 Bathymetric Modeling

3.2.2.1 Bathymetric Grid

Bathymetry files are generally given as two-dimensional matrix of numerical values or a list of x,y,z values. Bathymetric files describe a three-dimensional space as a regular grid of rectangular prisms (cells) with constant length (x-dimension) and width (y-dimension), but varying negative (depth is negative) heights (z-dimension). The resolution of a bathymetric file is given by the x and y (length and width) dimensions of its cells. For example, a 50 meter Bathymetry file has cell sizes of approximately 50 meters square (although these cells are not necessarily perfectly square). Bathymetric files list beginning and ending coordinates (North/South Latitudes and East/West Longitudes), as well as the grid size (in rows and columns) in cells. With these two measurements, one can compute the degrees per cell of latitude and longitude. Thus, particular latitudes and longitudes can be converted to rows/columns, and back.

Our program works on a two-dimensional, grid-based system, taking advantage of the grid given by the user-provided bathymetric file. As stated in section 3.2.1.1, the bathymetric grid is a grid containing numerical values that describe a third dimension (depth). The exact spatial extent of this description is given by the bathymetric file. Thus, the resolution of our program's output is dictated by the resolution of the input bathymetric file.

3.2.2.2 Bathymetric Filetypes

Two highly popular file formats in Geographical Information Systems are provided by NetCDF and ArcGIS. NetCDF provides an open source file format that lists a header of metadata and a white-space delimited matrix of numerical values. ArcGIS is a private institution that supplies many different file types, formats, and encodings for a family of GIS-related software systems. ArcGIS also supports the encoding and transcription of its proprietary formats to the NetCDF format. Due to the large number of possible format and encoding combinations in ArcGIS file formats and the ability to translate file these various formats to NetCDF, we natively support the NetCDF standard, and assume that users are capable of converting their data into the NetCDF format.

3.2.2.3 Bathymetric Resolution

Two key components of our program are the animal model and the bathymetric shadowing model. These models make decisions based upon the depth at a particular cell and the distance between cells, data which is governed by the input bathymetry file. As stated in Section 3.2.2.1, the resolution of the program's output is dependant upon the resolution of the input bathymetry file. Obviously, higher resolution grids will offer higher resolution results; but, high-resolution bathymetric files tend to be difficult to come by. These files are often held by private agencies, or simply never released to the public. It may then seem useful to artificially increase the resolution of the simulation by dividing the input file's bathymetric cells into sub-cells of finer resolution, but doing so increases the computational size of the program without meaningfully increasing the accuracy of the results.

When subdividing cells, either the sub-cells are given the same depth as their parent cell, or the depth of a sub-cell is interpolated from surrounding cells by some smoothing function. Subdividing a cell into sub-cells with the same depth makes the assumption that all sub-cells are actually the same depth. Furthermore, this results in the animal and bathymetric shadowing models making the same depth-based decision for all sub-cells that they would for the larger parent cell, increasing the computational load (see Figure 3.1b). Subdividing a cell into sub-cells with a depth governed by a smoothing function makes the assumption that there are no impeding obstacles between neighboring cells, and that there is a smooth transition between them (see figure 3.1c). Subdividing the two cells in Figure 3.1a half (ignoring for now the y component of our grid) results in the four sub-cells with depths given by a smoothing function (Figure 3.1c) would result in the large depth change at the sheer cliff face in Figure 3.1a being smoothed into smaller changes in depth, which would allow the unobstructed transmission of acoustic signals.

Both strategies (duplicating depth and applying a smoothing function) for artificially increasing the resolution of a bathymetric file disregard the manner in which the bathymetry was originally observed. Bathymetry is almost always computed as the average observed depth at several points within a geographic area of a given size (resolution). For example, imagine a particular cell in a bathymetric grid has a steep cliff running across the middle. Assuming that the sea floor at the top and base of the cliff were perfectly flat, and one measurement was taken at the top of the cliff and one at the bottom, the cell would have a depth equal to the average of the two observed depths. This average depth would then represent the depth for that entire cell, modeling it as a perfectly flat surface. Obviously this is problematic as the true nature of the sea floor is misrepresented. This misrepresentation leads to two conflicting arguments, the first is that the application of smoothing functions or duplicating depths of already averaged data makes faulty assumptions about real-world bathymetry. On the other hand, because source bathymetric files already represent aggregate data, one could argue that any conclusions drawn from the source bathymetry files are already faulty. We argue that the conclusions one can make are only as good as the bathymetric information available. As the resolution of measured bathymetry (bathymetric measurements taken from the real world) increases (becomes finer), so too does the accuracy of the simulation. While artificially increasing the resolution of a source bathymetric file may skew results, it is sometimes useful for meshing two bathymetric source files of varying resolution into one larger bathymetric dataset. Our program does not currently support modifying the resolution of source bathymetric files.

3.2.2.4 Bathymetric Shadowing

In the real world, the transmission of an acoustic ping originates from the tagged animal and propagates to the receiver. This propagation is governed by complex interactions with the surrounding environment (including the bottom substrate, water density, distance to the surface/sea-floor, thermoclines, the number of tagged animals in the vicinity, and ambient noise). Because it is difficult to model these phenomena without significant data on a large number of variables over a large area, our simulation uses a simplified propagation model relying on direct line of sight between a tagged animal and a receiver. Simply put, in order to receive a transmission, there must exist a physically-unobstructed path between a tagged animal and a receiver.

3.2.3 Animal Modeling

3.2.3.1 Behavior Grid

Animals exhibit many different movement patterns and habitat preferences (both of which can vary in three-dimensional space). This greatly affects their distribution within the study space, and thus the network configuration that should be deployed to capture their movement. To describe the distribution of transmissions released from tagged animals, we utilize a two dimensional grid of the same dimensions and resolution as the Bathymetry Grid (Section 3.2.2.1). Each cell in this ”Behavior Grid” contains a positive real value indicating, out of all transmissions that will be released over the entire experiment period, the percentage of transmissions that are expected to be released from within that cell's water column. This value can loosely be though of as the ”animal-residency” of a cell since we are effectively measuring the number of animal-hours spent in that cell.

To populate this 2D grid with values, we provide two behavioral models to simulate the horizontal distribution of animals, and two optional models for the vertical distribution. Note: we use the terms ”animal” and ”transmission” interchangeably as we are interested in capturing acoustic transmissions, which are given off by the tags carried by the target species.

3.2.3.2 Animal Movement Modeling

To simulate tagged animal movement across the two dimensional x/y space (as one would expect to see on a map from above), we provide two basic probabilistic movement models: Random Walk, and Ornstein-Uhlenbeck(OU).

3.2.3.2.1 Random Walk Model The Random Walk model assumes that animals move randomly through the environment. As a result, over the entire study period, each valid grid cell (as defined by the Restricted Vertical Habitat Range (Section 3.2.3.4)) will see an equal amount of animal traffic. The result is that every valid cell in the grid will have the same chance of capturing an animal's acoustic transmission. We assume that tagged individuals will be willing and able to very briefly (in probabilistically negligible time frames) pass through inhospitable (over dry land, through impassible terrain) cells to get to other cells. This means that disjoint sections of habitat are still equally likely to see animal presence.

3.2.3.2.2 Ornstein-Uhlenbeck Model The Ornstein-Uhlenbeck(OU) model[14] supports the idea that over time, animals will tend to congregate near certain points of interest. This concept models an animal's desire to seek out and remain near a physically significant structure, a region of high food availability, breeding grounds, shelter, etc. The OU algorithm allows for the modeling of the attraction towards a focal point in the x/y directions separately. Assuming a center point at the origin of a Cartesian grid, increasing the attraction in the x-direction will bring the distribution closer to the x-axis, and decreasing it will spread the distribution away from the x-axis. A correlation value ($-1 \leq x \leq 1$) allows for tilting the angle of the distribution. A positive correlation value tilts the distribution clockwise, and a negative correlation value tilts it counter clockwise. Correlations of 0 (no tilt), 1 (180° tilt), and -1 (-180° tilt) will have no observable effect on the distribution.

3.2.3.3 Habitat Preference

Some animals exhibit the preference to reside within a specific section of the water column; for example, prey animals may prefer hiding in reef heads at the bottom of the water column, while predators will prefer to hover several meters off the bottom. This preference can be incorporated into the animal model by specifying mean (preferred Depth") and standard deviation("SD of Preferred Depth") values. These values are given as a measure of the distance (in meters) from the bottom. For example, specifying a depth of '0' for Preferred Depth indicates that the animal prefers to live on the sea floor, while a value of '5' indicates that the animal prefers to live 5m off the sea floor. Allowing a standard deviation value allows for the modeling of animals that tend to be sedentary within the water column (a small deviation), and those that migrate through the water column (a large deviation). Currently, the simulation does not support the modeling of sub-surface animals, and any part of the distribution curve falling below the sea floor will be redistributed along the rest of the curve.

