1-1-2013

Analyzing Environmental Change And Prehistoric Hunter Behavior Through A 3d Time-Lapsed Model With Level Auto-Generation And Cultural Algorithms

Samuel Dustin Stanley
Wayne State University,

ANALYZING ENVIRONMENTAL CHANGE AND PREHISTORIC HUNTER BEHAVIOR THROUGH A 3D TIME-LAPSED MODEL WITH LEVEL AUTO-GENERATION AND CULTURAL ALGORITHMS

by

SAMUEL DUSTIN STANLEY

THESIS

Submitted to the Graduate School

of Wayne State University,

Detroit, Michigan

in partial fulfillment of the requirements

for the degree of

MASTER OF SCIENCE

2013

MAJOR: COMPUTER SCIENCE

Approved by:

____

Advisor Date

DEDICATION

I would like to dedicate this work to Gerald Keith Larsen, who was perhaps the most intelligent individual I have ever had the pleasure to meet in my life. Gerald regrettably died from cancer in February 2013, but I will never forget him as long as I live.

ACKNOWLEDGMENTS

I would like to acknowledge my advisor Dr. Robert Reynolds. I also would like to acknowledge the others in my research team without which this work would not have been possible: Tiffany McLean, Thomas Palazzolo, and Gerald Larsen.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

CHAPTER 1: INTRODUCTION

This paper details a system that researchers can use to design and implement experiments involving the Alpena-Amberley ridge and similar environments. Given data describing the rate of change of various factors within an environment, the system provides automatic design of the changed environment for all desired intervals within a given time period. This allows researchers to design and perform experiments where environmental change is a crucial factor without having to manually redesign the environment whenever it changes. Additionally, many helpful tools and functionalities for designing and implementing experiments are provided, such as the ability to easily move forward or backward one or multiple intervals and navigate through time in other ways. Various intelligent agents pre-loaded with AI algorithms are also included (i.e., herds of AI caribou, as well as hunting blinds whose placement is governed by cultural algorithms).

1.1 Platform

The system was created using Thomas Palazzolo's "Land Bridge GUI" program, developed using Microsoft Visual C# 2008 and Microsoft XNA 3.1, as a basis. This is a program created by Palazzolo to model the Alpena-Amberley Land Bridge, a land formation currently under Lake Huron that was above lake level for most of the time between about 9793 BCE to about 6343 BCE.

1.2 Overview of this Paper

The rest of this chapter (Chapter 1) provides a very general overview of our entire system, including functionalities that won't be used in our featured experiment *per* se, but are nonetheless useful to researchers in implementing their own tests and experiments with the system. Chapter 2 discusses the previous work that has been done concerning the Alpena-Amberley ridge. Early work by geologists, Dr. John O'Shea's seminal work, and simulation systems created by computer scientists since then are all discussed. Chapter 3 goes into further detail about the Alpena-Amberley Land Bridge itself and discusses the purpose and objectives of our simulation system. Chapter 4 provides the high and low level design of our cultural algorithm system, whose aim it is to provide tools to help the process of finding hunting blinds and other artifacts under Lake Huron. It also includes pseudocode implementations for the most crucial portions of the system. Chapter 5 provides the overview and framework of our main experiment. Chapter 6 demonstrates both a proof-of-concept and a full run of our system and provides results as well as a statistical analysis. Chapter 7 provides concluding remarks.

1.3 System Overview

General Overview

Immediately when the program is run, the system first generates a 3D simulation world and a 2D map of that world from raw text files containing height information. Included as well is a text file containing a list of (year, water level) ordered pairs. Each of these ordered pairs contains a year in the interval of 9793 to 6343 BCE, and the corresponding water level of Lake Huron for that year (taken from [1]). The system automatically interpolates the water levels for years not explicitly contained in, or entered from, the 2007 Lewis data.

The system has two basic modes. The first is the standard simulation, and the second is hunting blind placement simulation. The standard simulation is activated by pressing the "Start/End Sim" button. When active, it runs through the period of 9793 to 6343 BCE. Environmental changes over time are visible in both the 3D simulation world and the 2D hash map thereof. At any time, the standard simulation may be paused, and a herd of caribou may be generated and given a start point and an end point. (The caribou are currently equipped with a version of the A* path planning algorithm, and will use this to find their way from start to end.)

The second mode, hunting blind placement simulation, runs from a user or programmer-defined starting year to a user or programmer defined ending year, and in this mode stops every X years (where X can be determined by the programmer or user). For each run, the system creates a herd of caribou which finds its way from the left edge of the current map to the right edge of the current map. Again, the caribou use the A* path-planning algorithm to find their way. Since the shore closest to the water tends to have the most vegetation, the system automatically chooses start and end points for the caribou that are close to water so that the caribou tend to travel the path with the most possible vegetation. After start and end points are chosen, the system uses a cultural algorithm to determine the best locations to place hunting blinds. Once the hunting blinds are placed, the caribou actually travel from the start to the end point using the A* algorithm. During the run, the hunting blinds are scored based on various factors, and these scores help determine the placement of the hunting blinds in the next generation.

Once the caribou have reached the ending point, the system takes a screenshot of the hashmap representation of the environment and saves it to a jpeg file. The process repeats until the last simulation year has ended. The resulting screenshots can be used as frames to make a video of the changes in caribou migration patterns and hunting blind placement for either the entire Alpena-Amberley land bridge over its entire existence, or a part of it within a certain portion of the time that it existed.

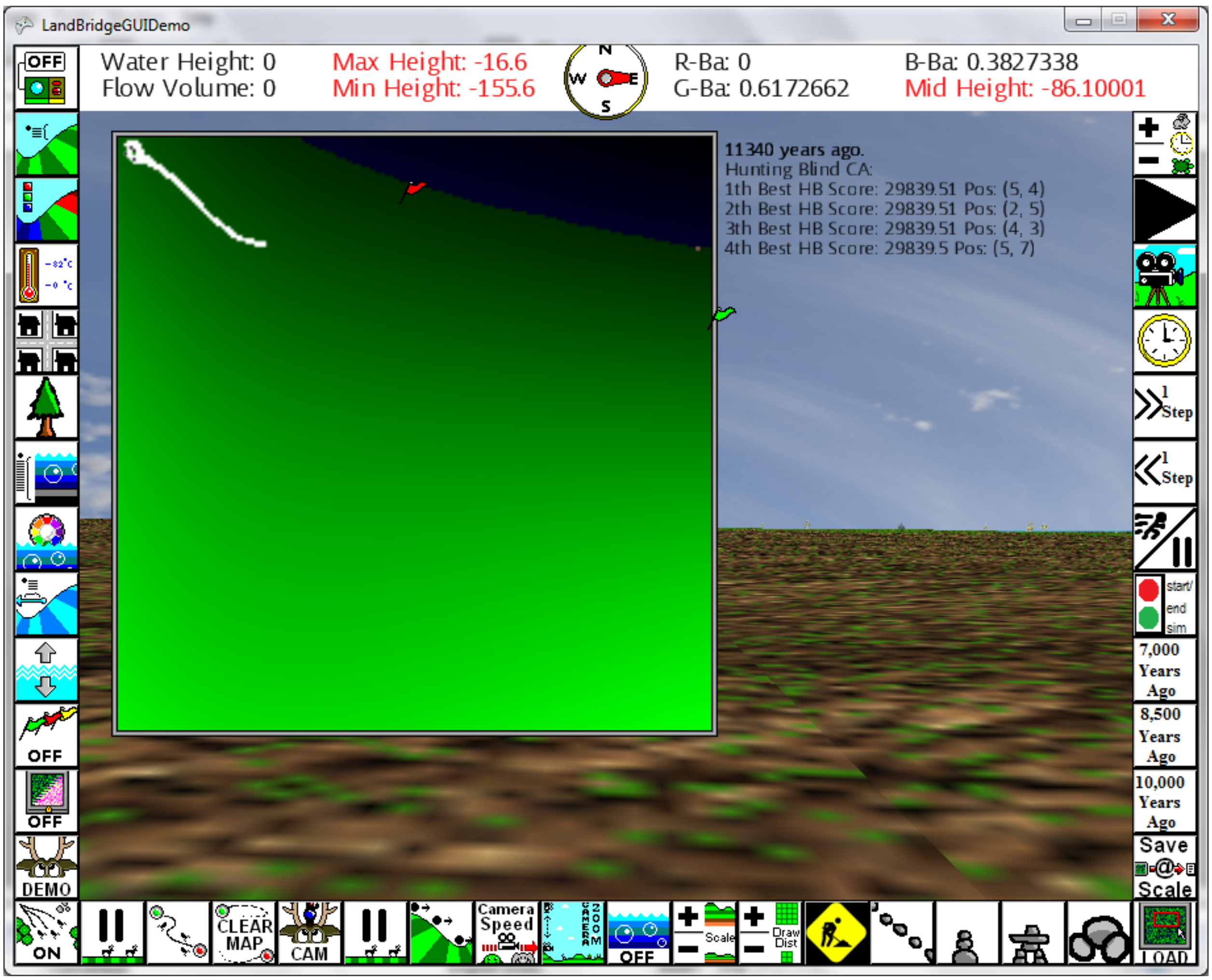


Figure 1: Screenshot of the Land Bridge System in Action

CHAPTER 2: LAND BRIDGE PREVIOUS WORK

2.1 Pre-2009 Work and O'Shea's 2009 Huron Expedition

The fact that the Alpena-Amberley ridge was at one point an isthmus connecting what is today northern Michigan and southern Ontario across Lake Huron is not itself new knowledge. Since at least the 1980's, the models of various respected geologists have shown it as an uninterrupted land corridor with two lakes on either side during part of the melt phase of the Laurentide ice sheet [2] [3]. However in 2009, under the leadership of Prof. John O'Shea, the University of Michigan launched an expedition to explore the bottom of Lake Huron using underwater autonomous vehicles (UAV's) launched from surface boats. During the expedition, the research team found what were later conjectured to be the remains of prehistoric hunting blinds and caribou drive lanes [4]. When the results were published, there was an immediate resurgence of interest within the research community concerning the land bridge, including the AI community who were interested in modeling the behavior of the caribou and hunters that roamed the land bridge in prehistoric times.

2.2 Kevin Vitale's Work

One of the first computer models of caribou behavior on the land bridge was implemented by Kevin Vitale in 2009, discussed in his paper "Learning Group Behavior in Games Using Cultural Algorithms and the Land Bridge Simulation Example" [5]. Vitale's program uses a cultural algorithm simply to teach caribou (represented as yellow triangles) how to successfully migrate as a herd across the land bridge.

Figure 2: Kevin Vitale's System

Vitale's cultural algorithm controls only the "wander behavior" of the caribou, that being defined as the deviation at any given time from the predetermined path from start point to end point. The wander behavior is determined by three values: The *wander target position*, the *wander circle radius*, and the *projection distance*. Vitale's pseudocode for his wander behavior mechanism is given below:

```
getSteering(&outputForce)
{
ΔX = current_X_Target * jitterValue
ΔZ = current_Z_Target * jitterValue
newWanderTarget = (ΔX, ΔZ)
newWanderTarget *= wanderRadius
newWanderTarget.X += wanderDistance
newWanderTarget.Z += wanderDistance
output.angle = SetOrientationTowardsTarget(newWanderTarget)
output.linearForce += wanderTarget * maxSpeed
}
```

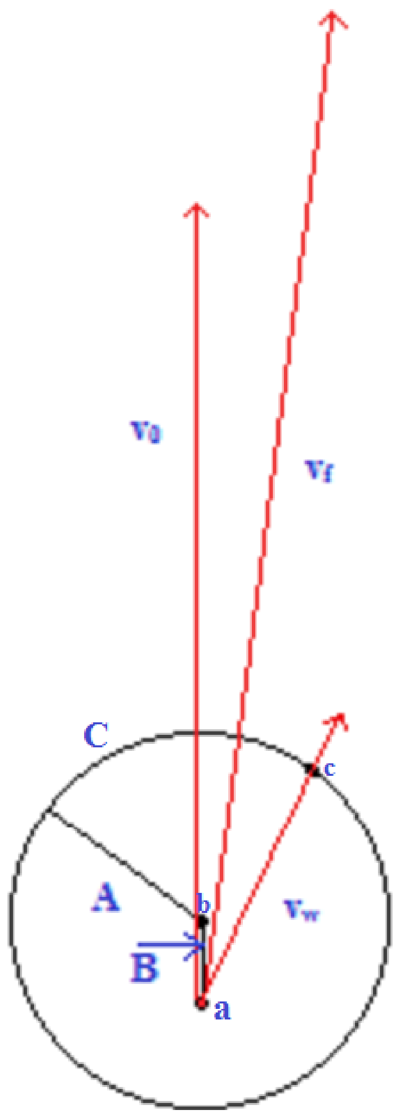


Figure 3: Schemata of Vitale's Wander Mechanism

The diagram above details how Vitale's wander behavior mechanism works. The point *c* is the wander target position, which is always located on the wander circle *C*, having radius *A. B* is the projection distance (the distance between the center of the circle and the caribou's current position, labeled *a* on the diagram). A fourth parameter, the jitter value, determines the change in the wander target position every time the *getSteering* function is fired. Of the four critical parameters, only the initial wander target position and the jitter value are determined by the cultural algorithm. The latter two, the wander circle radius and projection distance are simply hard coded into the program.

Not surprisingly, Vitale ultimately found that parameter values producing caribou that wandered very little were the most successful in getting the largest number of caribou safely across the land bridge.

2.3 "Serious Game Modeling Of Caribou Behavior Across Lake Huron Using Cultural Algorithms And Influence Maps"

The next major computer program for modeling caribou behavior on the Alpena-Amberley land bridge was written by James Fogarty and detailed in his 2011 paper "Serious Game Modeling Of Caribou Behavior Across Lake Huron Using Cultural Algorithms And Influence Maps" [6].

Influence Map

As a resource for its cultural algorithm, Fogarty's program uses an influence map in which each square is given a score resulting from the availability of food therein, the dangers previous generations have encountered therein, and the difficulty of the square's terrain (peaks and valleys are considered "difficult terrain", as opposed to level ground which is considered "easy terrain".

A* Algorithm

In Fogarty's program, the land bridge map is divided up into square cells. The program uses the A* algorithm to create a path from a given start square to a given finish square.

The A* algorithm itself is Dijkstra's algorithm with a heuristic (provided by the algorithm user) included. Dijkstra's algorithm is a search algorithm for graphs that finds the shortest path through a given graph from a given initial vertex to a given terminal vertex. It, and A* which derives from it, *always* find the shortest path, provided at least one valid path exists [7]. "Shortest" in this context means not merely the path containing the least number of vertices, but the path containing the smallest sum of *vertex weights*.

Fogarty's influence map provides a weight for each of his map squares (which can be thought of as graph vertices), and is calculated from the factors described in the *Influence Map* subsection. His A* algorithm generates a shortest path across the land bridge based on these weights.

