Search for a car parked on a forest road

Midterm Report: March 25th, 2014

Math 485

Group Members:

Jayson Anthony Rickel, Alex Nguyen, Rachel Anne Braun, Mohammed Alabkari, Yu Gong

Mentor:

Gleb Zhelezov

Introduction

Mathematical modeling is a very useful and helpful tool for studying the search for a parked car. Imagine parking on the side of a forest road in order to go hiking. After hiking through the woods for a while, you emerge from the trees and are once again on the road. However, you are no longer near your car. You can assume that you know the probability density for the location of the car, since you didn't hike too far. What is the best way to go about finding your car? Through mathematical modeling, we can discover an optimal search pattern for finding the parked car.

In the traditional linear search problem, an object is placed on a line using a given probability density. Starting with an initial step length, the searcher goes back and forth using step lengths calculated from a specific formula until the object is found. To simplify our analysis, we consider the one-sided search problem (Figure 1). Applications of such a problem can include a simple robot that returns to the origin after each step to report whether it has found the car or not.

The overall goal of our project is the analysis of a linear search problem for an object placed using a given probability density function using analytical and numerical methods. Currently, we have developed numerical methods simulating the search for an object placed using an exponential distribution function and the minimization of the search for the object. Using Matlab, we have modeled an assortment of searches to determine an optimal sequence for our fixed parameters.

Through mathematical modeling, we can reduce the cost and time spent determining optimal paths that can later be applied in real-world situations. For example, we can test multiple probability distributions for the location of the car without physically setting up the parked car. In our project, mathematical models can help to predict the position of a car and further serve as guidelines for travelers and primary sources for researchers who want to observe the visualize aspects of each case on each experiment. For researchers, models provide an overview for a case that may not be practical in the real world. In summary, development of a model for the search for a parked car helps us to predict the car position, minimize the cost, and provide insight into real world behavior.

Mathematical Model

The following assumptions were made in our reference paper (Ref. 1):

The search problem is only on a one-sided gatherer version and the hidden object H, the car position, is located on the *half-line* $\mathbb{R}_+$. In our analysis, we consider the same one-sided searcher problem.

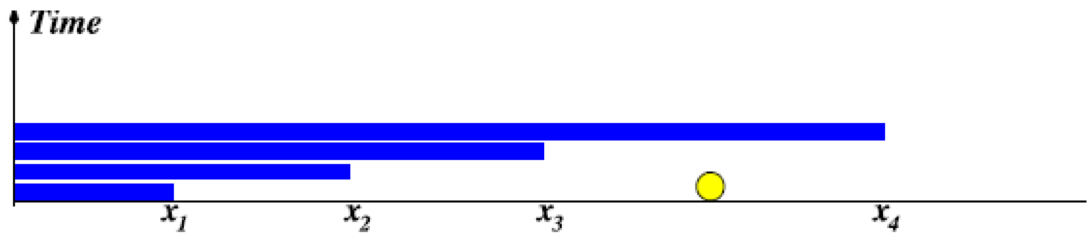

Figure 1. Diagram of one-sided gatherer movement (Ref 1.)

In the paper, the distribution function used was the homogeneous tail distribution function, also called a Pareto distribution, shown below.

$$x = \{0 = x_0 < x_1 < x_2 < \cdots < x_k < \cdots\}$$

However, in our analysis, we will use an exponential probability density for the car's location.

Using the Pareto distribution, the paper arrived at the following conclusion. The optimal plan should satisfy a two-term recurrence, the variational recursion.

$$f(x_{k-2}) + f'(x_{k-1})x_k = 0$$

$$x_k = \frac{f(x_{k-2})}{-f'(x_{k-1})}$$

$$E(x) = \mathbb{E}[L(x,H)] = \mathbb{E}\left[\sum_{k=1}^{n(x,H)} x_k\right]$$

However, in the theory section below, we analyze the problem from the perspective of a general given probability density function.