3.2.3.4 Restricted Vertical Habitat Range

Some animals will live only in a specific depth range. For example, a deep sea fish may live only in depths of 300-400 meters. To incorporate this into the behavioral model, users can specify a minimum and maximum vertical habitat range for their animal. If this option is selected, the program will only simulate animals in cells whose maximum depths are between the given minimum and maximum depths. As in the Random Walk Model, we assume that animals are willing and able to move between disjoint areas of habitat.

3.2.4 Receiver Modeling

3.2.4.1 Acoustic Attenuation

As acoustic transmissions travel away from their point of origin, they experience significant attenuation. We model this attenuation of acoustic signal as a deteriorating function following a Gaussian distribution. Given a particular distance between a tag and receiver, this distribution describes the probability of capturing the transmission. Depending upon the environment, the highest probability of detection generally occurs a few meters away from the receiver.

3.2.4.2 Detection Range

The distance at which a transmission from an acoustic tag can be captured is largely determined by the transmission power of the acoustic tag. In our model, we assume that all animals have the same model of acoustic tag, and that ”transmission range” is instead a property of the receiver referred to as ”Detection Range” (D_{range}). We define the Detection Range of a receiver to be the average distance (specific to the given study site) at which the probability of detection drops to 5%.

3.2.4.3 Detection Area

We define the Detection Area of a receiver to be the square plane of grid cells around a receiver which will be considered when evaluating a potential receiver emplacement. Theoretically, the Detection Area of a receiver is a circle with a radius equal to the receiver's Detection Range (D_{range}). Within our program, we model the Detection Area as a square grid of cells with an edge length equal to $2^*D_{range} + 1$, centered on the cell where a receiver will theoretically be placed. The additional cell ensures odd dimensions for the square, so that there exists a central cell to model as the receiver. We model a square instead of a circle for simplicity of book keeping (we simply keep track of the Detection Range, and starting row/column indexes). The theoretical Detection Range can then be imagined as a circle circumscribed in the square of our model. At first glance, it might seem that this simplification would alter the probabilistic distribution models as additional ”gray” cells (those outside the circle but within the square) are being considered. However, this is not the case as those models utilize exponential functions based on the absolute distance from the receiver. Gray cells will therefore contribute exponentially less to the total probability than those cells within the Detection Range, and their effect on the final probability will be negligible.

3.2.4.4 Network Specification

There are three distinct ways to place receivers into the model: user specification, optimal placement, and optimal projection. User Specified Receivers (USR) represent receivers that are already deployed at the study site and are being integrated into a larger network. Program Placed Receivers are receivers that will be optimally placed by the program, taking into account existing (user specified) receivers placements using the suppression dynamic explained in section 3.4. Projected Sensors are PPSs that do not count towards the network statistic rates returned by the program. Instead, their contributed recovery rates are given separately, and represent the marginal benefits of incrementally placing more receivers.

3.3 Evaluation of Receiver Emplacements

Section 3.2.3 discusses the models used to simulate animal movement. These animal models populate the "Behavior Grid", a 2D grid (of the same size as the specified bathymetry grid) which gives the distribution of animals across the study area, and a shape function which describes a normal distribution of animals within the water column at various depths. Section 3.1a discusses the Line of Sight (LoS) model used to simulate the obstruction of acoustic transmission. Section3.2.4 discusses how attenuation affects the propagation of acoustic signals and thus the probability of detecting (P_{range}) acoustic transmissions from a distance. Together, these models are used to evaluate receiver locations throughout the study site.

3.3.1 Goodness Grid

In modeling our 3D environment (the study site), we make extensive use of 2D grids with a third dimension. However, we assume all receivers in the study will have the same elevation off the sea floor. Recall that our program will evaluate every cell in the Bathymetry Grid as a potential receiver emplacement. Together with the assumption of fixed receiver elevation, the search space considered by our program is a 2D surface parallel to the sea floor. Thus, it makes sense to store the evaluation metric for this search space in a 2D grid with the same dimensions and resolution as the Bathymetry Grid. This gird is referred to as the "Goodness Grid" (G).

3.3.2 Evaluation Algorithms (Bias)

The Evaluation or Goodness algorithm is the driving force behind the evaluation of receiver placements. Given a particular cell at row i and column j, an Evaluation Algorithm will compute a non-negative, rational value representing how much data a receiver placed d meters off the bottom of cell (i,j) would recover, according to the particular bias of a given evaluation algorithm. These values are computed for all cells within the Bathymetry Grid, and stored in the Goodness Grid. $G_{i,j}$ refers to the goodness value of the cell at row i, column j on the Goodness Grid(G).

While users are able to write their own Goodness algorithms, three basic algorithms are provided. All Evaluation Algorithms compute the goodness of a potential receiver location ($G_{i,j}$) by summing the number of Estimated Receivable Transmissions (ERT) from all cells within Detection Range of $G_{i,j}$. Each algorithm has a particular bias in its approach to computing ERT. Note: we utilize the terms "bias" and "algorithm" interchangeably when referring to "Evaluation Algorithms".

Explicitly, $G_{i,j} = \Sigma_{i=i_0}^{i_n} \Sigma_{j=j_0}^{j_n} ERT(x, i, j)$, where x is the chosen Evaluation Algorithm/Bias, i_0 and j_0 represent starting row/column indexes (respectively) for the Detection Area, and i_n and j_n represent ending row/column indexes (respectively) for the Detection Area. Given that the Detection Area is of size $(2 * D_{range} + 1)^2$, the expected computation time of evaluating

3.3.2.1 Animal Only (Option 1)

This option prefers to place receivers in areas of high animal activity, completely oblivious to the surrounding bathymetry. This is useful for when no bathymetric information is available or when animal activity occurs well above the sea floor. The "Animal Only" option computes ERT for a cell as the animal-residency of a cell (according to the Behavior Grid), multiplied by the probability of detection due to attenuation for that cell's distance from the receiver.
Explicitly, $ERT_{1,i,j} = B_{i,j} * P_{Attenuation}(Range(i, j))$

3.3.2.2 Visible Fish (Option 3)

This option chooses receiver locations that have the best view of areas of high animal residency. Both animal presence and visibility due to topography are considered. This option is most useful when both well-documented animal behavior and high resolution bathymetry are available. ERT is computed as animal-residency (according to the Behavior Grid), multiplied by the percentage of visible fish (given either as a simple percentage of the visible water column, or by integration over the shape function described in Habitat Preference), multiplied by the probability of detection due to attenuation. Figure **??** depicts this operation.
Explicitly, $ERT_{3,i,j} = B_{i,j} * P_{observation} * P_{Attenuation}(Range(i, j))$

3.3.2.3 Topography Only (Option 2)

This option places receivers in areas that have the best visibility of the surrounding area, regardless of the expected animal-residency. This is useful for experiments where animal habitat is unknown or to be determined. Here, ERT is computed as the number of observable transmissions from a

uniformly distributed animal model. While this algorithm is referred to as ”Topography Only”, we implement this by using the ”Visible Fish” algorithm with a simplified animal model. Specifically, we assume that animals are uniformly distributed throughout the environment. Because animals are uniformly distributed, the more volume a receiver can see, the more transmissions it will receive. Thus, the algorithm will value cells with the best possible view of the surrounding water columns. Explicitly, $ERT_{2,i,j} = B_{i,j} * P_{observation} * P_{Attenuation}(Range(i,j))$
Note: With this option, B is initially described as a uniform distribution.

3.3.3 Line of Sight Evaluation

Section 3.2.2.4 explains the concept of Bathymetric Shadowing, which requires a bathymetrically un-obstructed Line of Sight (LoS) to exist between an acoustic tag and a receiver in order for transmissions from that tag to be received. Section 3.3.2.2 discusses how the evaluation algorithms utilize this concept to compute the quantity of transmissions that are receivable. Specifically, the evaluation algorithm needs to know the deepest depth (D_{max}) in a cell that is visible to a receiver. To determine the D_{max} for a cell q, relative to a receiver in cell p, we must first determine which cells could potentially obscure LoS between p and q. We refer to a list of n such potentially obscuring cells as $C_{0...n}$. Figure 3.1a illustrates this process. The shaded cells in the left image potentially obstruct the vision from cell p to cell q. Herein, we use D_x to refer to the depth of cell x, as given in the bathymetry file. Additionally, we declare that x_r and x_c refer to the row and column indexes (respectively) of the cell x on the Bathymetry Grid.