When Fogarty's A* algorithm actually generates the path, it shows up as a series of blue diamonds on the program's display, as shown in the following figure:

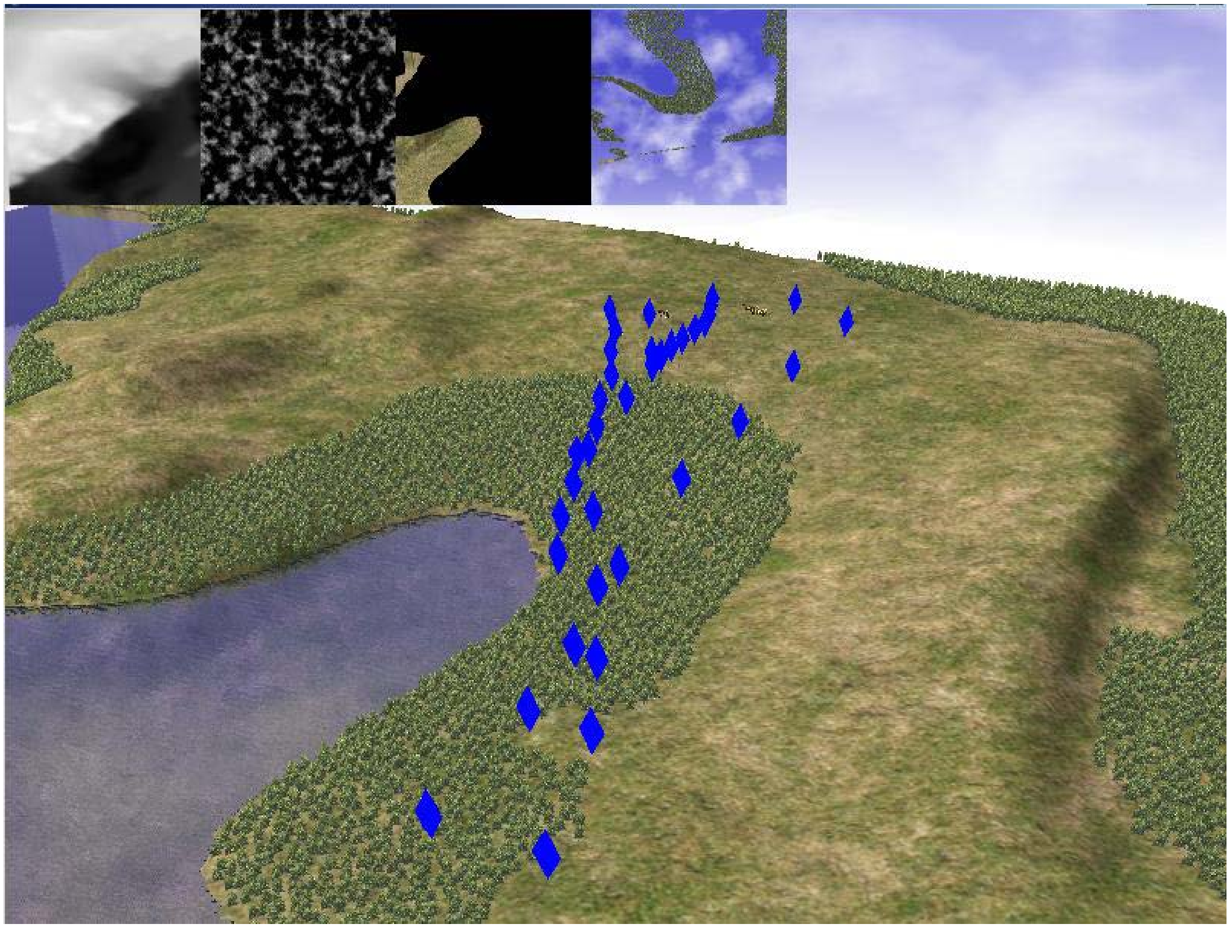


Figure 4: Fogarty's System

2.4 "Path Planning in Reality Games Using Cultural Algorithm: The Land Bridge Example"

Jin Jin, in his work entitled "Path Planning in Reality Games Using Cultural Algorithm: The Land Bridge Example", provided another variant of the A* algorithm for calculating caribou paths similar to Fogarty's. Jin's A* variant returns the least-total-value path from a start vertex to a terminal vertex. It uses terrain difficulty value, food value, and distance value as the terms which determine the raw value of an individual square [8]. The total value is determined by these three terms multiplied by a terrain weight, a food weight, and a distance weight, respectively. These weights can be either baked into the program (as they were in the June

2012 version of Jin's program within Palazzolo's framework), or they can be determined using a cultural algorithm.

Geometry Value

In Jin's A* variant, the geometry value of a given vertex is determined by the terrain that the vertex is located on, whether it be rocks, grass, sand, water, or another terrain type. "Easier" terrains have lower geometry values than terrain types deemed "harder". (Note that the mid-June 2012 version of Jin's program effectively contained only two terrain types: water and non-water. Water squares were given a geometry value of 255, whereas non-water squares were given a geometry value of 0.

Distance Value

In Jin's A* variant, the "distance value" of a given square is the Euclidean distance from that square to the terminal square. The higher this Euclidean distance, the higher the distance value of the square.

Food Value

In Jin's A* variant, "food value" in a given square is the same as the vegetation value in that square, which is ultimately determined by Palazzolo's program which provides the framework for both Jin's program and ours. Generally speaking, Palazzolo's program assigns higher vegetation values to squares which are closer to water, and lower vegetation values to squares which are further inland. Unlike the previous two terms, the higher the food value, the more desirable the square.

Total Value of a Square

In Jin's A* variant, as it was in the June 2012 version of Palazzolo's program, the total value of a square is given by the following equation:

$$V_s = W_g \cdot g_s + W_d \cdot d_s - W_f \cdot f_s$$

Equation 1: Total Value of a Square in Jin's A* Variant

In this equation, V_s is the overall value of the square, W_g, W_d, and W_f are the geometry, distance, and food weighs, respectively, and g_s, d_s, and f_s are the square's geometry, distance, and food values, respectively.

Finding the Minimal Value-Sum Path

Again, Jin's program finds the path from a given start point to a given end point which has the minimal combined value of all squares within that path. In other words, it finds the path P out of all possible paths which minimizes the following function.

$$T(P) = \sum_{\forall\, s \in P} (f_s - d_s - g_s)$$

Equation 2: Total Value of a Path P in Jin's A* Variant

Learning Curve Diagram

Below is the learning curve diagram for a sample run of Jin's program using his cultural algorithm.

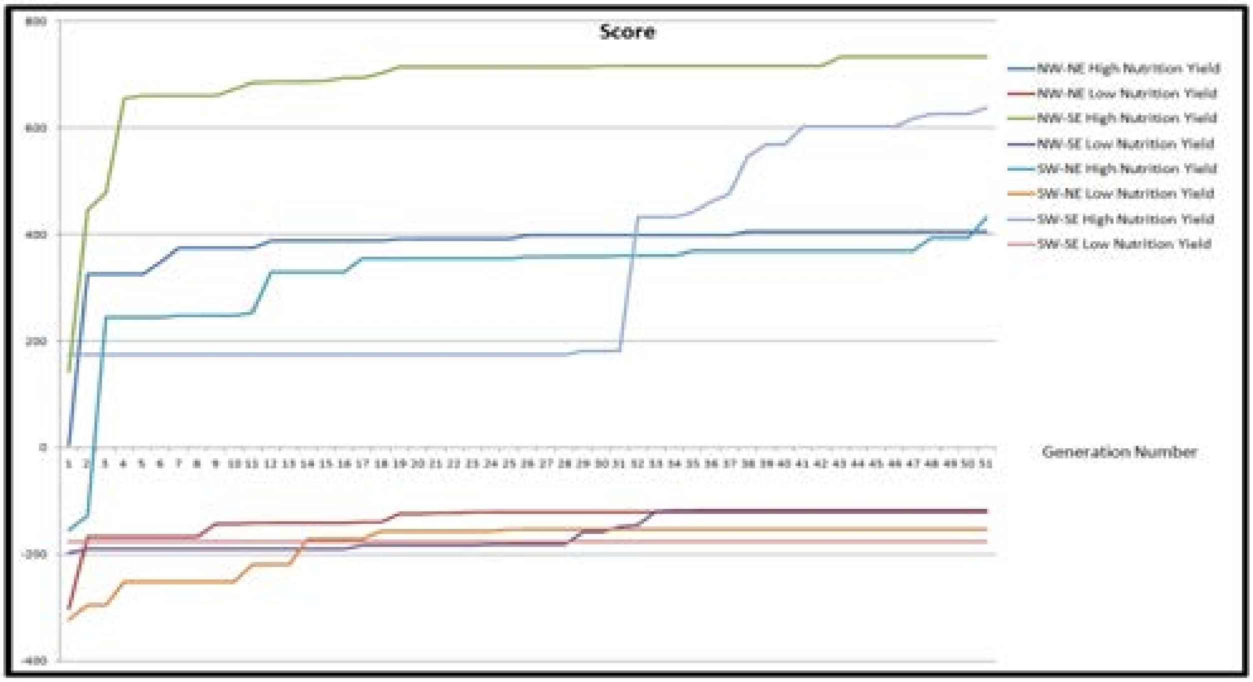


Figure 5: Jin's CA Learning Curve (Total Score vs. Generation) [8]

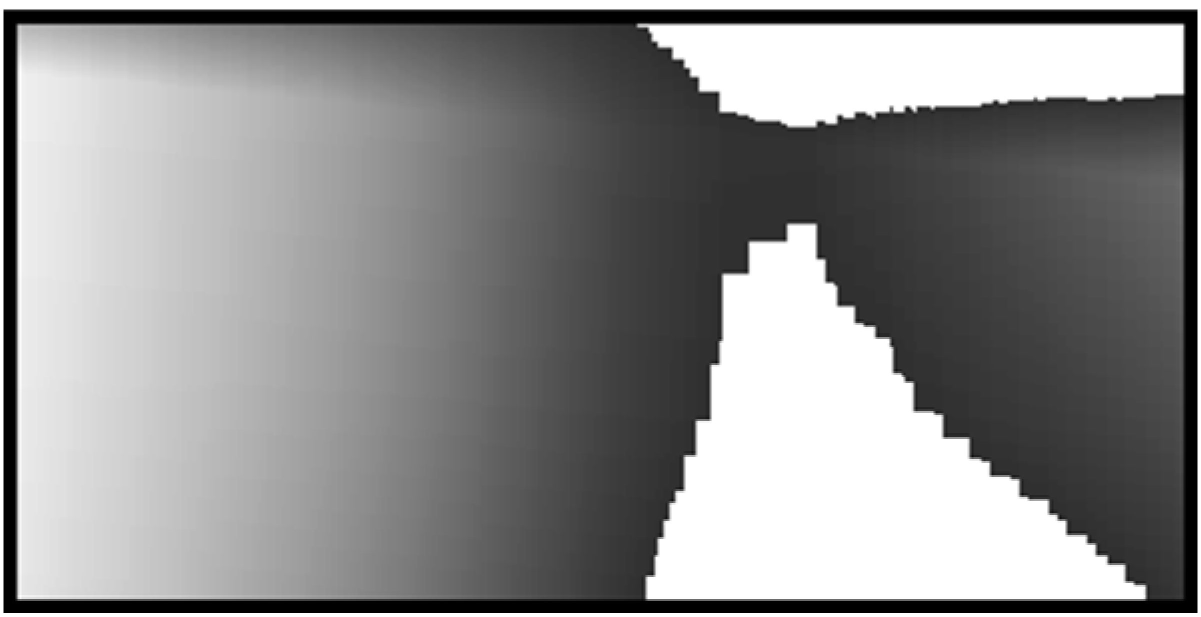


Figure 6: The Terrain Upon Which Jin Performed His Experiment [8]

Unfortunately, most of Jin's work did not make it into Palazzolo's program by the time the author of this piece commenced the main portion of the coding work required here. Therefore our work uses the A* algorithm included in Palazzolo's program as of June 2012 to calculate caribou paths. However, Jin's work is still important to mention here because it uses such a similar approach to the one used in this work.

2.5 "Cultural Algorithms in Dynamic Environments"

Introduction and System Overview

In 2000, Saleh Saleem devised a semi-dynamic CA system somewhat similar to the one we are about to introduce. Saleem's system consists of an environment containing a number of cones. The objective for his CA is to find the x and y coordinates containing the top of the tallest cone in an environment containing multiple cones. However, Saleem's system gives his CA only a certain number of generations to reach the goal before the environment changes, i.e., the cones are moved to new locations.

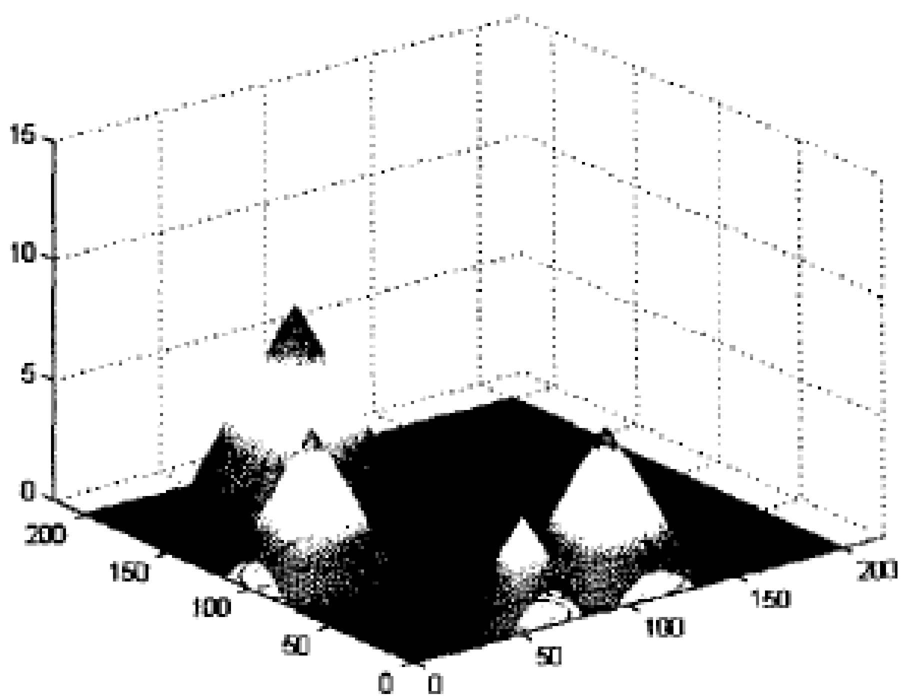


Figure 7: Snapshot of Saleem's Experimental Environment [9]

Experimental Setup

In Saleem's main experiments, there are two cones, the first with a fixed height of 9.5 and a fixed slope of 7.5, the second whose slope may vary between 2 and 7 and whose height may vary between 3 and 6. The x and y coordinates of the centers of both cones may vary randomly within the interval [-1, 1], although the first cone always starts out with its center at (x, y) = (0.15, 0.20) before the first instance of environmental change.

Experimental Results

Saleem performed various experiments with his system, the most interesting of which involved the comparison of his CA against a self-adaptive, population-only, non-cultural evolutionary program (EP) in a situation where the environment changed every five generations. Saleem provides the results of this experiment in the following graph of shift magnitude vs. best value. Note that the absolute best value is 9.5, recalling that the first cone's height is fixed at 9.5 and the second cone's height must be between 3 and 6. Saleem demonstrates that his algorithm which contains a cultural component performs much better at the task of finding the top of the tallest cone than the competing algorithm which has no cultural component.