The following variables are used in our analysis:
$L(x,H)$: the total distance travelled until the point H is found, given as a function of x
$E(x) = \mathbb{E}[L(x,H)]$: the cost/expectation of the search plan as a function of the total distance travelled
F(x): chosen probability density function

f(x): cumulative distribution function for the chosen probability density function

Theory

We are attempting to minimize the expectation for the cost function for the total length of the search ($EL = \mathbb{E}[L(x, H)]$). We define the expectation of a search of length L as follows, where F(x) is the probability density:

$$EL = \int_0^\infty F(x)L(x)dx \qquad \text{Eq. 1}$$

For eq. 1, we define L(x) as follows:

$$L(x) = \begin{cases} 2x_1 & 0 < x \leq x_1 \\ 2(x_1 + x_2) & x_1 < x \leq x_2 \\ \quad \dots & \\ 2\sum_{i=1}^{N} x_i & x_{N-1} < x \leq x_N \end{cases} \qquad \text{Eq. 2}$$

Substituting into Eq. 1 our definition of L(x) shown in Eq.2, we are left with the following equality:

$$EL = \int_0^{x_1} 2x_1F(x)dx + \int_{x_1}^{x_2} 2(x_1 + x_2)F(x)dx + \dots + \int_{x_{N-1}}^{x_N} 2\sum_{i=1}^{N} x_i \; F(x)dx \qquad \text{Eq.3}$$

In order to simplify Eq. 3 we will define the cumulative distribution function (CDF) for our given probability density F(x):

$$\text{CDF: } f(x) = \int_0^x F(x)dx \qquad \text{Eq. 4}$$

Using the CDF from Eq. 4 to evaluate the integrals from Eq. 3:

$$EL = 2x_1[f(x_1) - f(x_0)] + 2x_1[f(x_2) - f(x_1)] + 2x_2[f(x_2) - f(x_1)] + \dots. \qquad \text{Eq. 5}$$

After some algebraic computations, we can re-write EL in the following form:

$$EL = 2\left[\sum_{n=1}^{\infty} x_n - \sum_{m=2}^{\infty} x_m f(x_{m-1}) \right] \qquad \text{Eq. 6}$$

In our analysis of the search problem, we will use an exponential probability density function for the location of the parked car. However, we are unable to find an analytical solution for EL in Eq. 6 when the exponential probability density function is used. Therefore, we instead use numerical methods and computer simulations to determine the expectation length.

Numerical Methods and Computer Simulations

We know that the infinite sum that defines the expectation of the length is infinite; however, we don't have an easy way of analytically computing the values of the first two step points which would recursively solve the entire pattern. This means we have to go to computer simulations in order to solve this problem; more specifically, we need to use the Monte Carlo Method. The Monte Carlo Method is a type of simulation that is used when there is a random event occurring in the simulation. For example, one can roll two, six-sided die over and over again and tally the results in order to find the probability of an individual roll begin rolled. In our case, our random event is the probability of the car being placed along the forest road. We need to run many (~one million) simulations of placing the car and finding the length to get to the car in order to optimize the step points that we want to use. We will pick two points to begin with, run the simulations, average the length to get to the car, move the points slightly, and continue this process until we find two step points that give us the minimum length to the car.

Before we do this, we want to see what sort of data the computer is dealing with to give us some insight into the behavior of this problem. We can run the simulation many times and create a histogram in order to see the curvature of the lengths that we receive by mixing the continuous probability density function with the piecewise function that represents the lengths for each car placement. In the paper, the probability density used is a parabolic density with exponent -α. This density gives an optimal pattern that is a geometric series. The density that we will be using is an exponential function with λ = 1 to simplify this function. However, unlike the parabolic equation, this density function will not give a perfectly geometric series of steps. We still know that the geometric series will give us a very good result, so we will be using this pattern to simplify these preliminary simulations. The geometric series will be represented in the following form: $x_n = \Delta\alpha^n$. The following figure shows us what the frequency of different lengths are when we set α = 1.10 and Δ = 0.30. This gives a nice curve that shows the mix of the geometric series that is seen to dominate the graph early and the exponential probability curve that creates the tailing off as the values of the steps get large:

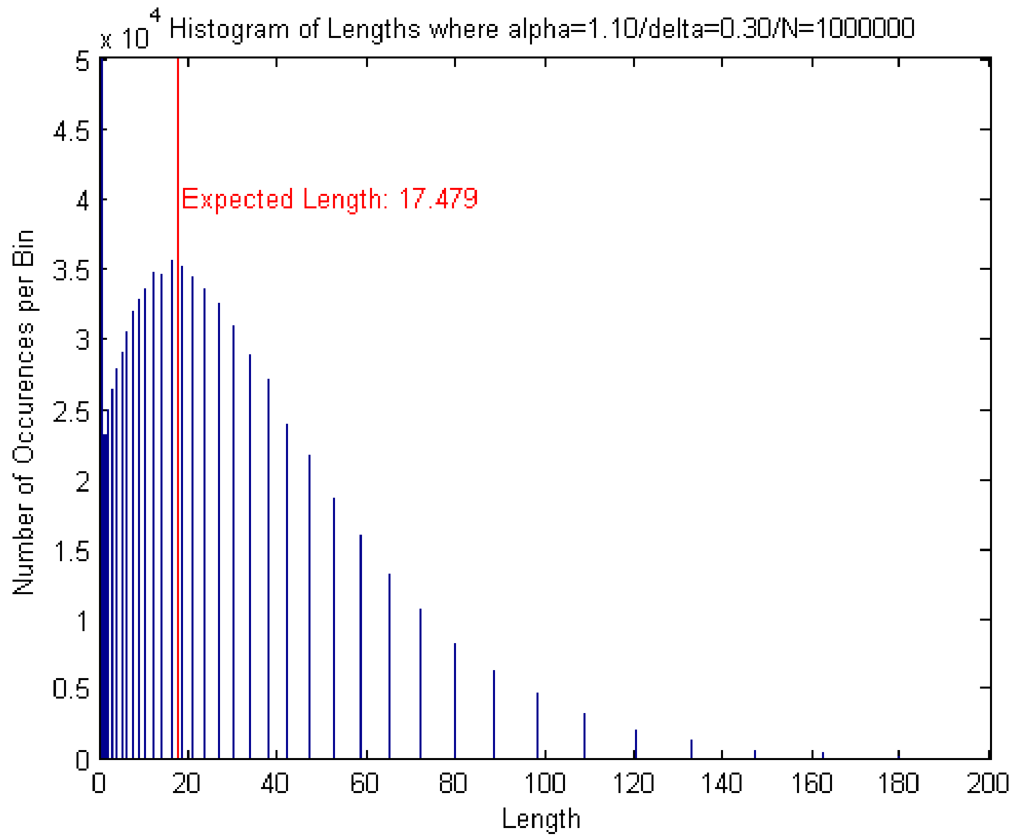

FIGURE 2. Histogram of Different Car Placements

Now, we need to find out what the optimal α and Δ should be for this exponential density function. In order to do this, we need to pick α and Δ values in a certain range in order to zero in on the correct pair. For each pair, we will run a large number of simulations to see what value gives the minimum expected length for finding the car. The graph below shows how this data becomes quite accurate when large sample sizes are taken. When only 1000 simulations are run, we see a very piecewise and choppy graph, but when the sample size is closer to one million, we get a nice graph that helps to verify our results. In the following graph, the different curves represent different α values and the x-axis denotes different Δ values. The expected lengths are shown on the y-axis:

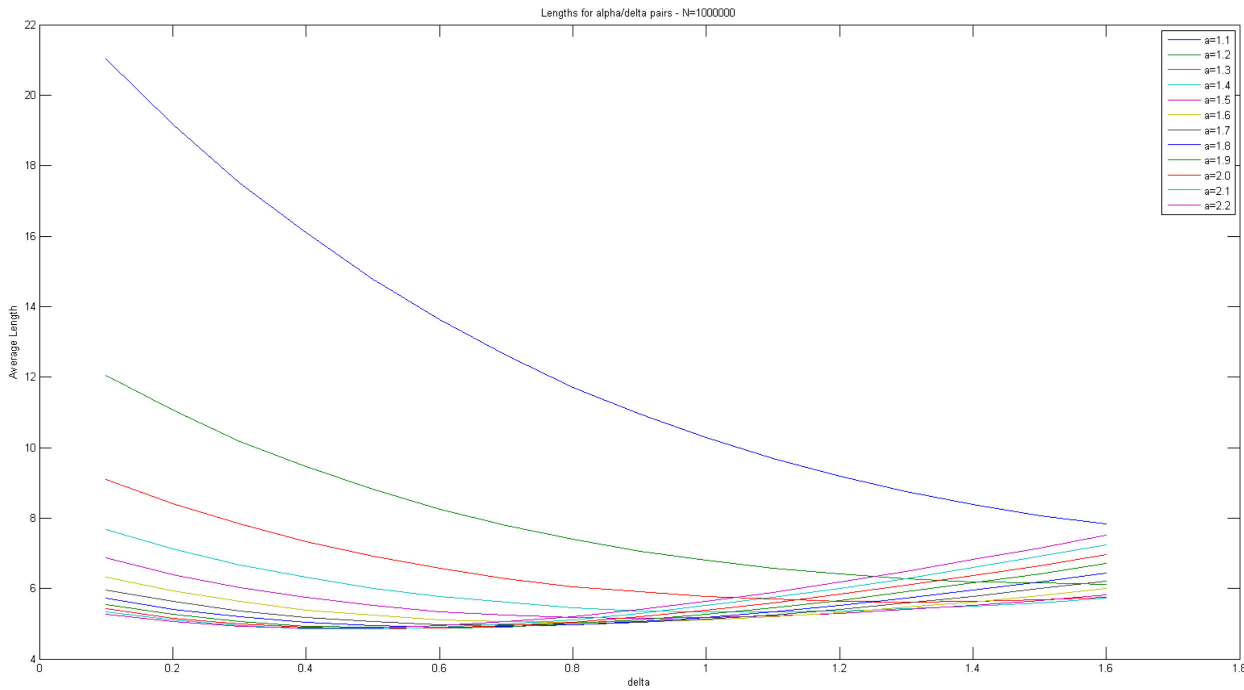

FIGURE 3. Optimization of Geometric Sequence

As we can see, the values tend to converge close to α = 2.1 and Δ = 0.5 with a length of 4.7932. From here, we can choose smaller ranges around these values to get a more accurate result, but we are just using this graph to show how to get these optimal values, so we will not be taking this step here. This graph is a great representation of how the Monte Carlo Method can get us very good results when using a high sample size, and we can see that all of these curves appear to be differentiable. From here, we can use a similar method using two pivots instead of an α and Δ value in order to find the steps that need to be taken to minimize the length for an exponential distribution as opposed to a parabolic one.

Conclusion

Through our current work, we are able to develop numerical methods for finding the expectation values given a probability density for the car location using various step lengths. From our preliminary simulations, we know that the expectation length converges. However, we are unable to find this value analytically. For future tests, we will continue to use numerical methods and Monte Carlo simulations to find the expectation length.

Future Work

In order to simulate a problem closer to real-world situations, we will add additional information to the formulation of each model. For example, we are going to consider the problem where someone gets into the forest, wanders for a while, returns to the road, and starts the search for the car. To model the person wandering in the forest, we are going to consider this to be a diffusion process in time. The process ends when the equation used for the diffusion process has the value of reaching the road (Figure 4). Another option is to consider a random walk in two dimension that stops when the x-axis (i.e. the road) is crossed/reached.