Next, we determine the slope between the receiver at p and each cell in C. The slope between a receiver elevated k meters off the sea floor of cell p, and v ($m_{p,v}$), is defined as: the difference in elevation between $D_p + k$ and D_v divided by the Euclidian distance $Edist_{p,v}$ between cell p and cell v. Explicitly, $m_{p,v} = \frac{(D_v - D_p + k)}{Edist_{p,v}} = \frac{(D_v - D_p + k)}{\sqrt{(p_r - v_r)^2 + (p_c - v_c)^2}}$. The greatest (largest signed value) slope $m_{p,v}$ for all cells v in C is the Critical Slope (m_{crit}). The greatest obstruction to the LoS between p and q will occur along this critical slope. If we project a line with this slope from the receiver at p to our target cell q, the line would be tangent to the tallest obstruction along our LoS. Thus, the critical slope determines D_{max} for cell q. Explicitly, $D_{max} = m_{crit} * Edist_{p,v} + D_p + k$.

3.3.4 Selection of Optimal Emplacements

Section **??** describes the evaluation of cells as potential receiver locations. The Optimal Design and Optimal Addition work flows (section 3.1.2) require the identification and selection of a user-specified number of receiver emplacements. Once all cells within the area of interest have been evaluated, the program simply selects the highest rated cells. The program first selects the user-specified number of optimal receiver locations, and then the number of specified projected receivers.

3.4 Suppression

As previously mentioned, the optimality of a network depends greatly upon the way in which data quality is defined. Some users will want to design a network that covers as much of a study area as possible, while others might want to heavily saturate a small area with receivers to facilitate higher resolution data. Others may wish to find the receiver locations that return the highest number of unique data points. Also previously mentioned was the metric of Unique Data Recovery Rate (UDRR) and Absoloute Data Recovery Rate (ADRR) [Section 1.3.2.1]. This model reflects the idea that the UDRR is more important than the ADRR, as transmissions received by one receiver are no longer desirable to any other receiver.

3.4.1 Suppression Area

As an input to the program, users can specify a Suppression Range Factor as a positive real number. This factor is used to scale the Detection Range (Section **??**) and define a grid similar to the Detection Area (Section **??**). The resulting grid is referred to as the Suppression Area. The Suppression Area is centered over a receiver placement and cells within this area have their number of undetected transmissions reduced according to a Suppression Algorithm. The suppression mechanic is used during the selection phase of the program. After selecting each optimal receiver location (Section 3.3.4), the Suppression Area around that receiver location is suppressed. The result of this mechanic is that emplacement selection is a sequential process, where selecting each emplacement depends upon the previous selection.

3.4.2 Suppression Algorithms

To promote the flexibility of our program, we provide users two algorithms for suppression. The first offers a lower computation time at the cost of divergence from our theoretical model. The second offers a higher fidelity model, but requires significantly more computation.

3.4.2.1 Static Suppression

The Static Suppression algorithm takes a ”cookie-cutter” approach and reduces the number of undetected transmissions in the Suppression Area by a user-specified factor (all cells within the Suppression Area are multiplied by a user-specified non-zero real number). This computation is very simplistic, requiring only a scalar multiplication, and as such runs very quickly. This is useful for simulations that intend to place a very large number of receivers. The cost of this speed is that the algorithm ignores the attenuation and line of sight models completely. For instance, a cell that is very far away from the receiver, and whose line of sight to the receiver is obstructed will be reduced by the same factor as a cell near the receiver with an unobstructed line of sight. Obviously this is not a faithful representation of the conceptual models discussed earlier, but supplied as a computationally-simple alternative.

3.4.2.2 Exact Suppression

The primary purpose of this algorithm was to discount exactly those transmissions which are likely to be observed by a placed receiver. This algorithm uses the ERT given by the evaluation algorithms (the user-specified evaluation algorithm is used in both Goodness and Suppression calculations) to determine the number of transmissions to discount a cell by. $ERT_{A,i,j}$ denotes the ERT in cell (i,j) observed from cell A. Exact Suppression works first by computing $ERT_{A,i,j}$ for all cells (i,j) within the Detection Area. Then, each corresponding cell in the Behavior Grid, $B_{i,j}$, is reduced by $ERT_{A,i,j}$. Finally, the goodness of all cells within Detection Range of a suppressed cell (all those that would have their goodness affected by the reduction) is recalculated. Figure **??** illustrates the Exact suppression algorithm.

3.5 Optimal Sensor Projection

In normal research situations, users will have a set number of receivers to place within their study site. The process of arriving at this number is likely unscientific, perhaps relying on user estimate (such as the user-perceived feasibility of receiving a given number of receivers). Rather than guessing at the number of receivers to use, and hoping for an adequate data recovery rate, users should be able to calculate the marginal benefit (additional detections, increased Data Recovery Rates) of utilizing a variable number of receivers. To this end, the program allows for the projected incremental benefit for a given number of additional receivers. Our program facilitates this by allowing users to specify a number of receivers to project, returning graphs and metrics of the marginal increase in Data Recovery Rate. Armed with this data, users can determine an appropriate number of receivers to purchase, or construct an argument for purchasing more receivers.

3.6 Outputs

Our program outputs 6 files:
1) The Bathymetry Grid as a heat map
2) The Behavior Grid as a heat map
3) The Goodness Grid as a heat map
4) The Coverage Grid as a heat map
5) The marginal gain in Unique Recovery Rate as a function of the number of receivers placed
6) Text representations of the 5 Grid files, tabular receiver data, and the specified input parameters.

3.6.1 Grid Graphs

The four Grid files (Bathymetry, Behavior, Coverage, and Goodness) are visualized as a heat maps. All of these are overlaid with the resulting receiver locations as numbered circles. Receiver locations show their detection range as dotted outlines centered over the receiver locations. User placed receivers are colored white, while receivers placed optimally by the system are colored blue. The numbering on user and system-placed receivers denotes the rank of each receiver's Unique Data Recovery Rate. User and system-placed receivers are ranked separately. Optimal receivers consist both the optimally placed and projected receivers. The highest ranked receiver locations are returned as the optimal receiver locations, with the lower ranked locations as the projected receiver locations.

3.6.2 Data Recovery Graphs

The program also produces a graph of the marginal increase in and cumulative sum of the Unique Data Recovery Rate as a function of the number of optimally placed receivers used (Figure 3.9). The graph of the marginal increase in UDRR (the lower graph) is especially useful. Given that Receivers have a fixed per-unit cost, it makes sense to weigh a receiver's effectiveness (perhaps UDRR contribution) against that cost to determine the utility of purchasing an additional receiver. The graph of cumulative UDRR is useful to quickly identify the number of receivers necessary to reach a particular UDRR.

3.6.3 Text Files

The program returns a comprehensive text representation of the program output (text dump of gridded data and input parameters), and a short, human-readable document that lists the primary simulation parameters (Evaluation Algorithm, Suppression Algorithm, Behavioral Model, Input Grid Size, and Detection Range), as well as a tabular output of receiver placements (coordinates), data recovery rates for each receiver, and the sparsity of the network. Projected receivers are excluded from the total UDRR, ADRR, and sparsity values.

Coordinates are returned in both a Global (with respect to the original Topography file) and local (the user specified area of interest) frame. While the curvature of the earth is well documented, different bathymetric maps may handle the mapping of a 3D curved plane to a 2D grid differently. For example, one grid may implement some scaling on grid cells a function of the cell's latitude or distance from a certain point. Other grid files may simply provide non-square grid files. By proving a small-scale (local) and large-scale (global) frame of reference for our receiver locations, irregularities of this nature are more easily detected.

3.7 Time and Space Complexity

To evaluate the temporal and spatial complexities of various elements of our program, we define the following variable inputs to the program:
n - The square root of the number of cells in the Bathymetry Grid, (the edge length of a square grid). Also recall that the Bathymetry, Behavior, Goodness, and Coverage grids are all of identical dimension, so n roughly gives the edge dimensions of any of the afore mentioned Grids.
D_{range} - The Detection Range. The square root of the number of cells in the Detection Grid (which is guaranteed to be a square).
k - The number of optimal placements to find.

3.7.1 Bathymetry Grid

The bathymetry data for the study site is provided as a grid of n^2 cells. This data must be copied into local memory (RAM), each cell in the grid takes O(1) time to copy, and O(1) space to hold. Thus, the creation of the Bathymetry Grid will take O(1) * n^2 = O(n^2) time and space..