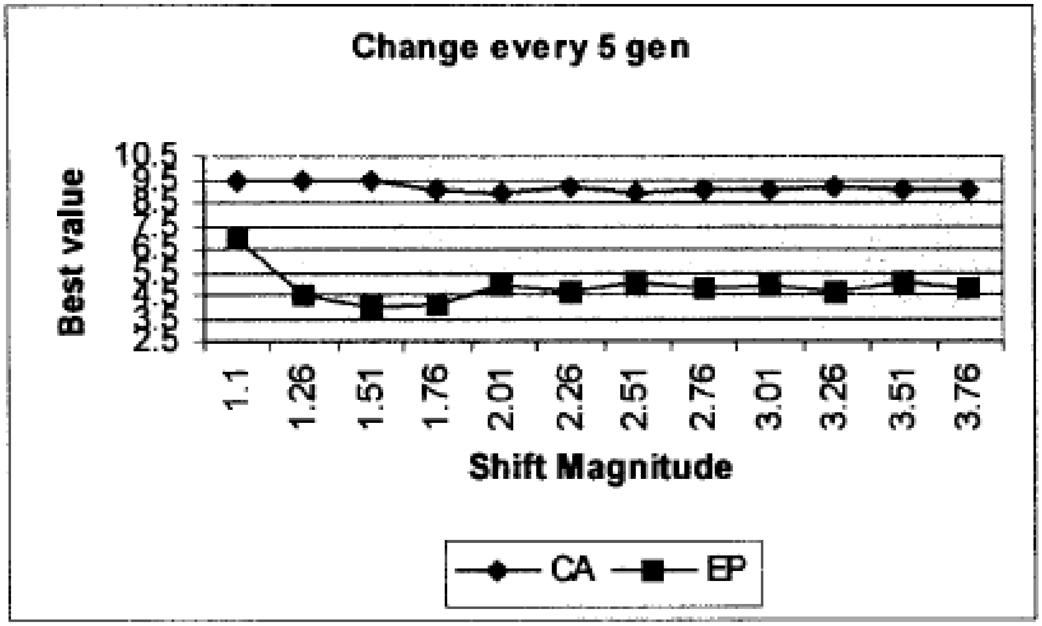


Figure 8: Avg Best Value with Environmental Change Every 5 Gens (CA vs. EP) [9]

Conclusion

Saleem's system is certainly valuable prior work on testing how cultural algorithms deal with a changing environment. Still, we would contend that Saleem's system, while valuable as pioneering work, is only partially dynamic and not fully dynamic. This is because in his system, the environment does not change every single generation. Rather, his CA is given a certain number of generations in which the environment remains static (5 generations was the smallest number he used in any of his experiments) in which the CA attempts to reach peak fitness. It is only after those generations have concluded that the environment changes [9]. In our system, as is the case in the natural world, the environment is *never* static. Our CA is forced to truly learn "on-the-fly", in other words to continually adapt to an environment that never ceases to change, not even for a single generation.

CHAPTER 3: OBJECTIVES AND HISTORICAL OVERVIEW

We now provide the overall objectives, motivations, and background for our own work. Although our system may be used as a tool to simulate any natural environment that changes over time, the immediate motivation for its creation is the study of the Alpena-Amberley Land Bridge. Although simulation tools already exist to simulate parts of the Alpena-Amberley Land Bridge during a specific instant in time, no tool existed before now that could simulate the entire Land Bridge over the entire time period that it existed. To accomplish this, it was necessary to draw upon the body of Paleolithic-Era geological research concerning the Great Lakes system in general and Lake Huron in particular.

3.1 Choke Points

First of all, the Alpena-Amberley Ridge has two low points that serve as “choke points”. When the lower of these, shown on the map as “Choke Pt 1”, becomes covered with water, the Alpena-Amberley ridge is not a Land Bridge, but at best two peninsulas with a strait between them. (Choke Point 1 is about 57.5m below today’s Lake Huron level, or about 118.5m above today’s sea level.) When the other choke point, shown on the map as “Choke Pt 2”, becomes covered with water, the Alpena-Amberley ridge is at best two peninsulas with one island in the middle of them, the island separated from the peninsulas by two straits. (Choke Point 2 is about 52.5m below current Lake Huron level, or about 123.5m above current sea level.)

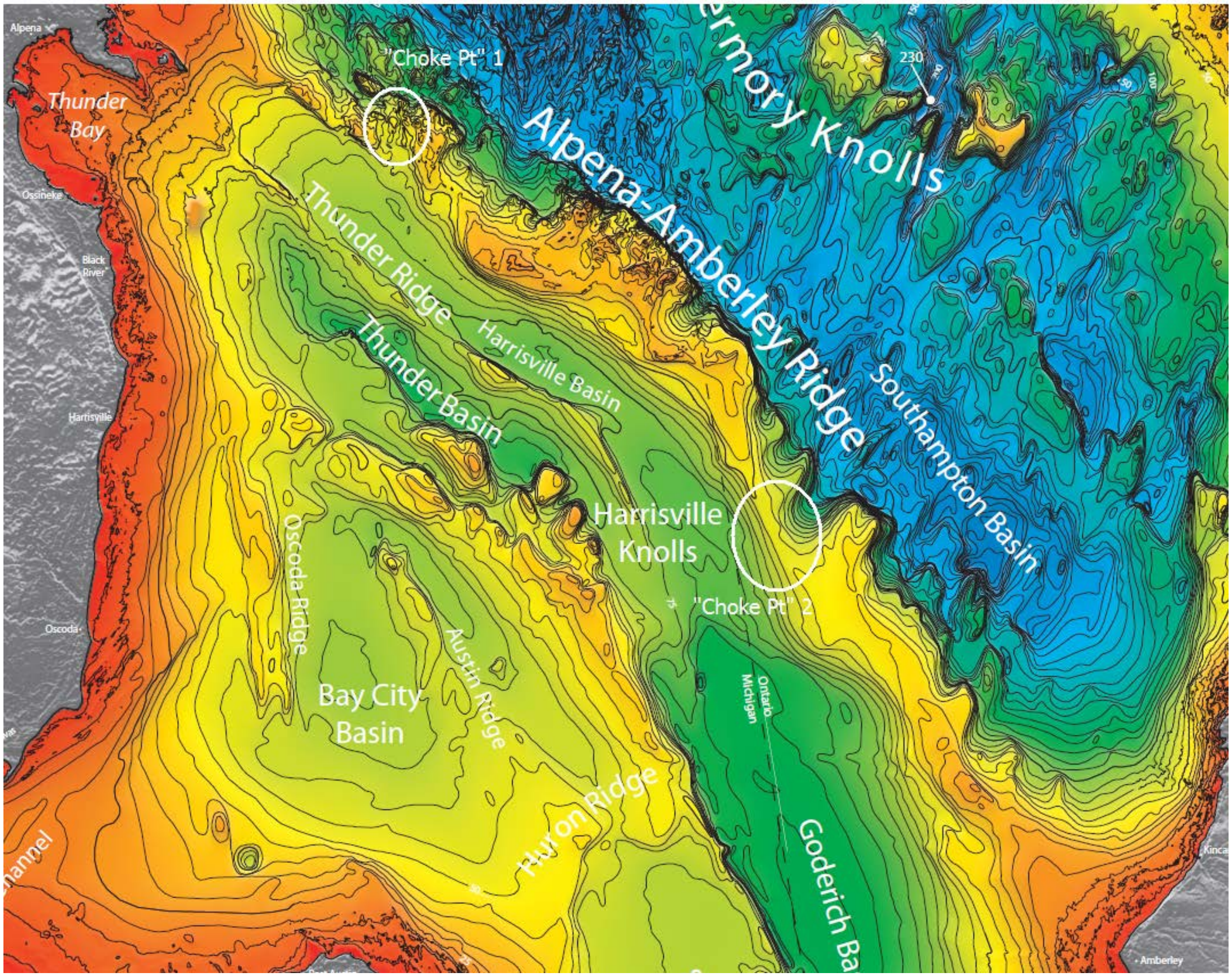


Figure 9: NOAA Bathymetry Map of the Great Lakes Basin [10]

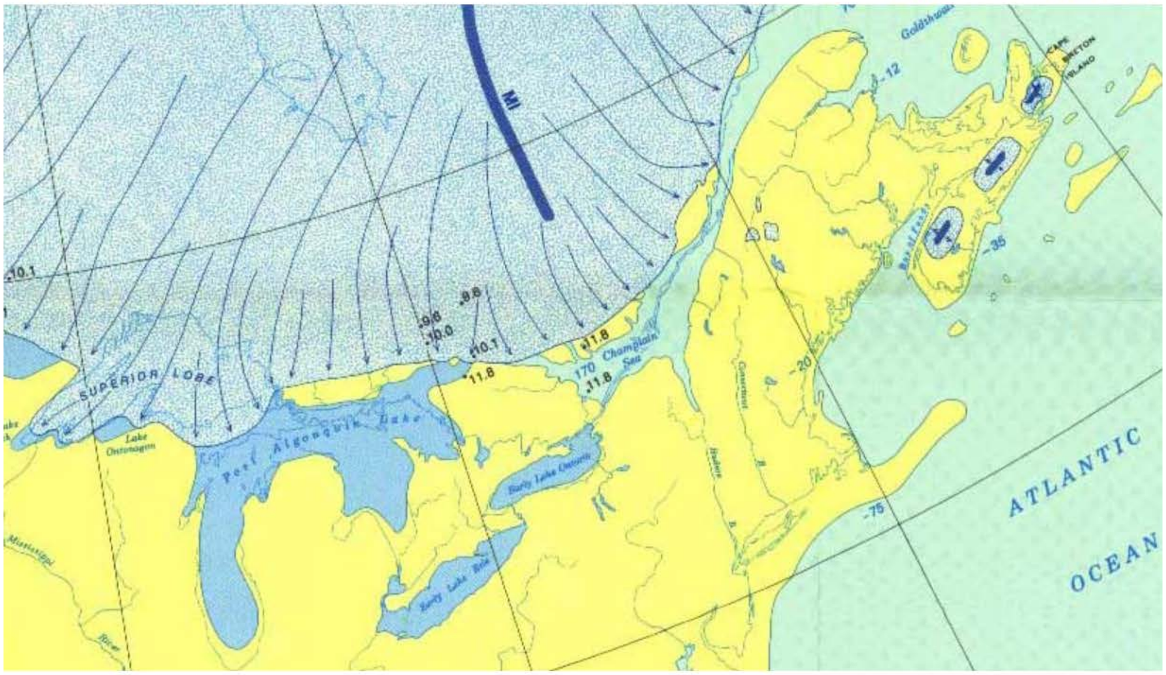
3.2 Dyke Model

According to [2], at some point (which [2] conjectures was shortly after 11,000 years ago), a glacial "plug" separating the Great Lakes from the North Bay Outlet leading to the Atlantic Ocean melted. As most of the water in the Great Lakes was at a significantly higher elevation than the North Bay Outlet, most of it flowed out through the Outlet into the ocean when the glacial "plug" blocking it melted.

From [1], we know that [2]'s estimate for the time of the melting of the "plug" is probably too late. However, [2]'s diagrams are still the most useful out of any in the literature for understanding the *sequence of events* that led to the Alpena-Amberley Land Bridge's formation.

We can see very clearly from Dyke that during Early Stanley, the water was at its lowest point, the land bridge was at its greatest extent, and thus provided the best opportunity for caribou to use the land bridge as a crossing point, and therefore for hunters to hunt on the land bridge. Eventually, however, the point was reached where the elevation of the North Bay Outlet was higher than the lake level of the Great Lakes System. This point was reached because of the rise of the North Bay Outlet's elevation due to undergoing a reversal of elevation depression from the weight of the prior glacial ice. At that point, water began flowing back into the Great Lakes System through the North Bay Outlet. The water continued its remorseless rise until the land bridge was no longer crossable, and eventually submerged entirely [2]. Eventually, it the water level tapered off at about 176m above sea level, which is where it is today.

11,000 Years Ago **(Dyke & Prest, 1987)**



10,000 Years Ago **(Dyke & Prest, 1987)**

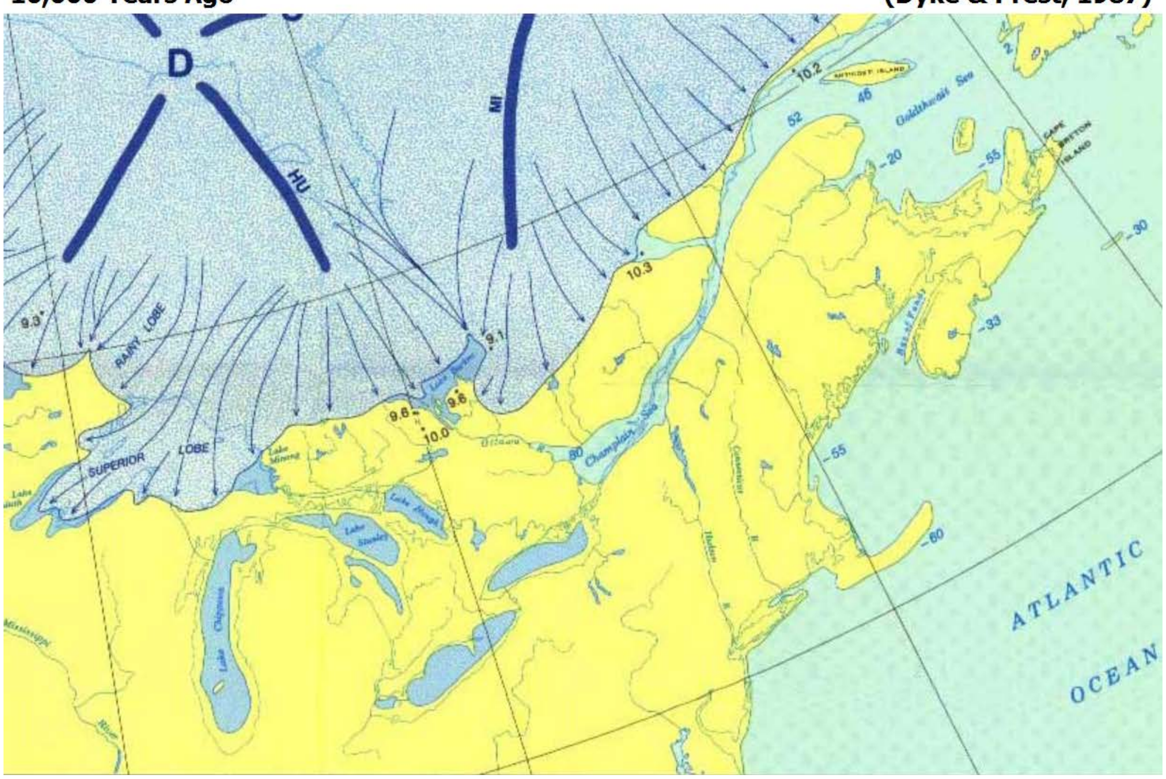


Figure 10: Prehistoric Michigan and the Surrounding Area [2]

3.3 Lewis Lake Level Reconstruction

For our experiments, we use the Lewis reconstruction of prehistoric lake levels [1]. It should be noted that according to Lewis, there may have been three brief "reversals", one at ~8,593 BCE, one at ~7,783 BCE, and the last at ~6,997 BCE, during which the Land Bridge was temporarily submerged, only to re-emerge shortly thereafter [11]. We include these reversals in our model. Nevertheless, we can infer from Lewis's results that in about 6343 BCE, the two land bridge "choke points" mentioned before became submerged, never to be above lake level again. Although certain portions of the Alpena-Amberley ridge did remain above water longer, it was never to be a full land bridge again after about 6343 BCE [1].