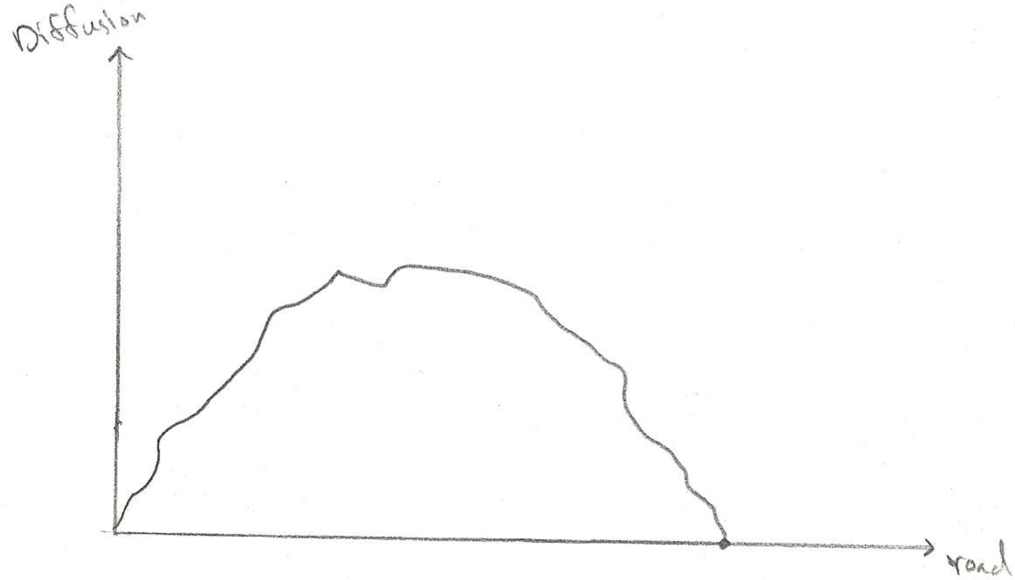

Figure 4. Example of diffusion through forest

Another consideration for future work is to model the search for the car using a robot in a more realistic fashion. To accomplish this, we are going to use a random step size that is generated according to some probability density function (such as exponential, normal, etc.). The search process will start according to a geometric series and then the random step size will continue the process in a progressive way. In other words, the robot will progressively walk as in the figure below until it finds the car and then return in a regressive way according to the inverse geometric series to the person.

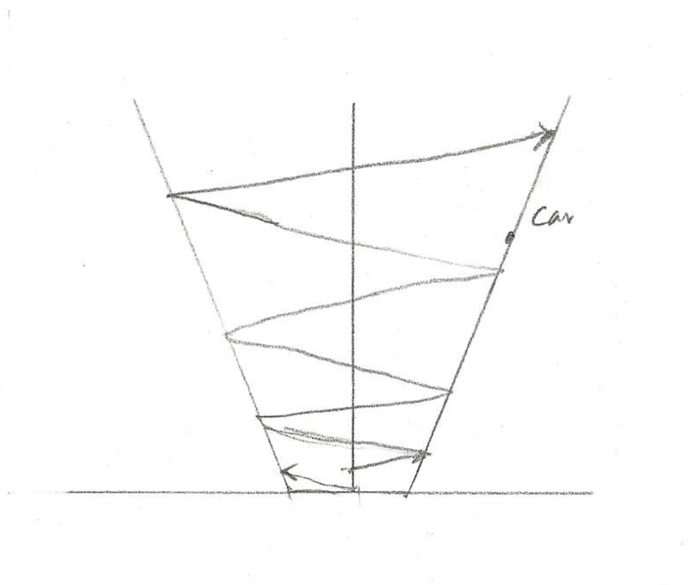

Figure 5. Example of possible search pattern

We are going to find the expectation value for this search process analytically by finding and solving the following equation:

$$E\big(L(x)\big) = E(\propto) + E(\frac{1}{\propto})$$

Where $E(\propto)$ and $E(\frac{1}{\propto})$ are the expectation of the progressive and regressive processes, respectively.

For each new situation, we are going to do Monte Carlo simulations in Matlab to obtain results and determine an optimal path. Furthermore, we will use Monte Carlo simulations to get the results for the diffusing process and find the expectation value of the search problem.

Reference

1. Yu Baryshnikov and V Zharnitsky,
   Search on the brink of chaos, Nonlinearity, 25 (2012), 3023–3047,
   doi:10.1088/0951-7715/25/11/3023