3.7.2 Behavior Grid

Animal residency is computed as a function of the depth of a particular cell (Restricted Vertical Habitat), and the cell's location (OU/RW modeling). This computation takes a constant [O(1)] amount of time. The space required to store a single residency value for each of n^2 cells is also O(1). Therefore, the population of the Behavior Grid takes O(1) * n^2 = O(n^2) time and space.

3.7.3 Line of Sight Computation

As discussed in Section 3.3.3, the LoS algorithm is given as:
1) Determine the intervening cells $C_{0...m}$ between p and q
2) Compute the slopes between p and C_i for all i in $[0...m]$
3) Choose the Critical Slope
4) Project a line from p to q along the Critical Slope to find D_{max}.

Step 1 finds intervening cells by ray tracing, which can be done in linear ($O(n)$) time. The number of cells considered (n), depends upon the distance between the receiver and the target cell[15]. Given that the Detection Grid has a radius of D_{range}, and that receivers are located in the center of the Detection Grid, the greatest distance between a receiver and a target cell in the Detection Grid occurs between a receiver and the corner cells of the Detection Grid (a distance of $\sqrt{2} * \frac{D_{range}}{2}$). Thus, the number of cells considered when creating a list of intervening cells is $O(\sqrt{2} * \frac{D_{range}}{2}) = O(D_{range})$.

Step 2 computes the slopes for each of the m cells in C. Computing the slope between two points takes $O(1)$ time to compute and space to store. Therefore computing m slopes takes $O(1) * O(m) = O(m)$ time and space. As stated above, m is at worst $O(D_{range})$ on the Detection Grid. Thus, the slope computation is at worst $O(D_{range})$, requiring $O(D_{range})$ extra storage.

Step 3 chooses the largest slope amongst all m slopes. Therefore, we must consider m items, each requiring $O(1)$ time to consider as the largest. Since, m is $O(D_{range})$ on the Detection Grid, this step takes $(O(D_{range}))$ time to compute, and $O(1)$ space to store.

Step 4 is a direct computation, requiring $O(1)$ time to compute and $O(1)$ space to store the result.

In total, the LoS computation time is $O(D_{range}) + O(1) + O(D_{range}) + O(1) = O(D_{range})$, requiring $O(D_{range}) + O(D_{range}) + O(1) + O(1) = O(D_{range})$ temporary storage space.

3.7.4 Goodness Grid

Recall that the population of the Goodness Grid is performed by Evaluation Algorithms (Section 3.3.1), each of which computes ERT values for each of the D^2_{range} cells in the Detection Grid. These ERT values are then summed (requiring $O(D^2_{range})$ time) to provide the Goodness value for a single cell of the n^2 cells in the Goodness Grid. Thus, if we define the computational complexity of the ERT computation for Evaluation Algorithm x to be E_x, the computational complexity for populating the entire Goodness Grid is is the time for ERT computation plus the time for $O(n^2 * [D^2_{range} * E_x + D^2_{range}]) = O(n^2 * D^2_{range} * [E_x + 1]) = O(n^2 * D^2_{range} * E_x)$.

3.7.4.1 Evaluation Algorithm 1

Section 3.3.2.1 gives the ERT computation for Evaluation Algorithm 1 as:
$ERT_{1,i,j} = B_{i,j} * P_{Attenuation}(Range(i,j))$

Accessing the value $B_{i,j}$, finding the Range from i to j, and computing the attenuation due to range ($P_{Attenuation}(Range(i,j))$) and finding the product of the two can each be done in $O(1)$ time. Thus, the ERT computation for this algorithm is $O(1)+O(1)+O(1) = O(1)$. This computation will require $O(1)$ additional pieces of temporary storage for each intermediary value and the resulting product.

3.7.4.2 Evaluation Algorithm 2 & 3

As discussed in Section 3.3.2.2, the ERT computation for Evaluation Algorithms 2 and 3 is given as:
$ERT_{2,i,j} = B_{i,j} * P_{observation} * P_{Attenuation}(Range(i,j))$

As mentioned in Section 3.7.4.1, the product of $B_{i,j}$ and $P_{Attenuation}(Range(i,j))$ can be computed in $O(1)$ time with $O(1)$ temporary storage. Computing the observable number of transmissions ($P_{observation}$) requires substantially more work. Recall that we are using a 2-dimensional grid as the basis of our simulation and computing the third dimension as necessary by using a shape function. Therefore, we first determine the deepest depth in each cell that can be seen by the receiver, and then use integration to compute the number of detectable transmissions ($P_{observation}$) within the visible part of the water column above that cell. This leads us to the problem of finding the greatest depth (in each 2D cell) which is visible from the receiver. We compute this via incremental ray tracing which can be done in $O(D_{range})$ time.[15]. Then, using integration over the shape function, and known min/max visible depths, the total ($P_{observation}$) can be computed in $O(1)$ time. Thus, the ERT computational complexity for this algorithm is $O(1) + O(1) + O(D_{range}) + O(1) = O(D_{range})$. The algorithm will need to use $O(D_{range})$ temporary storage to compute Line of Sight, and $O(1)$ temporary space to store the resulting value. Thus the total temporary storage required is given by $O(D_{range} + 1) = O(D_{range})$.

3.7.5 Optimal Receiver Placement

The selection of the top R_{opt} receivers requires the Goodness Grid to have been populated. As previously stated, the selection process is iterative (requiring that suppression (if selected) be applied after each sensor selection), as suppression causes goodness values to be altered. The process of selecting a single receiver location is $O(n^2)$ since the algorithm must consider approximately n^2 possible receiver locations at each iteration. After each location selection, suppression must be applied to the chosen location. Due to the variability of the suppression algorithm, we use a variable ($E_{suppression}$) to represent the expected runtime of the suppression algorithm. Thus the computation time required for each selection-suppression step is $O(n^2 + E_{suppression})$. The number of iterations required to run is given by the number of optimal (R_{opt}) and projected (R_{proj}) receiver placements requested. In total, the runtime of the Optimal Receiver Placement step is $[R_{opt} + R_{proj}] * [O(n^2 + E_{suppression})] = O([R_{opt} + R_{proj}] * [n^2 + E_{suppression}])$.

3.7.6 Suppression

As discussed in Section 3.4, the suppression mechanic is applied after the placement of a receiver. The suppression mechanic has several options, each affecting the time and space complexity of the mechanism. To better capture the complexity of the algorithms involved, we denote the following variables:

D_{range} - The Detection Range of a receiver.
$d_{detection}$ - The edge size of the Detection Grid, equal to $2r + 1$ cells.
f_{supp} - The Suppression Range Factor. A positive real number.
r_{supp} - The Suppression Range. Equal to $f_{supp} * D_{range}$

d_{supp} - The square root of the number of cells in the Suppression Area. The edge dimension of the Suppression Area. Equal to $2 * r_{supp} + 1$ cells.

3.7.6.1 Static Suppression Algorithm

As stated in Section 3.4.2.1, the Static Suppression Algorithm multiplies all cells within the Suppression Area by a scalar constant. This is a scalar multiplication that can be done in place, requiring $O(1)$ extra space, but requires a multiplication operation for each cell in the Suppression Area, which is $O(d_{supp}^2) = O([2*f_{supp}*D_{range}+1]^2) = O([4*f_{supp}^2*D_{range}^2+4*f_{supp}*D_{range}+1]) = O(f_{supp}^2 * D_{range}^2)$. In most cases (excluding those where very sparse sensor networks are desired), the Suppression Range Factor, f_{supp}, will be rather small (less than 2.0), which reduces the runtime complexity to $O(D_{range}^2)$.

3.7.6.2 Exact Suppression

As stated in Section 3.4.2.2, the Exact Suppression Algorithm uses the user-specified Evaluation Algorithm to calculate the number of transmissions that have already been observed and should be discounted from future consideration. We define E_x to denote the expected runtime for Evaluation Algorithm x's computation of a single cell's ERT. We also define T_x to denote the expected temporary storage requirement for Evaluation Algorithm x single-cell ERT computation. The Exact Suppression Algorithm is broken into three distinct steps: Suppression Area ERT computation, Behavior Grid Updtate, and Goodness Grid Recalculation.

Suppression Area ERT Computation

The suppression algorithm first determines the ERT for each cell within the Suppression Area, and stores them in the ERT Area, a temporary Grid with the same dimension as the Suppression Area. This step is similar to the Evaluation process, with the exception that the Suppression Area's dimensions differ from those of the Detection Grid by a factor of f_{supp} (the Suppression Area is usually larger than the Detection Grid). The computation of the ERT Area requires E_x time and T_x temporary space for each of the d_{supp}^2 cells in the Suppression Area. An additional $O(1)$ temporary space will also be needed to store the resulting ERT values. Therefore, the ERT computation will require $O(E_x * d_{supp}^2)$ time and $O((1 + T_x) * d_{supp}^2) = O(d_{supp}^2 * T_x)$ temporary storage.