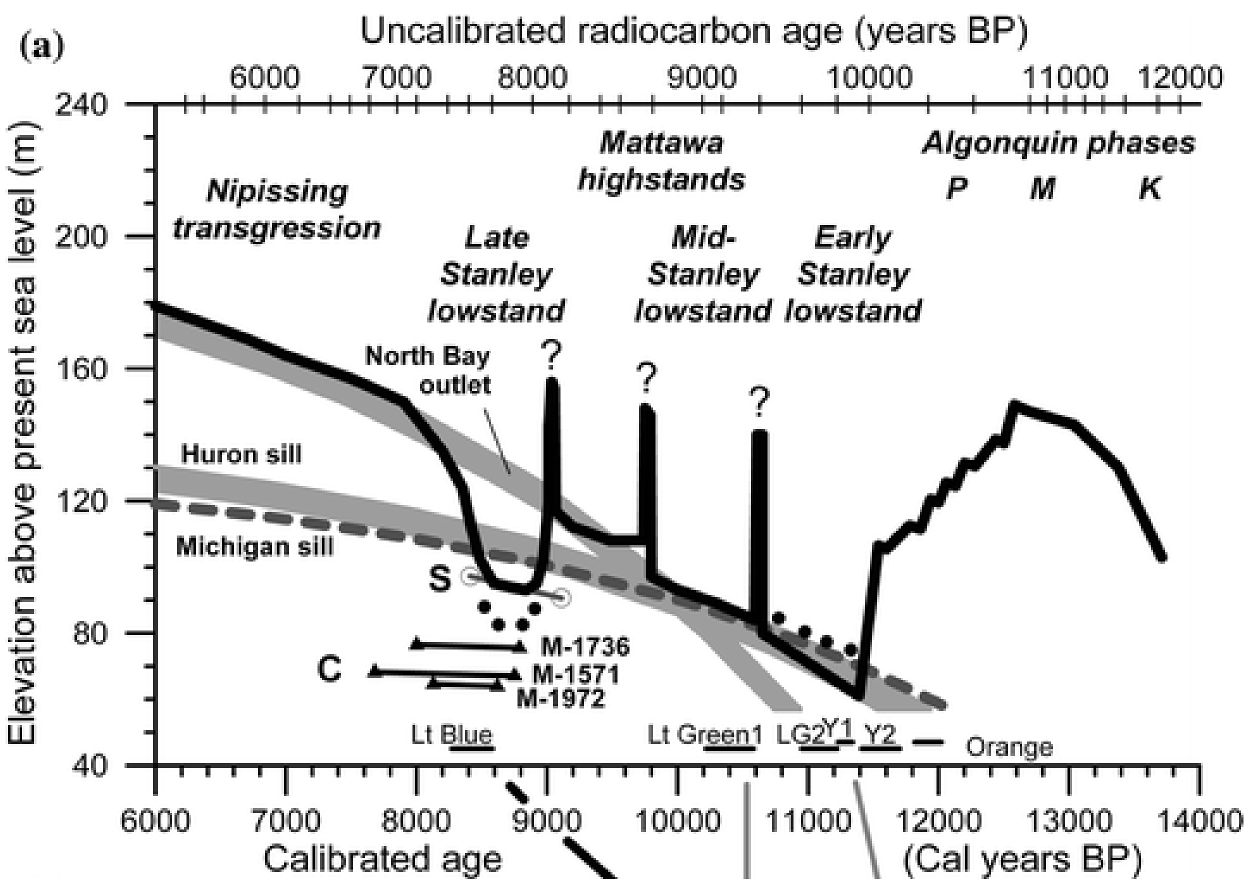


Figure 11: Prehistoric Great Lakes Water Levels According to C.F.M. Lewis [1]

3.1 Hunting Blinds

Hunting blinds of the type found on the Alpena-Amberley ridge are semi-permanent or permanent structures made of several large stones that form a rough circle enclosing a particular space. Their most obvious purpose was to keep the animals from seeing the hunters, so that the animals would not know to avoid the hunters and thus would wander into spear or atlatl range where they could be killed. However, it is curious that the hunters chose stone rather than wood or large mounds of dirt to build these blinds. The latter materials are not only also opaque and thus keep the animals from prematurely seeing the hunters just as well as the stones, they are lighter and thus would seem easier to use in building a hunting blind. It is quite possible that some of the hunters *did* use these materials on some occasions, but only the stone blinds have survived millennia of being underwater. Still, it is undeniable that *some* hunters chose to use large stones in lieu of lighter materials. A probable reason is that, although a temporary blind built of wood or dirt could last for a short hunt, but would be washed away should a flash flood hit the land bridge. Today, flash floods are a relatively common occurrence in certain parts of Michigan which are adjacent to the various large lakes. They occur when the wind blows the lake water so hard that a column of water actually leaves the lake and washes over the plain. In addition to this, meltwater pulses at various stages of the collapse of the ice sheet may have been yet another source of flash floods back in the prehistoric era. Since the Alpena-Amberley land bridge has never been very high above lake level, even when lake level was at its lowest, there is no doubt that the land bridge experienced many flash floods which would destroy the temporary blinds made of dirt or wood, but would leave the permanent ones. Permanent stone hunting blinds would thus seem to be reserved for the most important and strategic locations, places where the hunters thought that the caribou *always* passed by, at least at certain times of the year.

CHAPTER 4: CULTURAL ALGORITHMS AND SYSTEM DESIGN

It is very rare to find anything manmade as old as the prehistoric hunting blind and caribou drive lane remnants under Lake Huron that Dr. O'Shea's team found back in 2009. The research community is lucky to have found these few scattered remnants, yet they are not numerous enough for us to directly create a straightforward model of prehistoric hunting blind placement from their locations alone. We must thus do the next best thing, which is to use machine intelligence to simulate the prehistoric hunters' human intelligence and thus their ability to decide where and when to place the blinds.

Given that humans are tribal creatures with the ability not just to individually acquire but to *share* knowledge among groups, this situation calls for a technique that reflects not just an individual but a *tribal* ability to accumulate knowledge and store it for use in future situations. In the 1970s, such a technique was developed by Dr. Robert Reynolds called *cultural algorithms.* In creating cultural algorithms, Dr. Reynolds drew an analogy between group learning, the process of Darwinian natural selection in biology, and the tendency of group knowledge acquired in the past to influence current decisions by individual members of groups. [12]

4.1 Structure and General Algorithm

Cultural algorithms contain a *population space* which is influenced by a *belief space*. Population space is defined as a set of solutions to the problem which have the ability to evolve from generation to generation. The belief space can be defined as the collected set of experiential knowledge, which has the ability to be influenced by individuals within the population space according to their varying degrees of success, and which has the ability to influence subsequent generations of individuals within the population space.

The following is a general statement of a generic cultural algorithm:

1. The population space and belief space are initialized.

2. Population members are evaluated through a fitness function, and the population is ranked.

3a. The population members ranked highest are allowed to influence the belief space.

3b. In some cultural algorithms, the population members ranked *lowest* are also allowed to influence the belief space by providing various forms of *negative* information to it about their solutions.

4. The best solutions are allowed to reproduce and create children. Operators are applied to at least some of the children which make them into mutated variants of their parents.

5. The belief space influences the children's genomes and/or their behavior in the problem space.

6. Steps 2 through 5 are repeated until a stop condition is reached.

A visualization of this process can be found in the following diagram:

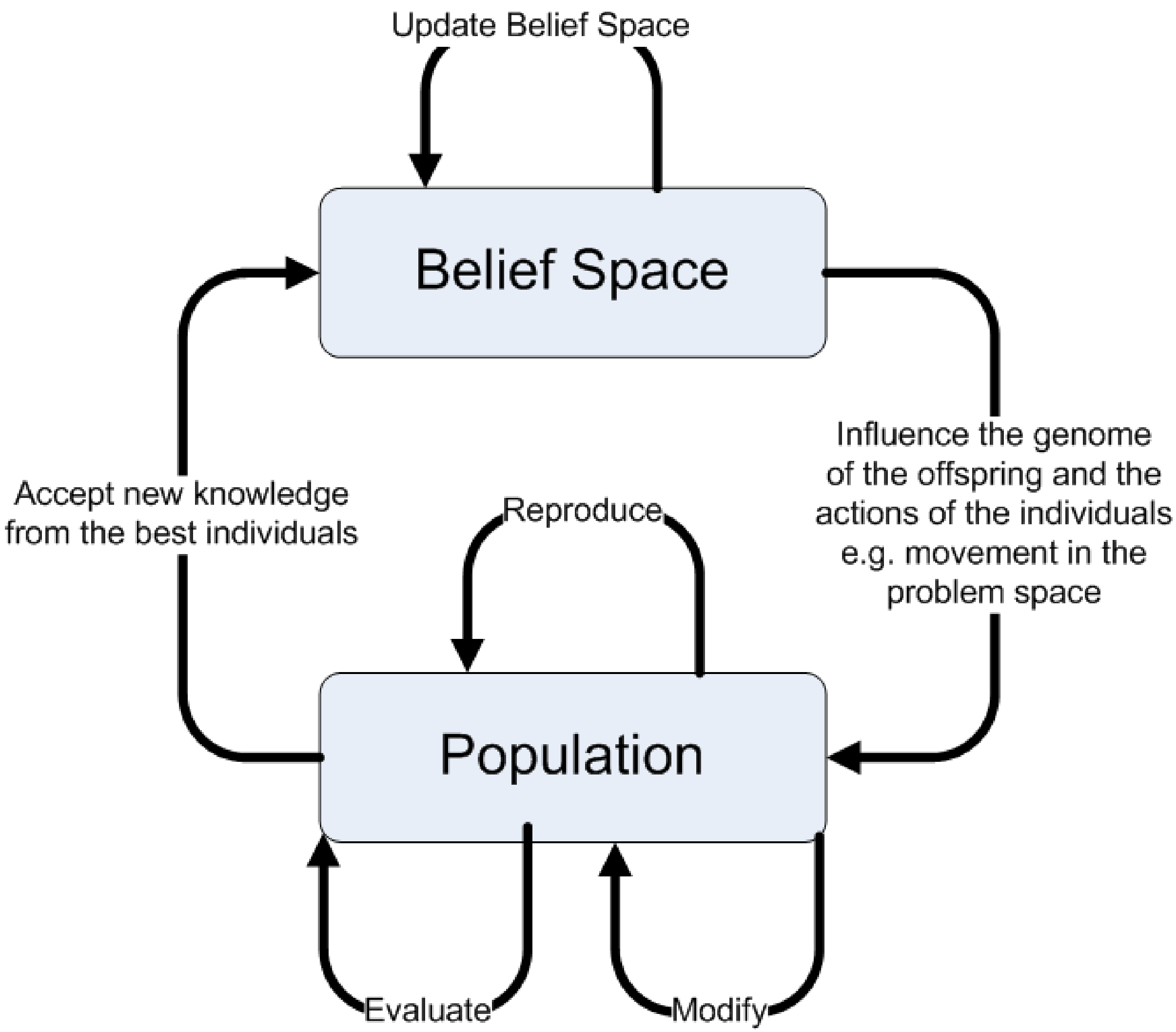


Figure 12: Schemata of Cultural Algorithms [8]

4.2 Belief Space Knowledge Types

Generally, researchers who use cultural algorithms divide knowledge into five different types, usually described as follows: [13].

Normative Knowledge

Normative knowledge is a set of variable ranges that either are initially expected to produce good fitness values for experimental agents or are known to have produced good scores in the past [14].

Domain Knowledge

Domain knowledge concerns the overall shape of the search space itself [14]. The purpose of domain knowledge is to deduce the shape of the search space. Once the shape is known, it often becomes much easier to find optimal values. This knowledge source is crucial when the task of the CA is to find extrema within a solid or hypersolid.

Topographical Knowledge

According to [14], topographical knowledge was first devised as a knowledge source in [15]. Topographical knowledge is knowledge concerning the regional features of the search space itself. It was used in [15] in order to exclude different parts of the search space as either totally infeasible or only partially feasible. Most subsequent uses of topographical knowledge take a similar approach, as being able to ignore whole ranges of infeasible solutions both reduces the opportunity for error and cuts down on search time.

Historical (Temporal) Knowledge

Historical knowledge (also called temporal knowledge) concerns important events that happened during the search and the general state of the search space at a specific point in time. It can contain a record of good (and bad) solutions that happened in the past so that future agents can go toward (or avoid) those solutions.

Situational Knowledge

Situational knowledge concerns positive and negative exemplars. Solutions that score high are considered positive exemplars, and cultural algorithms can take this into account and look for similar solutions that might be even better. In contrast, solutions that score low are considered negative exemplars, and cultural algorithms can take that into account and steer clear of similar solutions, so as to avoid wasting time with them.

In our problem, spatial and temporal knowledge are valued above other forms of knowledge. This is because our research problem format, unlike those of previous land bridge computer models, contains the additional difficulty that the environment *changes* with every year as the water level either rises or falls. Furthermore, the environmental change is nowhere near constant from year to year. In some years the change is very little, in others it is significantly more, in still others it is completely drastic (such as the entire land bridge being flooded or almost flooded). What is more, the purpose of our algorithm is not necessarily to find *optimal* solutions per se, but rather to simulate the actual human intelligence of the prehistoric hunters, given human imperfection plus all the difficulties that they would have encountered in planning each year considering the unpredictably changing environment. A system constructed properly in this way will enable archaeologists on Great Lakes expeditions to successfully use it to discover the *actual* locations of hunting blinds and other artifacts, as the range of locations that the system provides will reflect the choices of the actual prehistoric hunters.

In order to achieve this, there are several crucial assumptions that we must incorporate into our system:

The first is that the prehistoric stone circles found under Lake Huron by O'Shea's team are indeed hunting blinds and had a similar role in the hunt to those used by modern-day hunter-gatherers in their hunts today. In other words, we assume that hunters used them as structures to hide behind so that the animals would not see them, and would thus not deviate from their route bur rather get close enough for the hunters to make a successful kill. Although it is nearly impossible to say for certain what *any* prehistoric structure truly was, we feel that this is a fair assumption because the structures look so similar to hunting blinds used by modern-day hunter-gatherers, and it is difficult to conceive what else the structures might have been.

The second assumption is that not only were the hunters (like all humans) unable to see into the future, they didn't even have complete knowledge of present conditions everywhere on the land bridge. This was especially true towards the beginning, when the hunters first arrived upon the land bridge. At best, they knew how well current hunts had fared, they could speculate about what might have happened had they chosen neighboring areas to place their hunting blinds instead, and they could accumulate and remember this knowledge for future hunts.

4.3 High-Level Design

Overall Simulation

For simplicity's sake, we divide the 3,450 year time period in question up into one-year long "runs", and we will assume that the caribou migrate across the land bridge once per year. We will assume also that the hunting blinds are set up once per year, in anticipation of the caribou migration, and that they are not moved during the year once they have been set up. As described in previous sections, our system has the ability to update the terrain based on environmental change over time. To simulate the fact that the hunters cannot exactly predict how the environment is going to change, for each year/run we will *first* have the hunters choose locations for their hunting blinds at the beginning of the year (using certain weights provided by GA chromosomes, to be discussed later). *Then* we will use our system to update the terrain for that year, and *then* the caribou will migrate through. Each hunting blind will then be scored by an objective function containing several factors which together are a fair determinant of hunt success or lack thereof. Each hunting blind will then update the belief space with the values of these factors for its specific location (and, reflecting the assumption of local knowledge but non-omniscience discussed in the last subsection, the belief space will be updated with the values of these factors for each square within a Moore radius of the hunting blind as well). After this, the chromosomes for each hunting blind will undergo various mutation operations which will produce a new batch of chromosomes (this procedure biased, of course, towards the chromosomes that produced the best blinds in the generation that just finished). Each chromosome will encode a "weight" for each important factor in hunting blind placement. The value of the weight for a certain factor determines how important the hunters controlling that blind consider that factor compared to other factors. Actual blind placement is determined by looking at each square in the belief space and choosing the square with the "best" values for each factor based on the weights encoded in the hunting blind's chromosome. Once the blinds are placed, yet another year/run will begin (as described before), and this process will continue until a certain terminal condition is reached. As discussed in a previous chapter, 6343 BCE is when the land bridge becomes two peninsulas separated by a strait due to rising water levels and never reconnects again. This seems the most logical choice for our terminal condition (although the system's design of course allows the user to manually choose an alternate end year if desired).

Evaluating Hunting Blind Success and Failure

At this point, we need to consider precisely what factors make a certain hunting blind successful or unsuccessful. One obvious factor is how close the hunting blind is to the path of the prey. Obviously a hunter throwing his spear or atlatl at prey that is close by is much more apt to hit the prey than if he were throwing it from farther away.

Another factor is the difference between the altitude of the blind and the altitude of the caribou. If the hunting blind is higher (vertically) than the caribou, a hunter throwing at the caribou from that blind has the advantage that gravity is working for him. In other words, a projectile thrown from that spot at the caribou below will travel faster (due to gravity) than an equivalent projectile thrown at another caribou at the same vertical level as the blind. Thus in the first case, the projectile will travel faster increasing not only the likelihood of actually hitting a moving caribou, but also the damage done to the caribou, and thus the likelihood that the caribou will be hit but still get away is decreased. Similarly, a hunter trying to hit a caribou *above* him has gravity working *against* him. Projectiles that *he* throws are more likely to miss, and what hits he makes are more likely *not* to result in killing or at least halting the caribou.

A third factor is closeness to the nearest hunting blind. This is because hunting parties crowded too close together tend to interfere (unwittingly) with each other's success in the hunt. For example, when one hunting party makes a kill and goes to collect their kill, the herd usually reacts either by trying to move away, or sometimes even begins to panic. If the herd does panic, it is okay for the first hunting party because they have already made their kill, and all that is left is to haul it in and dress it, and then carry it back home. Even one kill was considered a very successful hunt. Caribou are quite large; a good-sized buck is apt to weigh over 300 lbs.

Also, prehistoric humans used *all* the parts of their kills without waste, including the eyes and other organs, so just one kill of a decent-sized buck provided fur for clothing, bones for tools, and could feed a small band for at least a week. However, the second hunting band, which has yet to make a kill, now must deal with a panicked herd or at least a herd which is wary of the spot where their fellow caribou died. This makes it much less likely that the second hunting band will have a successful hunt. Of course on other days, the second group will make a kill first and it will be the first group which will be in the disadvantaged position. Overall, both groups will have less kills on average per year than if they had been spaced farther apart. Even worse, overcrowded hunting parties have an increased risk of accidentally killing each other while trying to kill caribou or other game. Obviously the farther parties are spaced apart, the less likelihood there is of a tragic accident.

Closeness to water is a factor which, although it does not *directly* impact the quality of the hunting blind location, would still impact the hunters' decision where to place the blind. Locations that are close to water are more likely to be flooded during the course of the year, and an underwater hunting blind is of course useless.

Caribou and Hunting Blind Models and Their Behavior

Conveniently, Palazzolo has already created caribou and hunting blind models for his 3D Land Bridge platform which we are using, and Jin has provided the caribou with a variant of the A* algorithm which plots a path from start to end point while trying to maximize the amount of vegetation eaten. (Palazzolo has also provided an algorithm that creates vegetation for the map). Currently, the caribou do not automatically migrate across the environment. The user must manually choose a start and end point for them, i.e., there is currently no algorithm which automatically chooses a starting and ending point for a migration path across the map from edge to edge. This fact can be worked around, however by noting that the most vegetation-rich areas are always adjacent to water, and that the caribou always tend to seek out the most vegetation-rich areas while trying to minimize distance traveled. Thus if for each run/year we choose points adjacent to water on either edge of the map for our caribou start and end points (a process which can be easily automated), Jin's A* algorithm variant will, for each of our runs/years, generate the caribou migration path across the map that has the best balance between food availability and shortness of path length.

Final Products

At this time, we need to consider what information ought the system to *collect* from each run/year of the simulation, and how should it *present* that information to interested scientists when the simulation is finished. Recalling our main motivation, we want to analyze *how* the land bridge environment, including caribou migration patterns and human hunter patterns, changed over time, and we want to show archaeologists *where* they ought to look for new hunting blinds (as well as other significant paleoindian artifacts that they may also be able to find among them).

For the first objective, the best course of action seems to be to produce a video, about 5 minutes in length, displaying the changes in environment, caribou migration patterns, and hunting blind placement, over the entire 4,350 year period. Luckily, the 3D Land Bridge platform already contains a minimap which shows the general terrain plus caribou migration path. We can use that as a basis for producing still frames which we can later combine into a full video. All that is needed is to have the minimap also show hunting blind locations, and also include the year along with numerical information about the environment (such as the water level) somewhere else in the frame. Fortunately, the Microsoft XNA framework allows us to create screenshots of image portions of a running graphical program (and send them to an external file), and there are ways to add text within said screenshots as well. Unfortunately, the framework does not provide us with a way to combine said screenshots/frames into a video that can be run outside of XNA. However, if we instruct our system to make the filename of each frame the simulation year for which it was taken, then it will be very easy to place all the frame files in order using "File Explorer" type programs (natively available in all major operating systems), and then drop them into "Windows Movie Maker" type software (available either natively or for purchase in most major operating systems). The process of making the video with the software from the frames should take only a few minutes.

To fulfill the second objective, we need to produce a single map-grid of the land bridge area displaying in each tile the total number of hunting blinds that ever existed in that square during our simulation, in other words, a hunting blind heatmap. Although it really is almost impossible to say what the absolute probability of finding a hunting blind in any given location is, a heatmap will help us to produce relative probabilities, in other words to show the locations where a blind is *more likely to be* with respect to other locations. This sort of information is best displayed by shading tiles different colors relative to other tiles rather than placing probability numbers within tiles, so the former is what we will do. Archaeologists can use the map as an expedition aid to help decide which areas to go looking and which areas to spend less time in or ignore altogether.

4.4 Implementation

Pseudocode

The pseudocode for the main portion of the simulation is as follows:

//Initialization Steps
nCaribou = 99 */*No. of caribou that cross Land Bridge each generation is 99.*/*
nHuntingBlinds = 50 */*No. of blinds (i.e., population for the GA) is 50.*/*

beliefSpace.Initialize()
*/*The belief space is an influence map with a tile corresponding to each of the regular map tiles. Each belief space tile contains four parameters, each corresponding with one of the four factors described in the high-level algorithm (closeness to caribou, height above caribou, distance from nearest other hunting blind, and closeness to water). Discussed in further detail in later sections.)*/*

populationSpace.Initialize(numHuntingBlinds)
*/*This function initializes each hunting blind's chromosome, setting each binary bit within to a random value of 0 or 1. In this cultural algorithm, each hunting blind's chromosome consists of 16 binary bits, which are divided into 4 sets of 4 bits each. Each of these sets denotes a decimal integer which corresponds to one of the four weights that belong to this hunting blind and help to determine its actions in response to what it believes about the environment. The four weights, in turn, relate to the distance from a given square the closest caribou approach, the height of a given square above (or below) the closest caribou, the distance from a given square to the closest other hunting blind, and the distance from a given square to the nearest tile covered by water. (The weights themselves and the weight function process are more fully discussed in the next subsection, entitled "Weight Function.)*/*

//Main Loop
*/*A logical starting point is 9793 BCE, when Land Bridge is first traversable because glacier covering it has receded, but user can start sim at later points as well if desired.*/*
do

HBLocations = Simulation.GetHBLocs(population.genes, beliefSpace, WeightFunction)
*/*Determines the locations for the hunting blinds for this generation. If this is the first generation, locations are random. If not, locations are determined by a weight function which takes the hunting blind genes as an argument and is applied to each tile in the belief space in order to determine what the hunting blind buliders think is the most desirable spot for building the blind. (This is discussed in further depth in the "Weight Function" subsection.)*/*

Simulation.PlaceHuntingBlinds(population, HBLocations)
*/*Places the hunting blinds in the locations determined by the previous function.*/*

Simulation.Run()

population.ComputeActualScores()
*/*Computes a score for each blind based on each of the 3 important factors that determine success or failure described in the high-level design: distance to caribou, height above (or below) caribou, and closeness to the nearest other hunting blind. Any underwater blind, however, automatically receives a score of negative infinity representing its uselessness despite all other factors. (Here, high scores are considered good, while low scores are considered bad.)*/*

population.SortByActualScores()

beliefSpace.Update()
/**Updates belief space tile parameters with the hunting blind score parameters (plus closeness to water) for the tiles on which the hunting blinds were situated, and also for all tiles within a Moore radius of the blinds (representing the hunters' ability to speculate about what might have happened if they had chosen a slightly different location for their blind, as described in the high-level design.*/*

population.genes.Mutate ()
*/*Only the top four scoring hunting blinds are allowed to reproduce. All hunting blinds that scored below the top four become mutated versions of the top four in the next generation. Additionally, the top four themselves are not mutated at all. Granted, this is a very elitist approach, but ultimately it is one that works in this particular situation.*/*

population.genes.Crossover()
*/*A random point is chosen and each chromosome is divided into two parts with the random point as the pivot. All of the first portions of all of the chromosomes are placed into one list, and all of the second portions of all of the chromosomes are placed into another list. Each partial chromosome from the first list is randomly combined with a partial chromosome with the second list. This completes the creation of the new chromosomes for the next generation. Every chromosome is subject to crossover except for the best-scoring one.*/*

year = year + 1
generationNum = generationNum +1
map.Update() *//Updates the water level and the rest of the environment for the next year.*

until(end sim)
*/*A logical end point is 6343 BCE, when Land Bridge is permanently split by relentless water rise, but user can end sim at earlier points as well if desired.*/*

//End of Pseudocode

Weight Function

For each new generation, each hunting blind is placed within the tile containing the highest value of the "weight function", which is applied to each "known" tile in the belief space. The value of the weight function W at belief-space tile T is calculated as follows:

$$W(T) = -W_1B_1 + W_2B_2 + W_3B_3 + W_4B_4 \mid (W_1, W_2, W_3, W_4 \geq 0)$$

Equation 3: Weight Function

In the weight function, B_1 is T's value for the hunting blind's distance to the closest caribou approach, B_2 is T's value for the hunting blind's height above (or below) the closest caribou approach, B_3 is T's value for the distance to the closest other hunting blind, and B_4 is T's value for the distance to the closest underwater point. Recalling our high-level design, we can see how the weight function is crafted so that a tile is deemed less desirable if it is far from the closest caribou, but more desirable if it has a high vantage point above the closest caribou, is far from the nearest other hunting blind, and/or is far from water. Exactly *how much* more or less desirable is determined by W_1 through W_4, which are the weights for each of the B's, respectively. Their values are determined by the hunting blind's chromosomes. It is through these four W-values that the chromosomes determine how important the hunting blind's builders consider each of the key factors, which together with their beliefs about the values of each key factor (represented by the B-values), ultimately determines the choice of blind construction location for the given generation.

"Compute Actual Scores" Function

The actual score of a hunting blind H for the round is computed by the following function:

$$F(H) = -C_1A_1 + C_2A_2 + C_3A_3 \text{ (if H is above water), OR}$$
$$F(H) = -\infty \text{ (if H is underwater)}$$

Equation 4: "Compute Actual Scores" Function (Fitness Function)

In this function, A_1 is the distance between H and the closest caribou approach point, A_2 is H's height above (or below) the closest caribou approach point, and A_3 is H's distance from the nearest hunting blind. Recalling the high-level design, each of these A's represents one of the crucial factors which determine the success or failure of a hunting party using a particular blind. C_1, C_2, and C_3 are constants which reflect how important each of the three factors are compared to one another[1]. The objective is to maximize F(H); the highest scores are considered the best whereas the lowest are considered the worst. Generally, the score becomes lower if the closest caribou was very far away from the blind, but higher if the blind has a high vantage point above the closest caribou and/or if it is far from the nearest other blind. Underwater hunting blinds of course receive the worst possible scores. The genes of the highest-scoring hunting blinds receive the reward of being favored in the reproduction process for creating the next generation, while those of the lowest-scoring hunting blinds are punished by being kept out of the reproduction process.

Future Work on Weight Function

On the land bridge, building a hunting blind close to water was indeed often a bad choice due to the threat of a rapid water rise inundating the blind (spoiling a significant amount of work), and as mentioned before our current weight function reflects this. However, this need

[1] In the proof-of-concept run (to be discussed in the upcoming chapters), C_1 was -100, C_2 was 60, and C_3 was 30. In the full run (also to be discussed in the upcoming chapters), C_1 was -30, C_2 was 50, and C_3 was 8.

not always be the case, as locations close to water often have quite bountiful vegetation, especially given that Lake Huron is (and was) a freshwater lake, and this vegetation can certainly attract many caribou. Thus, at times when the lake level is generally receding rather than rising, it may be wise for hunters to place blinds close to the water to take advantage of likely caribou paths along the vegetation-rich shoreline. Reflecting this, in future versions we will provide the hunters with the ability to make an "educated guess" about whether the water level is rising or falling. (The mechanism for making such guess is still under design). If the hunters feel it is falling, then in the weight function we will allow them to make the value for W_4 negative, thus making the factor W_4B_4 positive and thus making locations close to water generally more desirable than locations far from water (rather than the reverse, which is usual). In other words, that group of hunters will be taking the chance (usually a good one, but not always) of believing that the water level will continue to fall or stay stable, and thus believing that the closer a blind location is to water, the more opportunity to reap the benefits of the caribou that come with ample waterfront vegetation, rather than merely risking hunting blind inundation (usually true, but not always). The addition of this functionality should make the overall simulation even more accurate.