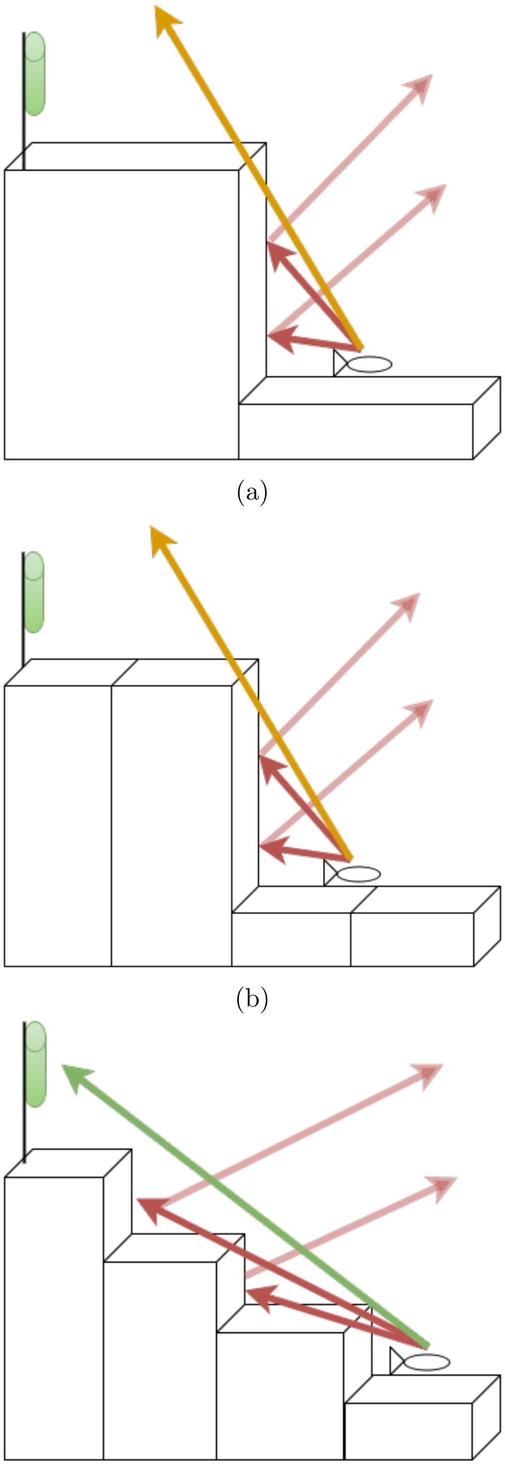
Behavior Grid Update

Next, the suppression algorithm subtracts the ERT Area from the corresponding area in the Behavior Grid. This is a simple subtraction operation over each corresponding cell in the grids. Because each subtraction takes $O(1)$ time and temporary storage, the time and space complexities are both given by $O(1) * O(d_{supp}^2) = O(d_{supp}^2)$.

Goodness Grid Recalculation

Finally, the goodness values of all cells within Detection Range of the Suppression Area must have their Goodness value updated (as we have just reduced the ERT of one or more cells within their Detection Grid). The area affected by suppression, and therefore requiring recalculation, is a square with edge diameter $2 * (d_{supp} + D_{range}) + 1$. Each cell in this area will require $O(D_{range}^2 * E_x)$ time to compute. The total time for updating the Goodness Grid is then: $O([2 * (d_{supp} + D_{range}) + 1]^2 * O(D_{range}^2 * E_x)) = O((d_{supp} + D_{range})^2 + D_{range}^2 * E_x)$. The Goodness Grid update will require $D_{range}^2 * (T_x + 1) = O(D_{range}^2 * T_x)$ temporary storage to compute and store intermediate ERT values. This temporary storage can be recycled for each sequential Goodness Cell computation.

The total computation time for the Exact Suppression Algorithm is then given by O(ERT Computation) + O(Behavior Grid Update) + O(Goodness Grid Recalculation). This is a total of $O(E_x * d_{supp}^2) + O(d_{supp}^2) + O((d_{supp} + D_{range})^2 + D_{range}^2 * E_x)$ time. The algorithm will also require $O(MAX(d_{supp}^2, D_{range}^2) * T_x)$ temporary storage, which can be recycled between each of the above stages.



(a)

(b)

(c)

Figure 3.1: Figure 3.1a illustrates the bathymetric shadowing model for two adjacent cells within a bathymetric grid. Figure 3.1b shows how artifically increasing the resolution of the bathymetric grid form Figure 3.1a using the duplication method of cell sub-division does not affect the bathymetric shadowing model. Figure 3.1c shows how artificially increasing the resolution using a smoothing function can lead to inflated signal reception.

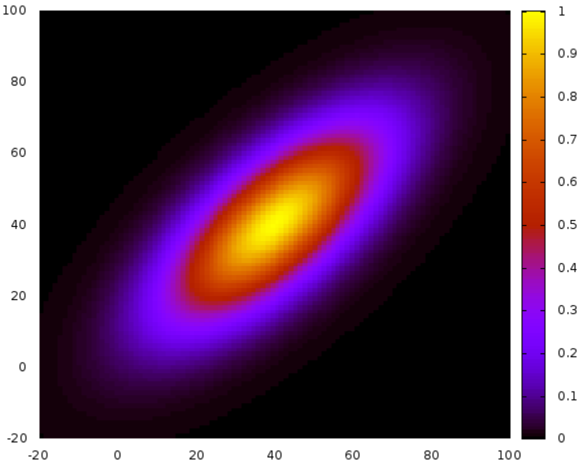


Figure 3.2

Figure 3.3: An example of a distribution given by the Ornstein-Uhlenbeck Model with a high attraction value in the x-direction, a low attraction value in the y-direction, and a correlation value of 0.7.

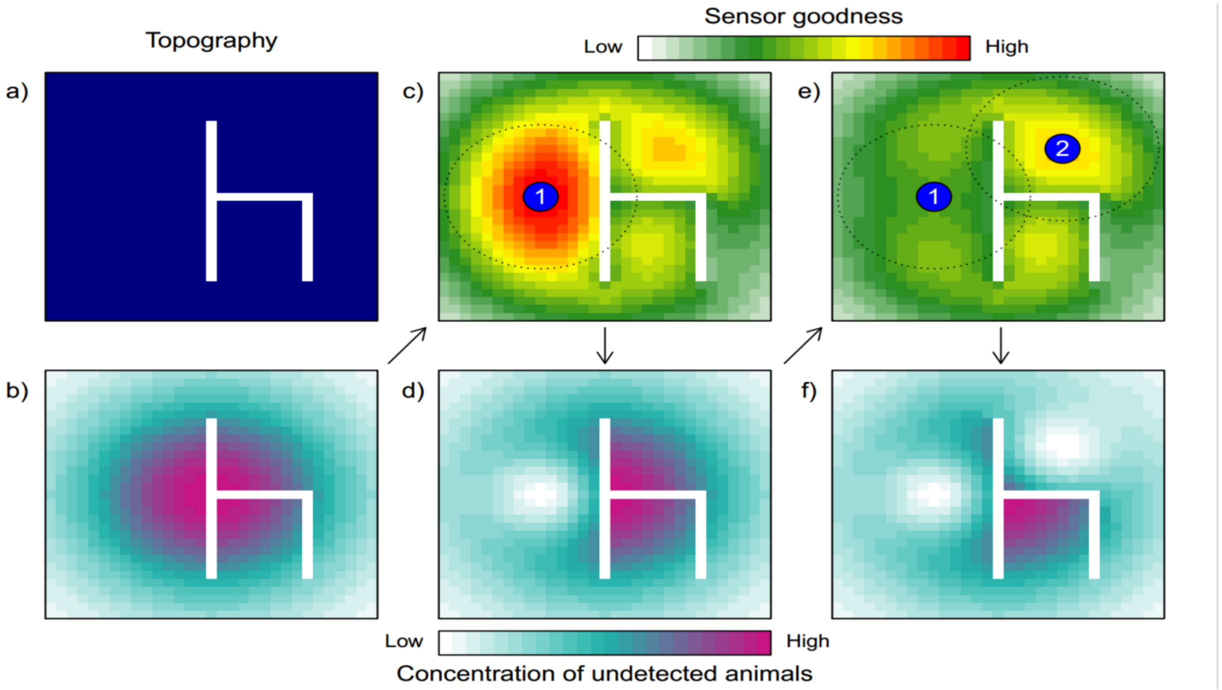


Figure 3.4: An illustration of the Exact Suppression Algorithm. In (a), we see a bathymetry grid with infinitely high walls (the white 'h' shape) on an otherwise flat plane (blue region). In (b), we see Behavior Grid with a distribution (given by the OU movement model) of animals around the walls. (c) shows the computed goodness of Sensor (receiver) locations within the study site. The program first identifies location 1 (the blue circle) as having the highest unique data recovery rate, and places a receiver there. The dotted lines represent the receiver's Detection Range. In (d), the program suppresses the Behavior Grid by subtracting the ERT of each cell within Suppression Range. Here the Suppression Range Factor is 1.0, so the Suppression Area is the same as the Detection Area). In (e), the goodness grid is recalculated, taking into account the suppressed Behavior Grid. Additionally, the program identifies location 2 as having the highest Unique Data Recovery Rate. In (f), the Behavior Grid is again suppressed to account for the placement of receiver (2).