CHAPTER 5: EXPERIMENTAL FRAMEWORK

In April 2012, the team working on Palazzolo's Land Bridge Simulation provided 400 component maps which together comprise an entire map of Area 1 of the Alpena-Amberley Land Bridge, the portion currently under the most intense archaeological study. Each of these component maps has 999,995 data points in it (giving a total of 399,998,000 points in all). The team created the component maps using a tool on the NOAA website which generates bathymetry maps for the region, and then interpolating in even more points.

Luckily, Thomas Palazzolo's land bridge program, which serves as the basis for this system, has the ability to combine points to make a simulation manageable on an ordinary mass-market computer.

The most prudent course of action seemed to be to first design a "proof-of-concept" simulation, then to do a full simulation involving the entirety of Area 1. The following portion of Area 1 of the land bridge (designated by the red square in the following picture) is quite an interesting spot, and is what we have chosen for our proof-of-concept. Our proof-of-concept area stretches from 378000E to 378995E, and from 4971005N to 4972000N.

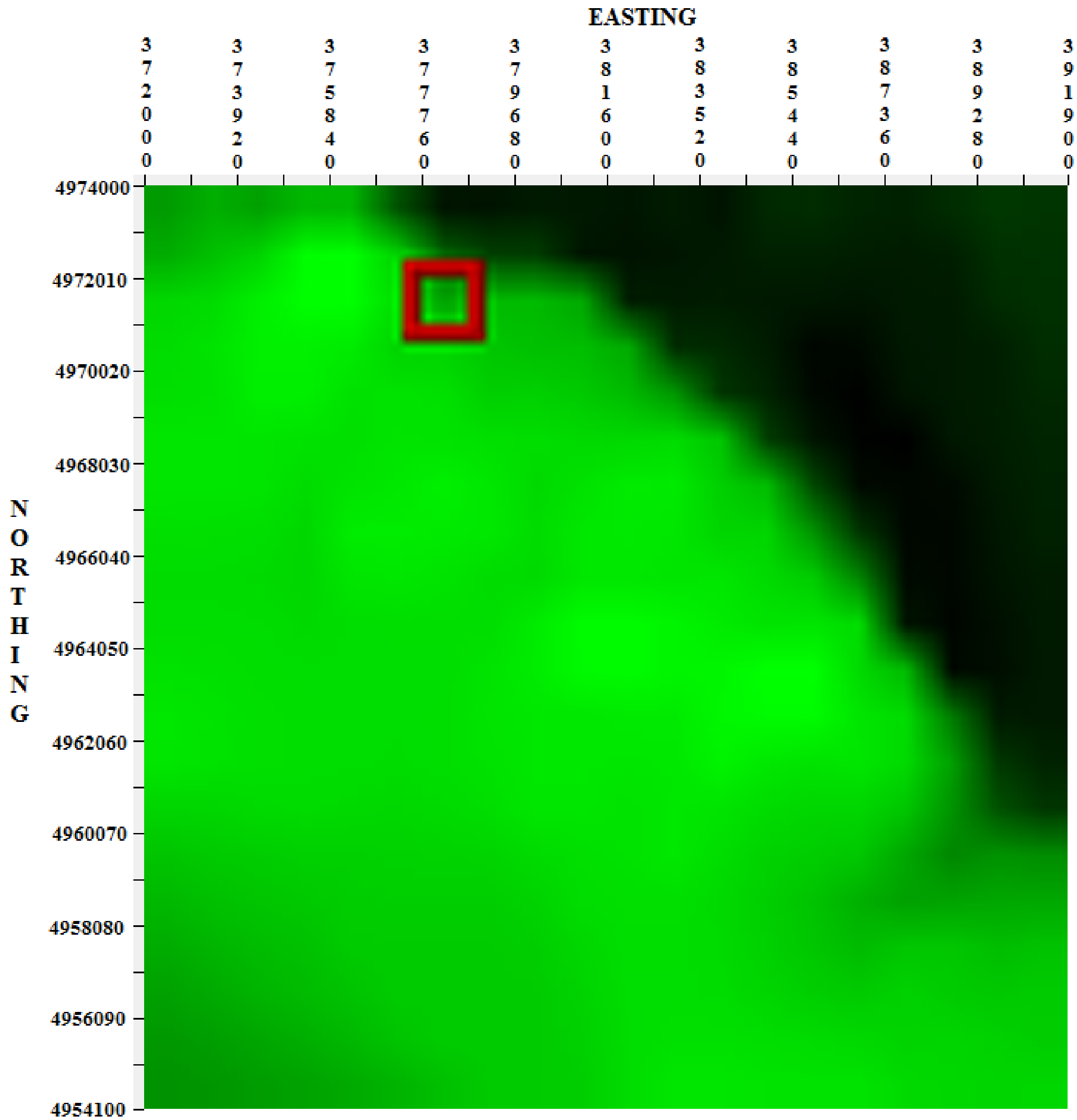


Figure 13: Relevant Portion of Area 1 of Alpena-Amberley Land Bridge

Even with this reduced landscape, given the constraints of computer memory and computation time, we have to reduce the number of points from 1000 x 1000 to 250 x 250. This compromises accuracy somewhat, but not to an intolerable degree.

CHAPTER 6: RESULTS

In this chapter are included numerous screenshots showing various milestones of the algorithm's development of intelligent behavior, as well as the algorithm's response to various difficult situations produced by environmental change. First we will show the results of the proof-of concept simulation, then the full simulation.

6.1 Proof of Concept

Parameters

In the proof of concept, we use a population of 50 hunting blinds and a caribou population of 99, and we do just one run over the full period from 9793 BCE to 6343 BCE.

9443 BCE

Although the land bridge itself initially becomes crossable in 9793 BCE, not all portions are crossable at that exact time, and many don't become crossable until later. Our proof-of-concept area happens to be such a portion. 9443 BCE is the first time that there is a viable path across it, i.e., a land path from the left to the right edge of the map that is not blocked by water. Since this is our "zeroeth" generation, the placement of the hunting blinds is completely random. Note that in this and in all other snapshots of selected years, the white curve and black dots represent the caribou path and hunting blind locations during the given year, respectively. The highest-scoring hunting blind has its numerical coordinate location and its score listed, and is designated by an orange rectangular overlay around the black dot indicating its location.

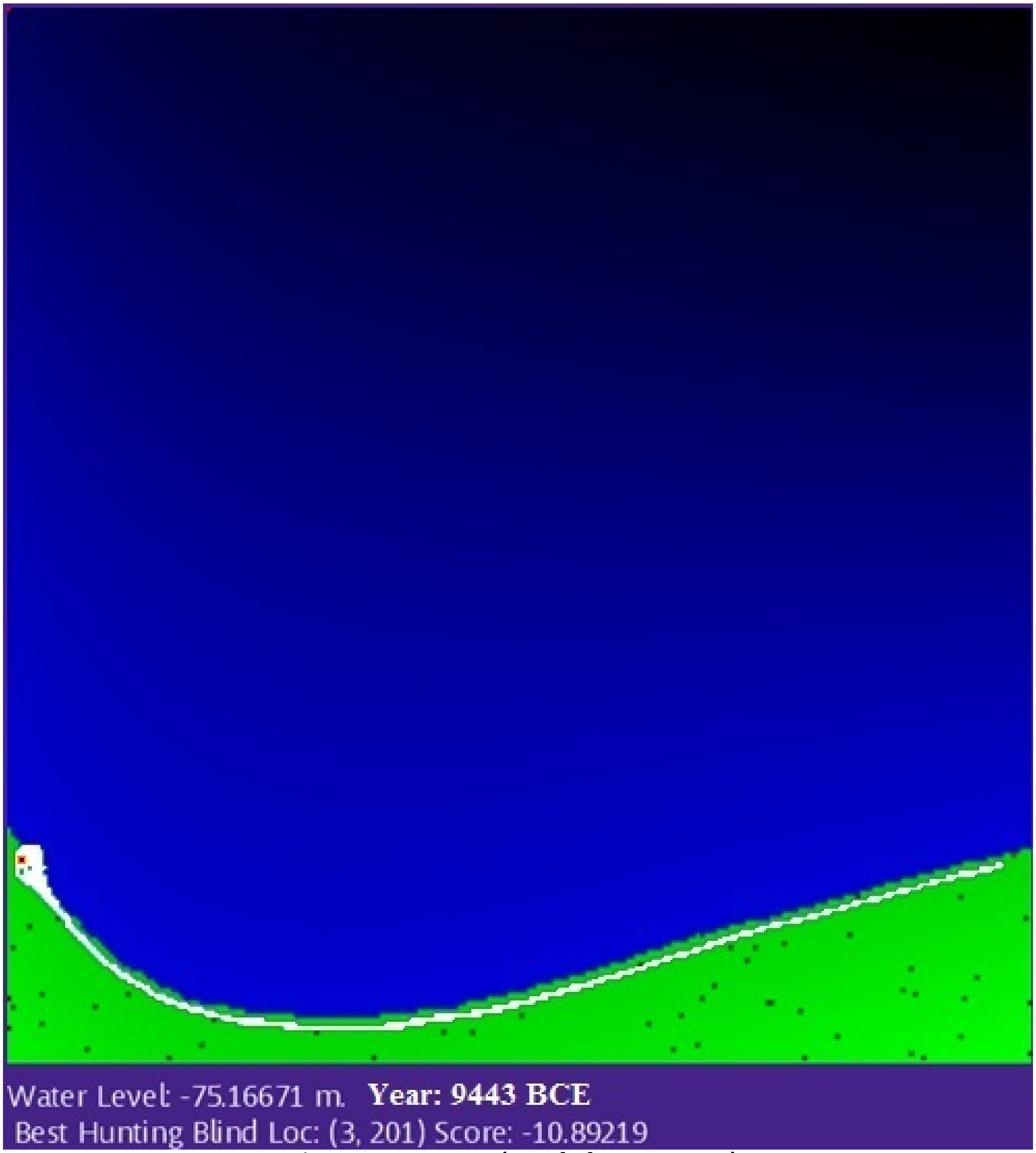


Figure 14: 9443 BCE (Proof of Concept Run)

9436 BCE

As one can plainly see, the algorithm is somewhat "stupid" just 7 generations into the simulation. When it first starts out, it has an especially hard time dealing with the fact that the caribou path changes rapidly in response to the rapidly changing waterline and vegetation patterns. Also it has not yet learned that clustering many hunting blinds close together is bad and that it ought to avoid doing this, hence the upside-down L-shaped cluster of hunting blinds toward the bottom center right.

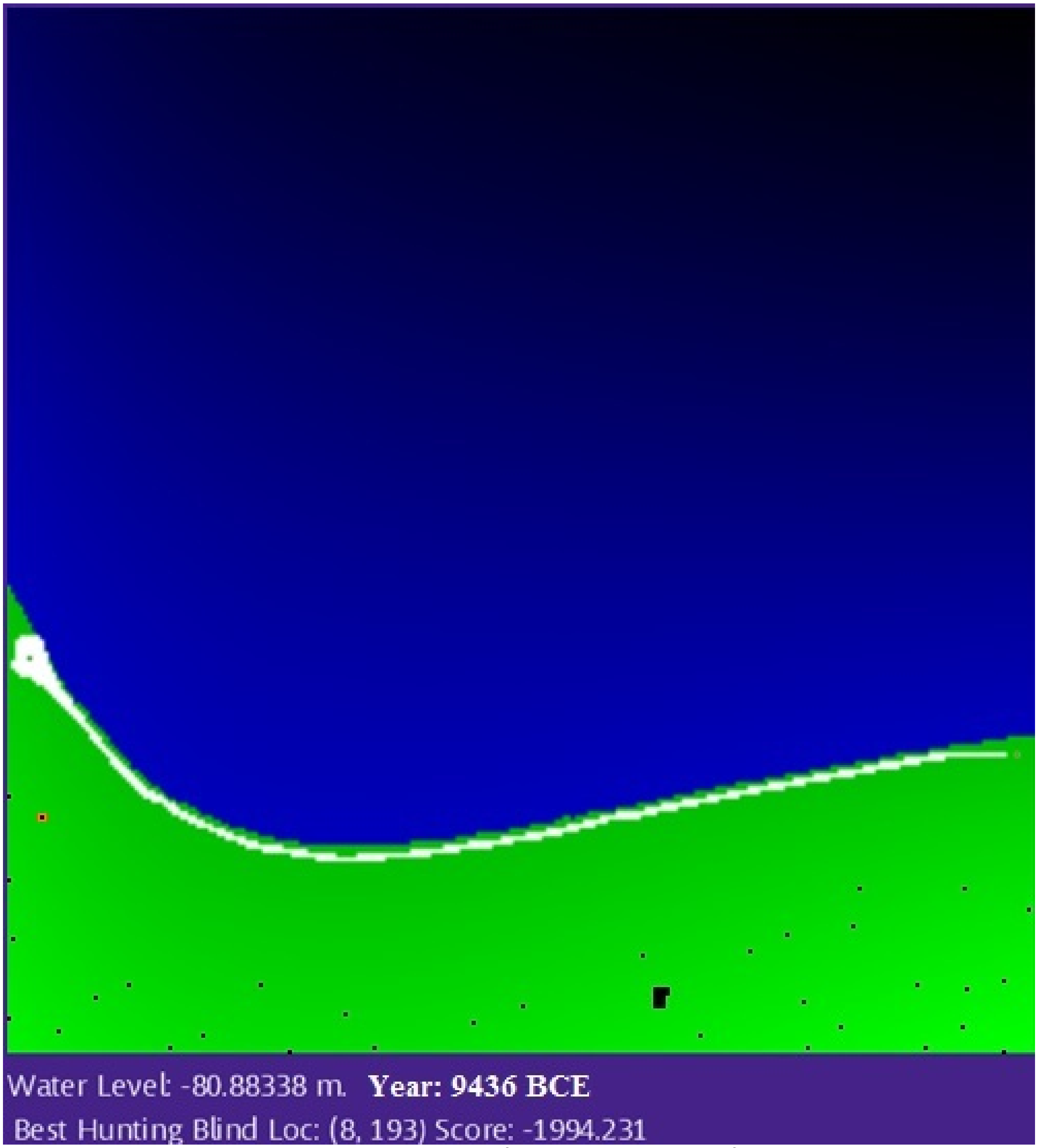


Figure 15: 9436 BCE (Proof of Concept Run)

9393 BCE

50 generations into the simulation, we can see that the algorithm is still rather stupid. It still has not figured out that clustering hunting blinds super close together is a bad idea. Despite this, here we do see the first signs of intelligent behavior. Notice how the region of the graph that the hunting blinds are clustering in is quite high elevationwise (shown by the fact that the map in this region is a light green). As mentioned in the design chapter, having an elevated venue above the caribou does earn points, although the blinds would earn more were they more spaced out and closer to the caribou.

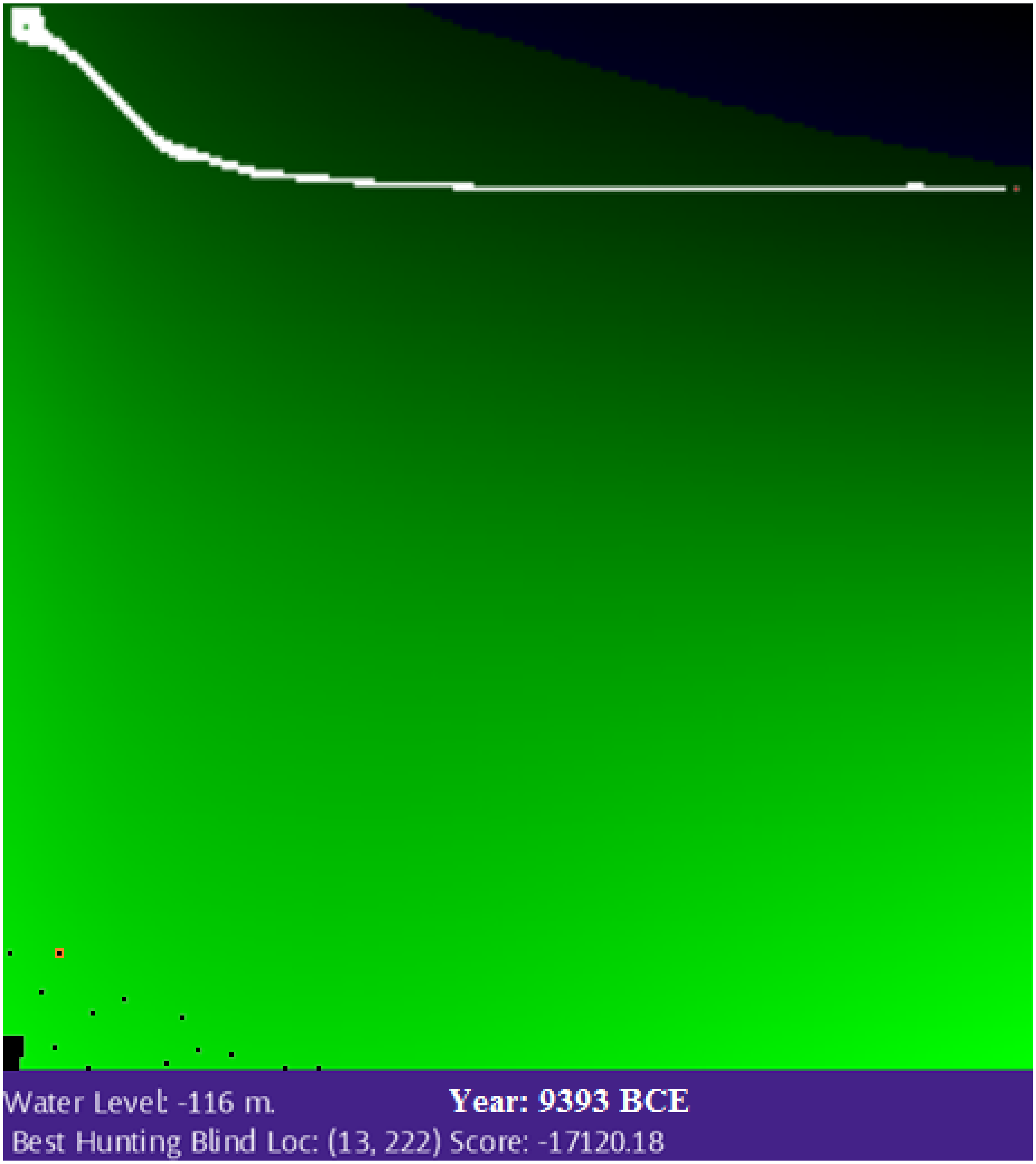


Figure 16: 9393 BCE (Proof of Concept Run)

9293 BCE

Now we are seeing more signs of intelligence. The hunting blinds have mostly discovered that clustering together is bad, and are starting to discover that moving closer towards the caribou is a good idea. Notice how the hunting blinds that are moving closer to the caribou have retained the knowledge that having a high vantage point over the caribou is a good idea, and are thus using the hill on the right edge if the screen to move closer rather than the roughly U-shaped valley. Other hunting blinds have started venturing over to the lower left corner which contains the highest peak on the entire map.

Figure 17: 9293 BCE (Proof of Concept Run)

9193 BCE

Now the AI behavior is becoming truly intelligent. One group of hunting blinds is sticking with the prior strategy of seeking out the highest points of the left ridge, while another is trying to get as close as possible to the actual caribou path. A few are trying a "hybrid strategy" of using locations still on the ridge although somewhat lower, but closer to the caribou path. Notice also how that by this point, the algorithm has completely figured out that it's a bad idea to cluster hunting blinds right next to one another, so it's no longer doing this at all.

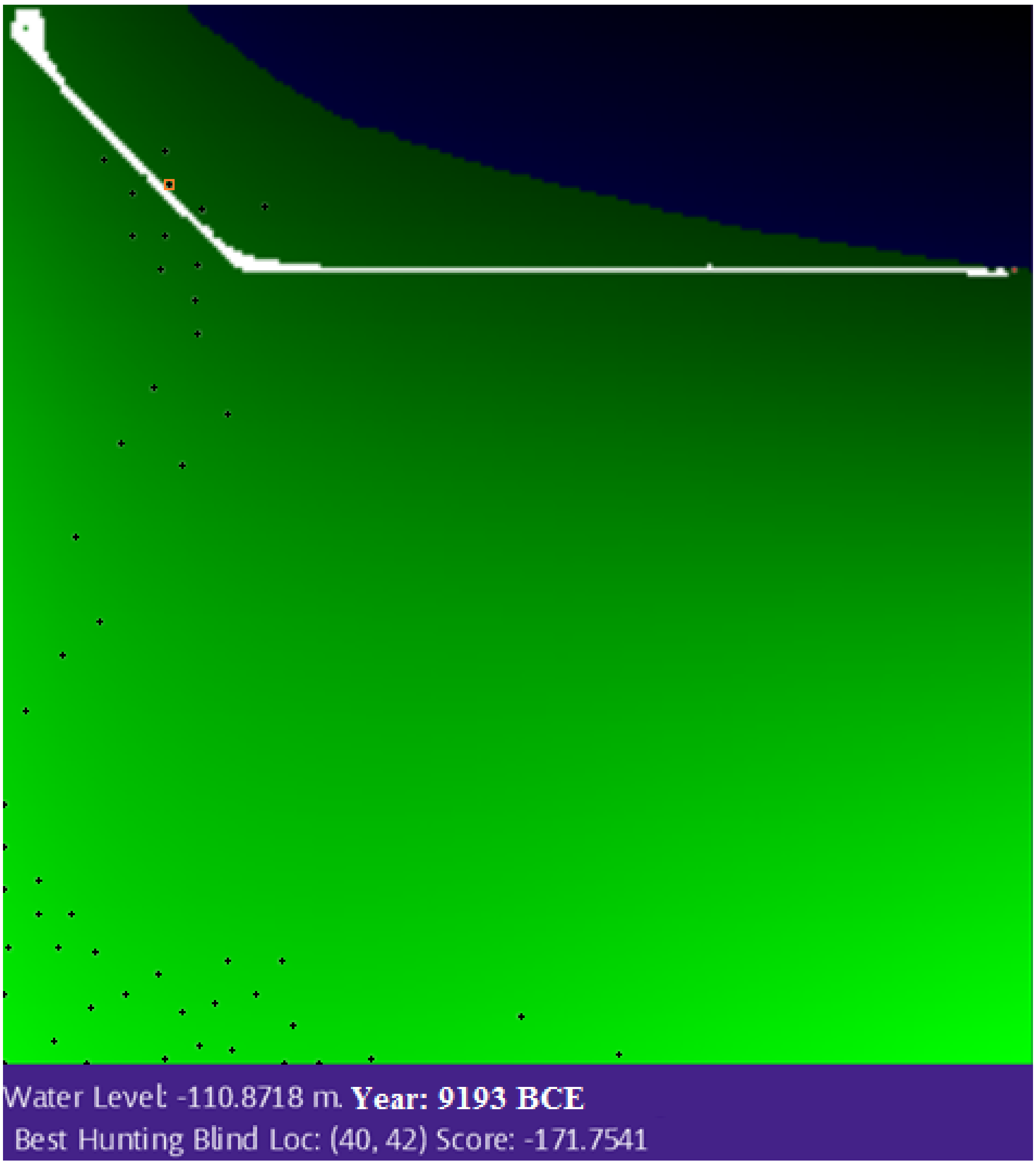


Figure 18: 9193 BCE (Proof of Concept Run)

8643 BCE

Now the algorithm has to deal with another difficult challenge: rapid rise of the water level. This means that not only does the caribou path change by moving rapidly closer to the south edge, many formerly reliable locations are quickly becoming submerged and therefore unusable. The algorithm is responding to the situation by having the hunting blinds flock to the safe high ground toward the lower right edge of the map. (Indeed, this is the highest ground on the entire map.)

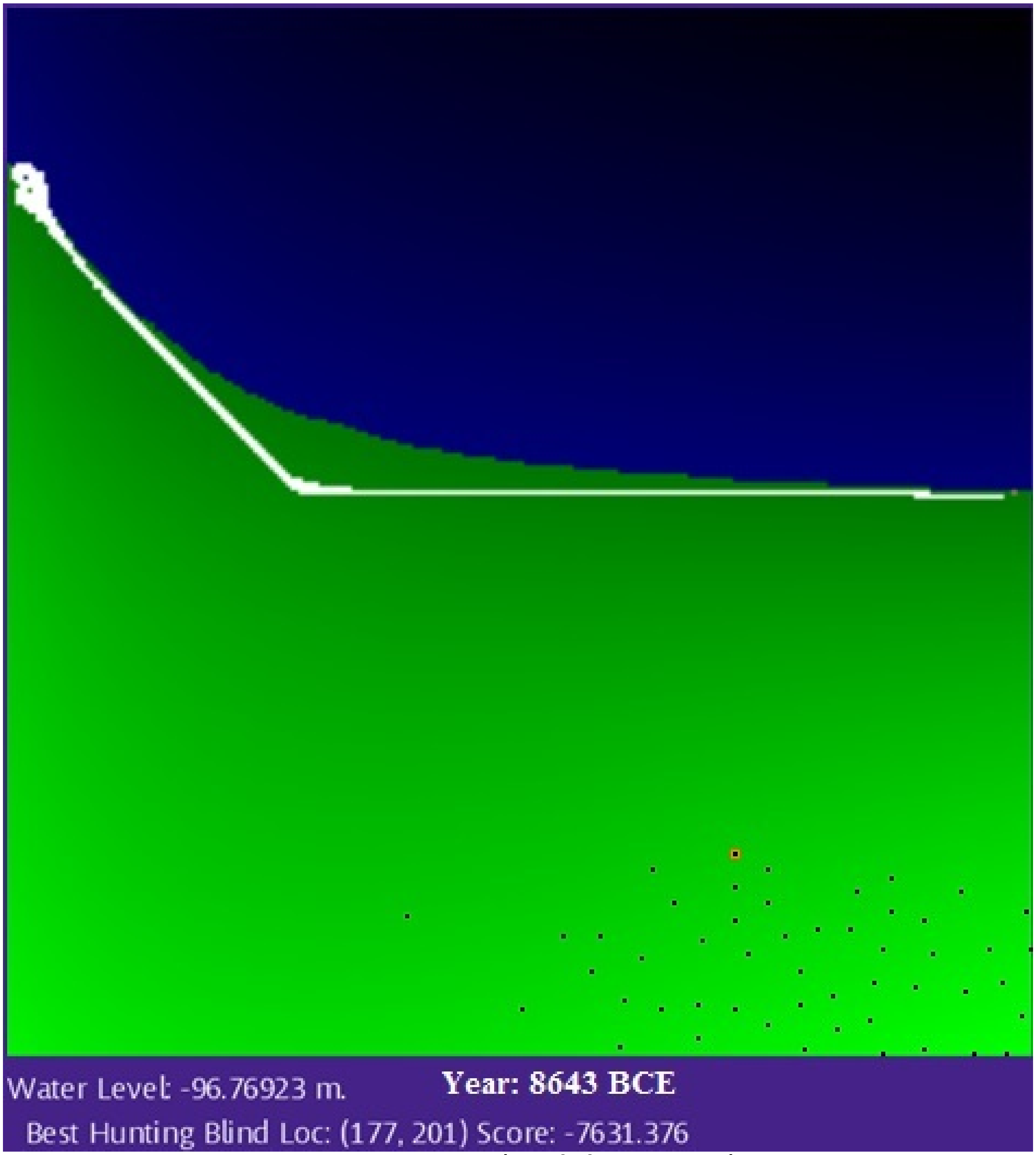


Figure 19: 8643 BCE (Proof of Concept Run)

8493 BCE

Some of the hunting blinds have now decided to take a risk despite the rising water and situate themselves close to the water -- and the caribou -- for a chance at a higher score.

However, it seems that a bunch of them all decided to do this at once and chose adjacent spots, which will cost them points.

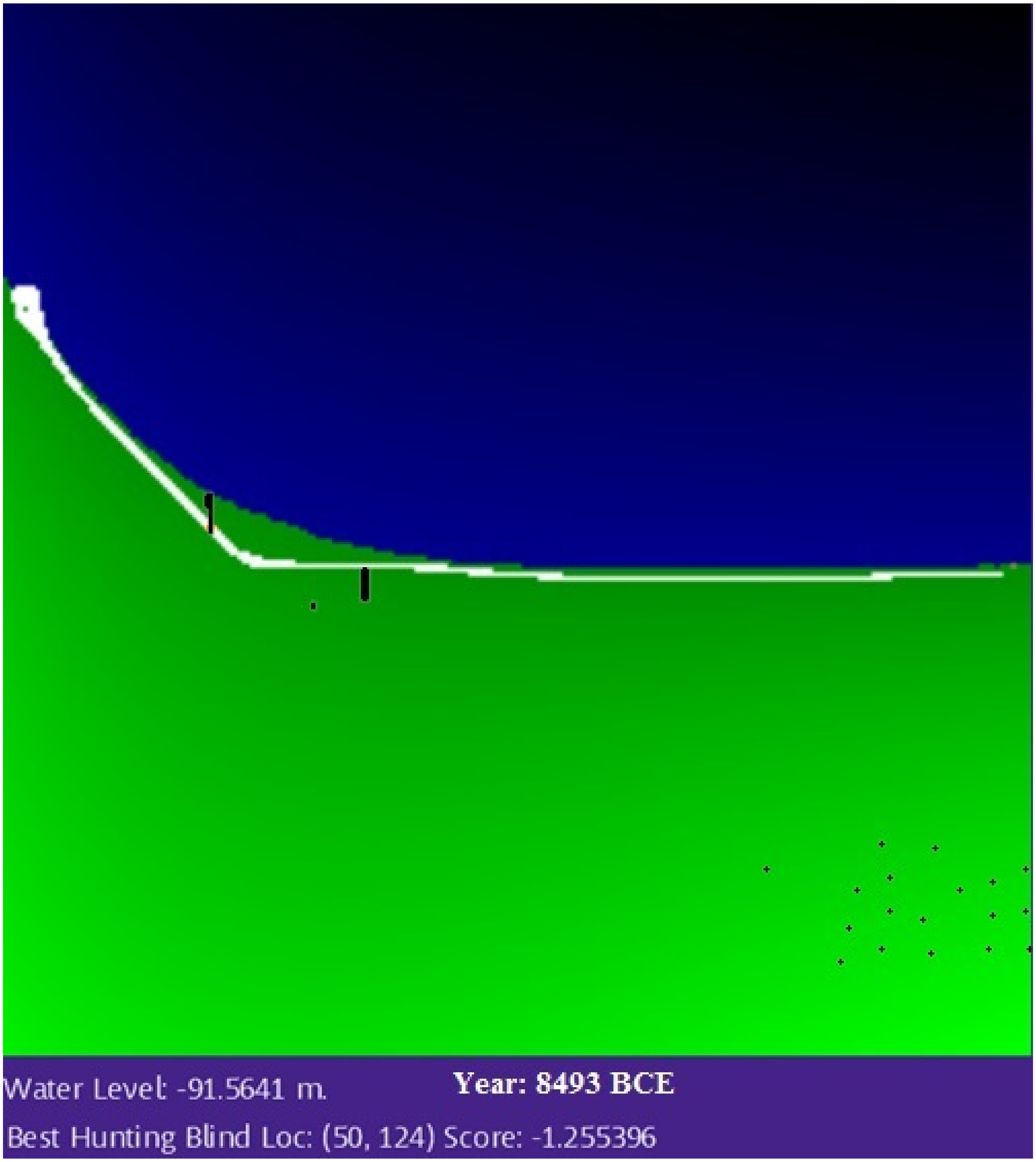


Figure 20: 8493 BCE (Proof of Concept Run)

7948 BCE

The water level is still rising. Less and less land is available for hunting blind placement with each new generation. Most of the hunting blinds are sticking to the high ground, with a few risking total inundation for the sake of outscoring the ones who are taking the "safe" strategy.