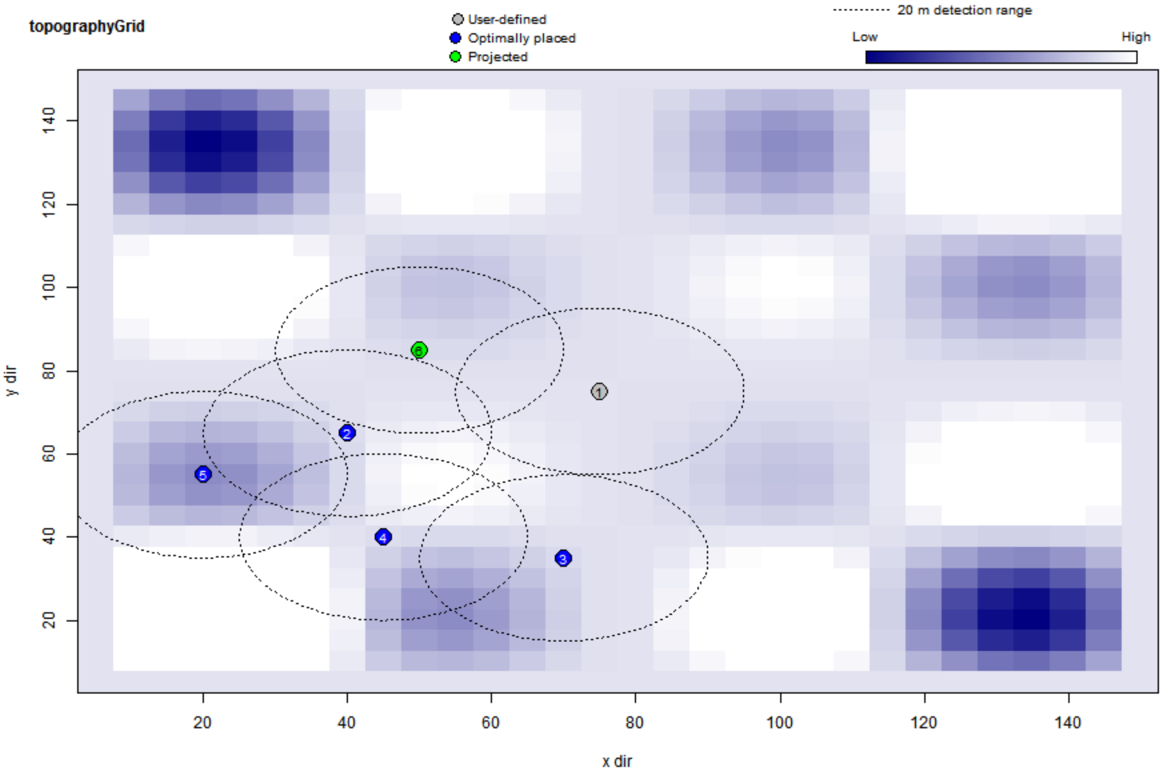


Figure 3.5: A graphical representation of an artificial Bathymetry file at a 1m resolution (each cell has an edge length of 1m). Darker cells represent greater depths, while white cells represent inaccessibel terrain (dry land). The optimal receiver locations are shown on the grid as blue numbered circles, user-placed sensors as grey circles, and projected receivers as green circles. All receivers have their Detection Range shown as dotted lines.

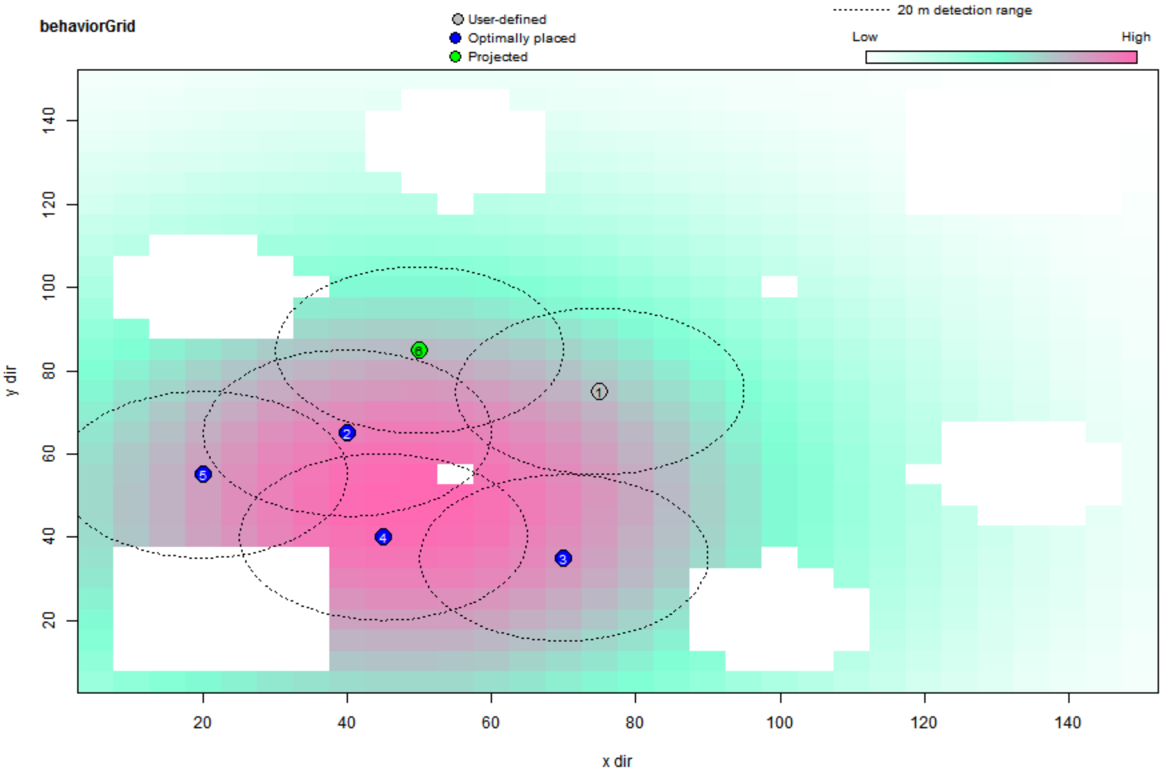


Figure 3.6: The Behavior Grid represented as a heatmap. Higher levels of animal residency correspond with pink cells, moderate levels as light blue, and white for non-residency (ihospitible habitat such as dry land). The optimal receiver locations are shown on the grid as blue numbered circles, user-placed sensors as grey circles, and projected receivers as green circles. All receivers have their Detection Range shown as dotted lines.

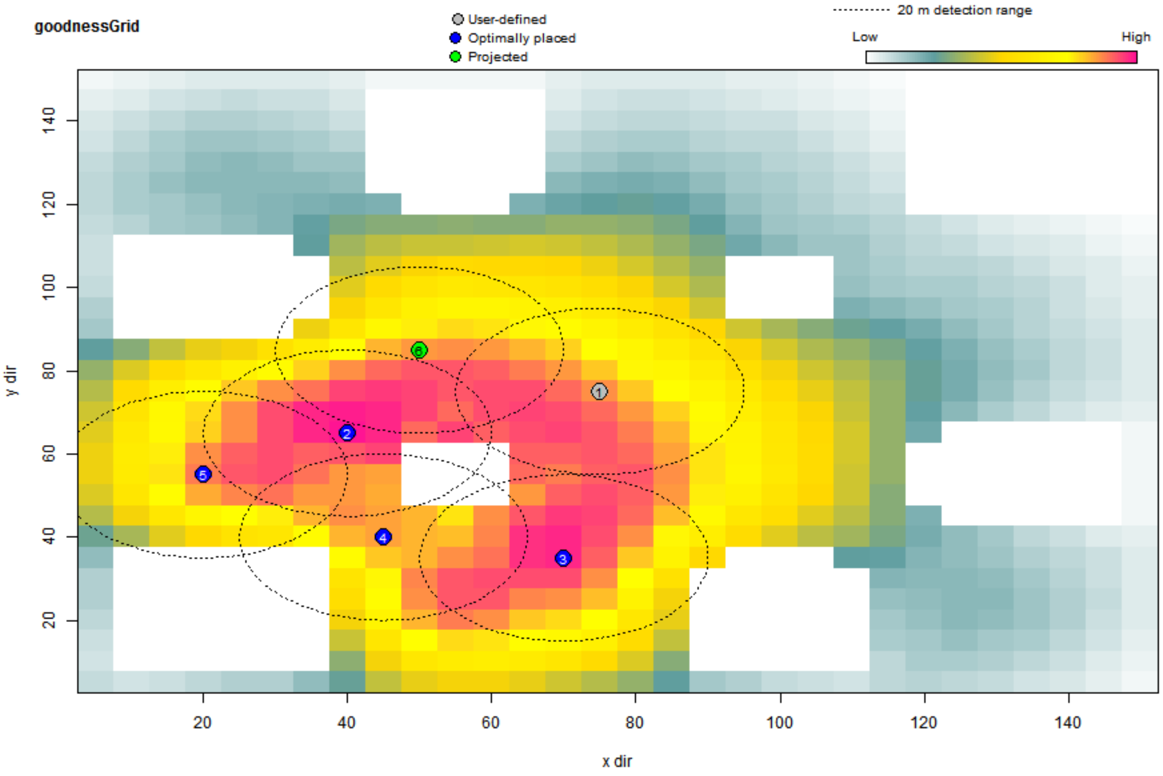


Figure 3.7: The Goodness Grid represented as a heatmap showing the quantity of Estimated Receivable Transmissions (ERT) if a receiver were placed in a particular cell. The legend in the top right assigns color coding to various ERT values. The optimal receiver locations are shown on the grid as blue numbered circles, user-placed sensors as grey circles, and projected receivers as green circles. All receivers have their Detection Range shown as dotted lines.

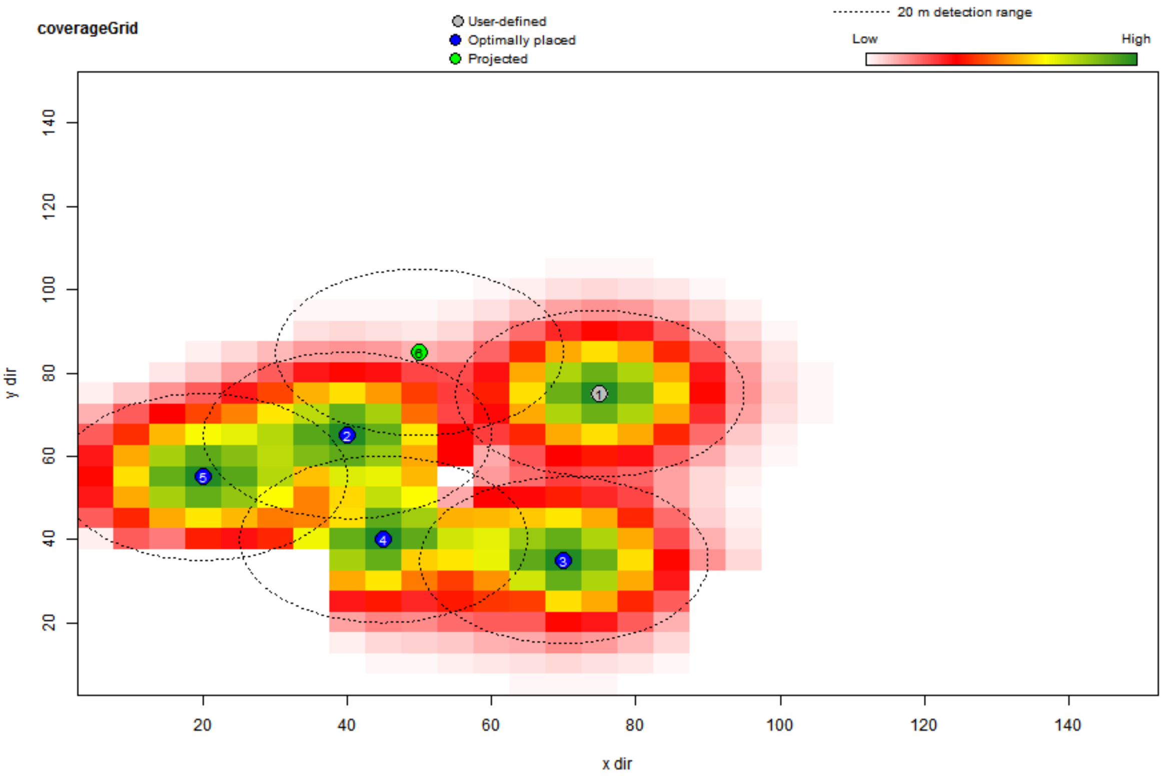


Figure 3.8: The Coverage Grid represetned as a heatmap showing the quantity of Estimated Receivable Transmissions from each cell in the study site, for the designated reciver array configuration. The legend in the top right assigns color coding to various ERT values. The optimal receiver locations are shown on the grid as blue numbered circles, user-placed sensors as grey circles, and projected receivers as green circles. All receivers have their Detection Range shown as dotted lines. The missing corner of of Receiver 4's Detection Area is due to the presence of an obscuring section of Bathymetry (dry land).

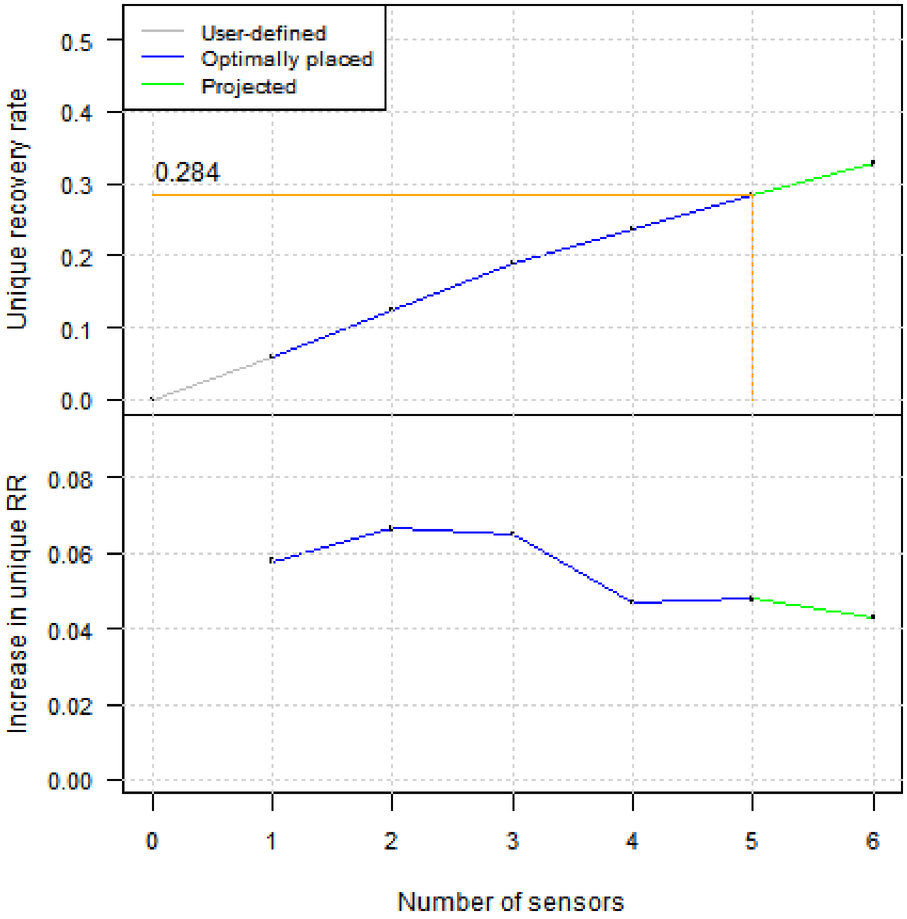


Figure 3.9

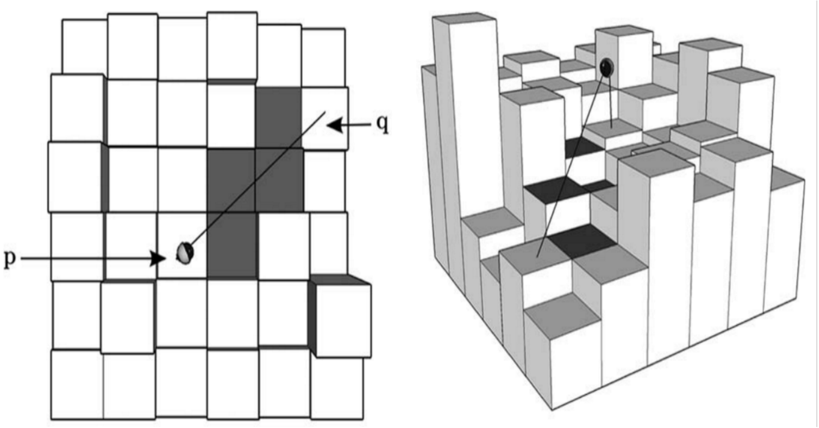


Figure 3.10: An illustration of how ray tracing works within a 3D environment. [6]

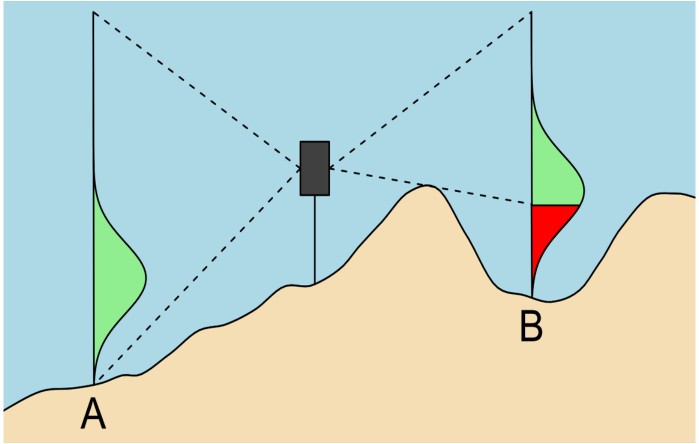


Figure 3.11: An illustration of how ray tracing and integration over a shape function can be used to compute the probability of detecting fish within a cell. Dotted lines indicate the maximum and minimum depths visible to the receiver. The normal distributions in green/red indicate the distribution of fish within a given cell as determined by a shape function. The green portion of the distribution indicates the portion of the distribution that is observable by the receiver and the red indicates the portion that is unobservable by the receiver. The observable distribution (green) is computed by integration over the shape function.

CHAPTER 4
RESULTS

Here is where you discuss the results from your evaluation.

CHAPTER 5
CONCLUSIONS

Here is where you discuss your conclusions and future directions.

APPENDIX A
TABLE OF NOTATIONS

A - Bathymetry Grid - A 2D grid of real numbers that describe the depth of a section of the sea floor. $A_{i,j}$ refers to a particular value in the cell at row i, column j.

B - Behavior Grid - A 2D grid of real numbers that indicate the percentage of all transmissions that are expected to be released from that cell over the entire experiment. $B_{i,j}$ refers to a particular value in the cell at row i, column j.

G - Goodness Grid - A 2D grid with the same dimensions as the Bathymetry Grid. Contains real numbers that indicate how good a particular cell will serve as a receiver placement. $G_{i,j}$ refers to a particular value in the cell at row i, column j.

$ERT_{x,A,i,j}$ - Estimated Receivable Transmissions - The number of transmissions that are expected to be received from a cell at row i, column j, relative to a receiver at cell A, according to Evaluation Algorithm x.

$P_{Attenuation}(x)$ - Probability of Detection due to attenuation - Describes the probability of detecting an acoustic transmission originating at a range of x meters away from a receiver.

$Range(i,j)$ - Distance from receiver to cell at (i,j) - Distance from a particular receiver location to the cell at row i, column j. Computed as the Euclidian distance between the center of a receiver-containing cell and the center of the cell at row i, column j on a 2D grid.

R_{opt} - The number of optimal receivers to be placed by the system. (Excluding projected receivers)

R_{proj} - The number of projected receivers to be placed by the system.

D_max

$m_{p,q}$

$$D_{range}$$

- The detection range of the receiver. Assumed to be the same for all receivers in the network. Given as the distance between a receiver and tag at which there is a 5% chance of detecting an acoustic transmission.

s - The square root of the total number of cells in the Bathymetry Grid (A). Approximates the edge dimension of a square Bathymetry Grid.

APPENDIX B
MORE ANCILLARY STUFF

Subsequent chapters are labeled with letters of the alphabet.

BIBLIOGRAPHY

[1] Animal migration tracking. <https://en.wikipedia.org/wiki/Animal_migration_tracking#Radio_tracking>.

[2] How argos works. <http://www.argos-system.org/web/en/337-how-argos-works.php>.

[3] Innovative technology/lab support proposal online form. <https://www.uaf.edu/tab/past-proposals/proposalDetails.xml?id=667>.

[4] Vhf systems for fish (fis). <http://www.telonics.com/products/vhfImplants/vhfFish.php>.

[5] Wildlife tracking. <http://www.wildlifetracking.org/faq.shtml>.

[6] Vahab Akbarzadeh, Christian Gagné, Marc Parizeau, Meysam Argany, and Mir Abolfazl Mostafavi. Probabilistic sensing model for sensor placement optimization based on line-of-sight coverage. *IEEE Transactions on Instrumentation and Measurement*, 62(2):293–303, 2013.

[7] Paul C Etter. *Professional Development Short Course On: Underwater Acoustic Modeling and Simulation.* 2004.

[8] Rey Farve. Technology and development at the usda forest service, satellite/gps telemetry for monitoring lesser prairie chickens. http://www.fs.fed.us/t-d/programs/im/satellite_gps_telemetry/wildlifetrackingtelementry.htm.

[9] Alan Frieze, Jon Kleinber, R Ravi, and Warren Debany. Line of Sight Networks. *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithmsProceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms, 968-977*, pages 968—-977, 2007.

[10] J Hansen and Michael Jones. perSpeCtIve : FISHERIES MANAGEMENT the value of Information in Fishery Management. 33(7), 2008.

[11] M. R. Heupel, J. M. Semmens, and a. J. Hobday. Automated acoustic tracking of aquatic animals: Scales, design and deployment of listening station arrays. 57(1):113, 2006.

[12] Andrew Howard, M.J. Mataric, and G.S. Sukhatme. Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem. *Proceedings of the 6th International Symposium on Distributed Autonomous Robotics Systems (DARS02)*, 5:299–308, 2002.

[13] Steven Thomas Kessel, Nigel Edward Hussey, Dale Mitchell Webber, Samuel Harvey Gruber, Joy Michelle Young, Malcolm John Smale, and Aaron Thomas Fisk. Close proximity detection interference with acoustic telemetry: the importance of considering tag power output in low ambient noise environments. *Animal Biotelemetry*, 3(1):1–14, 2015.

[14] Ross A. Maller, Gernot Müller, and Alex Szimayer. *Handbook of Financial Time Series*, chapter Ornstein–Uhlenbeck Processes and Extensions, pages 421–437. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

[15] James McNeill. Playtechs: Programming for fun: Raytracing on a grid, 2007.

[16] William D Pearse, Andy Purvis, Life Sciences, and Silwood Park Campus. Generation Tool for Ecologists. *Methods in Ecology and Evolution*, 4(10):920–929, 2013.

[17] Martin W. Pedersen and Kevin C. Weng. Estimating individual animal movement from observation networks. *Methods in Ecology and Evolution*, 4(10):920–929, 2013.

[18] S. Poduri and G.S. Sukhatme. Constrained coverage for mobile sensor networks. *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, 1, 2004.

[19] Anna E Steel, Julia H Coates, Alex R Hearn, and a Peter Klimley. Performance of an ultrasonic telemetry positioning system under varied environmental conditions. *Animal Biotelemetry*, 2(1):15, 2014.

[20] Acoustic Telemetry. Technical white paper understanding the performance of vemco 69 khz single frequency acoustic telemetry. 2008.

[21] Yuan Zhaohui, Tan Rui, Xing Guoliang, Lu Chenyang, Chen Yixin, and Wang Jianping. Fast sensor placement algorithms for fusion-based target detection. *Proceedings - Real-Time Systems Symposium*, pages 103–112, 2008.