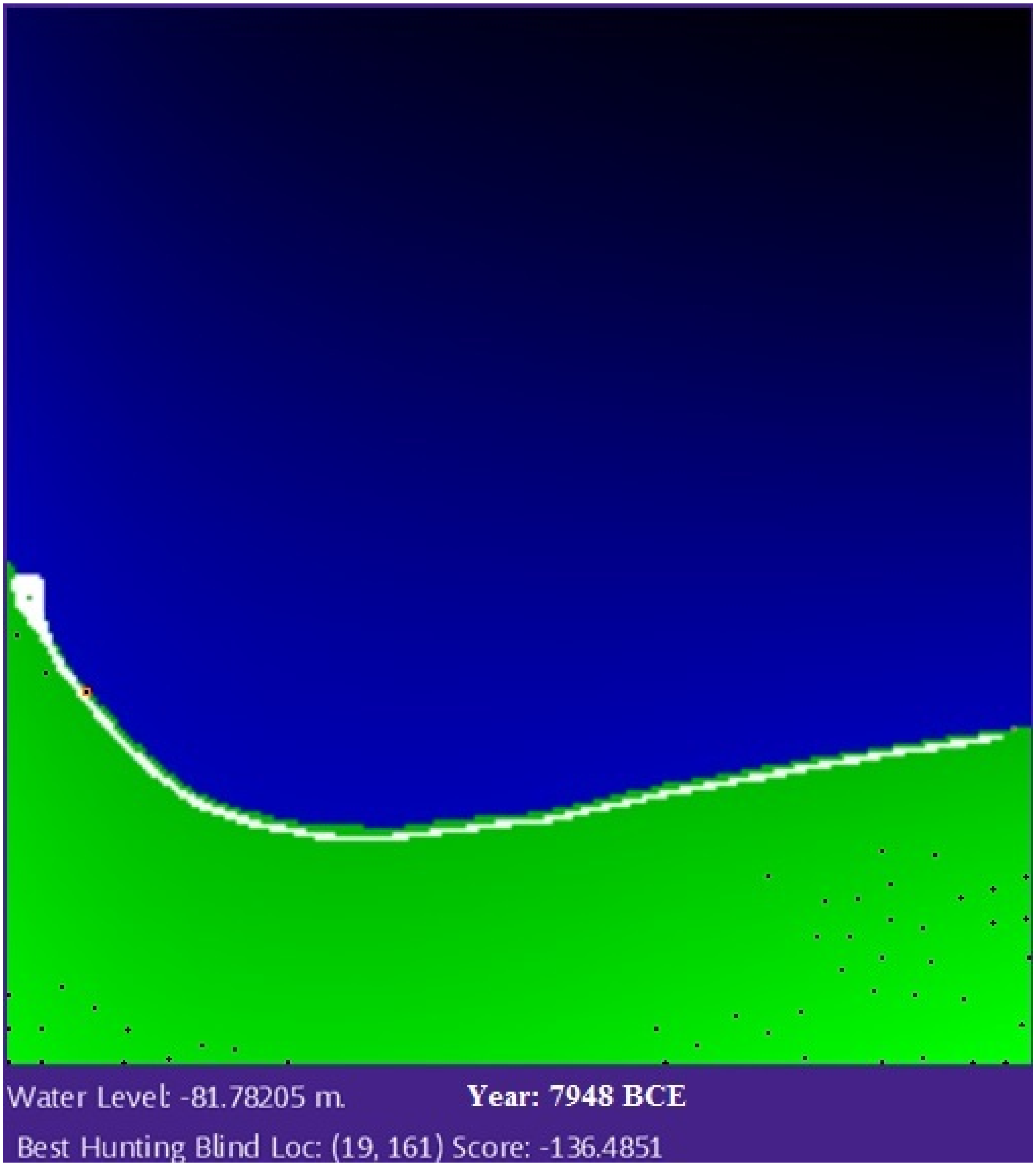


Figure 21: 7948 BCE (Proof of Concept Run)

7794 BCE

The water is now rising extremely rapidly, and soon this portion of the land bridge will be uncrossable. This is the last year that the caribou attempt to make a crossing. With the exception of one straggler, the hunting blinds have all situated themselves on the highest, safest ground, located in the bottom-right portion of the map.

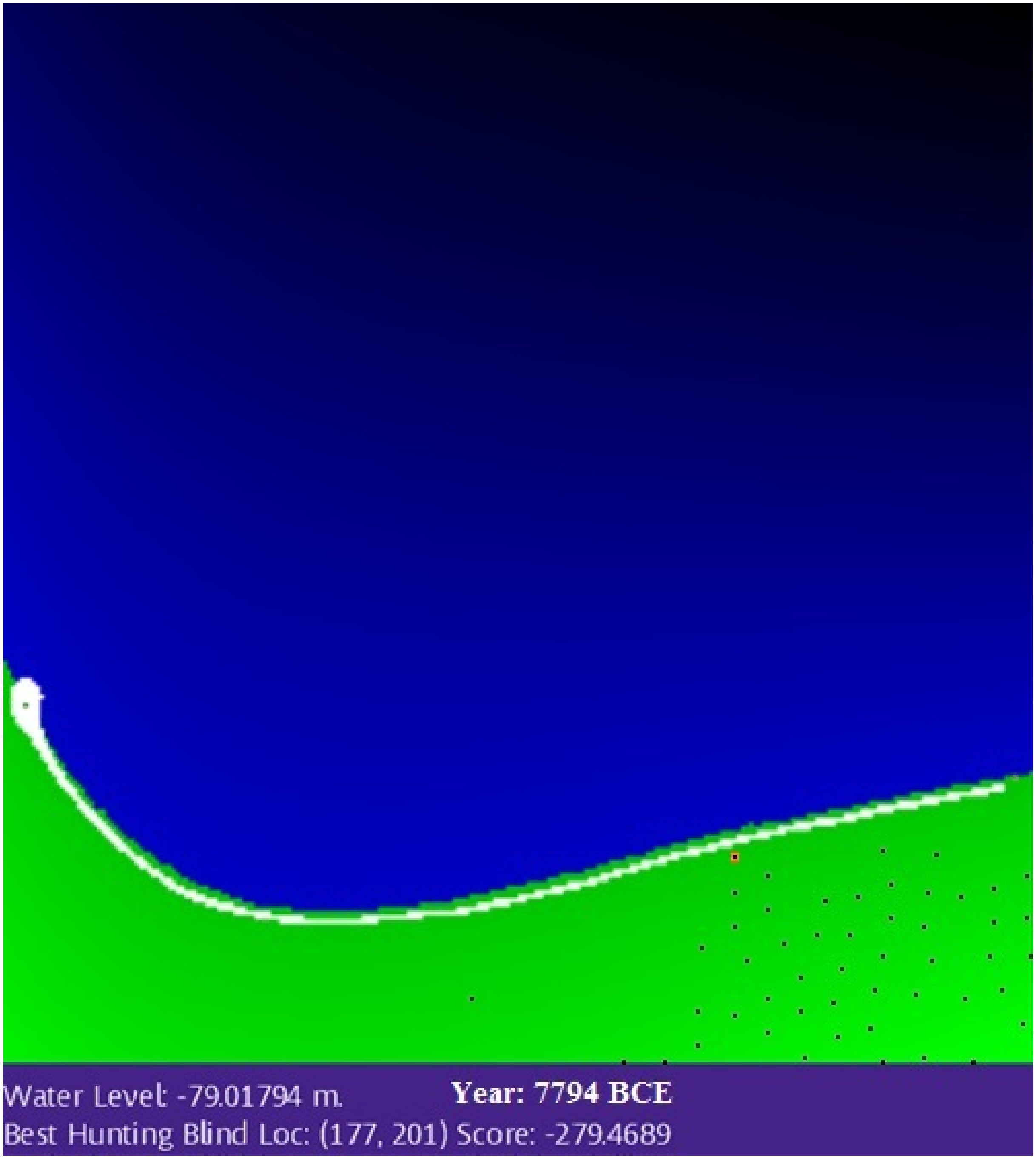


Figure 22: 7794 BCE (Proof of Concept Run)

7791 BCE

Although the Alpena-Amberley Land Bridge itself is crossable until 6343 BCE (as discussed in a previous chapter), not all portions are crossable for that long a time. Our proof-of-concept location is one such portion. Due to rising water levels leaving no viable path, 7791 BCE is the third year in a row that the caribou have not even attempted to cross through our proof-of-concept area, and conditions here will only get worse as time continues to pass. We therefore end the proof-of-concept simulation here.

Figure 23: 7791 BCE (Proof of Concept Run)

Heatmap for Proof-of-Concept

As mentioned in a previous chapter, the heatmap shows the number of hunting blinds in a certain square relative to other squares. The larger a dot, the more hunting blinds that were found within that particular location over the 3,450 year period. Looking at the heatmap, we see that it is roughly L-shaped with a slight bulge toward where the "bend" of the caribou path was when the water level was low. The lower portion, and especially the lower right portion, of the "L" is the darkest because these portions were above water longer than anywhere else.

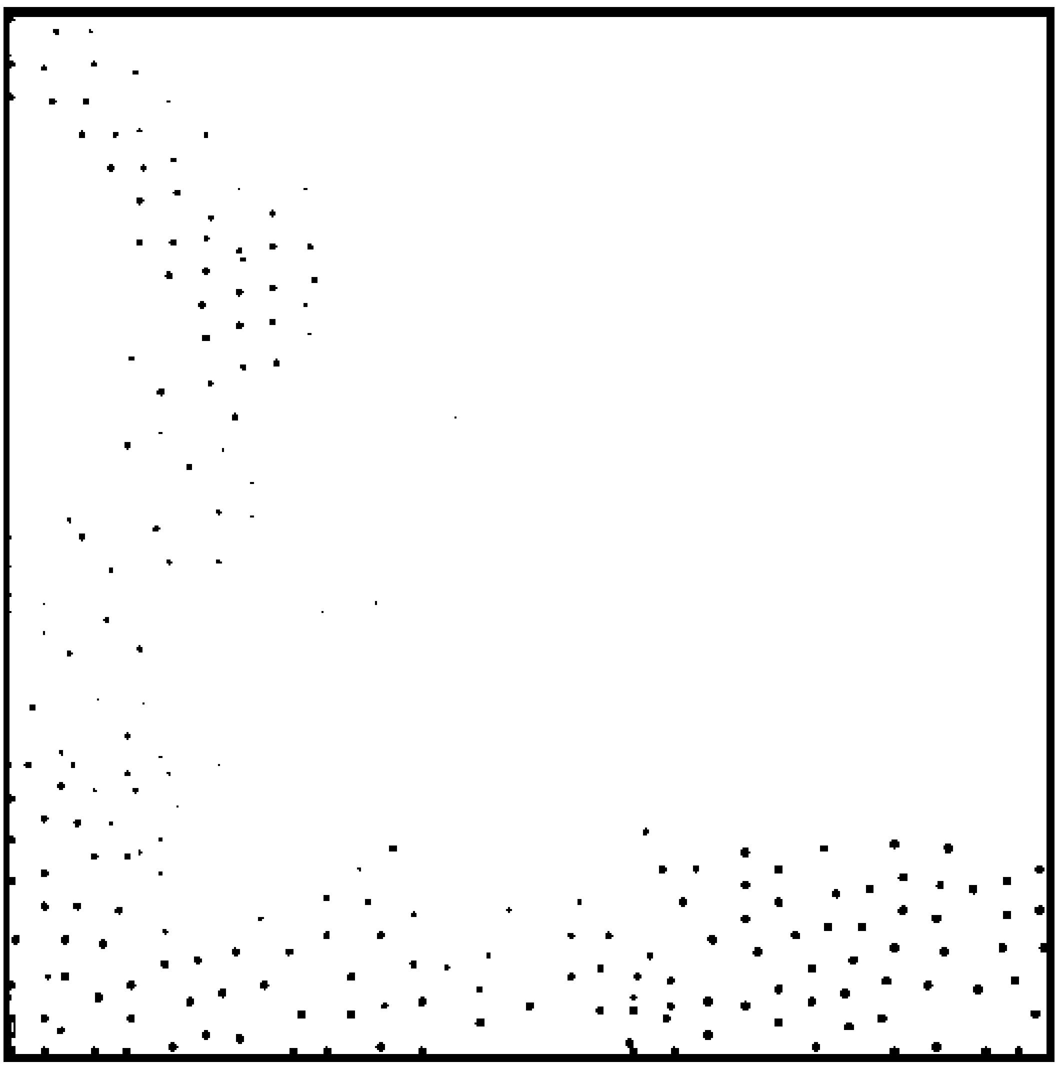


Figure 24: Heatmap (Proof of Concept Run)

6.2 Full Experiment

In January 2013, the author of this work was informed that there was interest in actual corroboration of Dr. O'Shea's findings with results generated by this model. At that point, the author of this work began devising a full experiment to be run over the entirety of Area 1. Once that experiment had been set up and was ready to be run (in early February, 2013), the author asked the team for the actual coordinates of Dr. O'Shea's team's hunting blind findings. Upon hearing the author's request, Thomas Palazzolo provided the author with (4964407.461 Northing, 0381773.819 Easting) as the exact coordinates of the "Funnel Drive Structure", a structure which contains 2 to 3 hunting blinds arrayed in a strategic fashion.

The full experiment consists of 16 simulation runs over all of Area 1 over the whole of the 9793 BCE to 6343 BCE period. Our objective now is to check our results against the Funnel Drive Structure's location. As we've already visually demonstrated the evolutionary process in the proof-of-concept section, for the sake of space we are going to show slides for only a few individual years of the first run before displaying the overall results of the experiment.

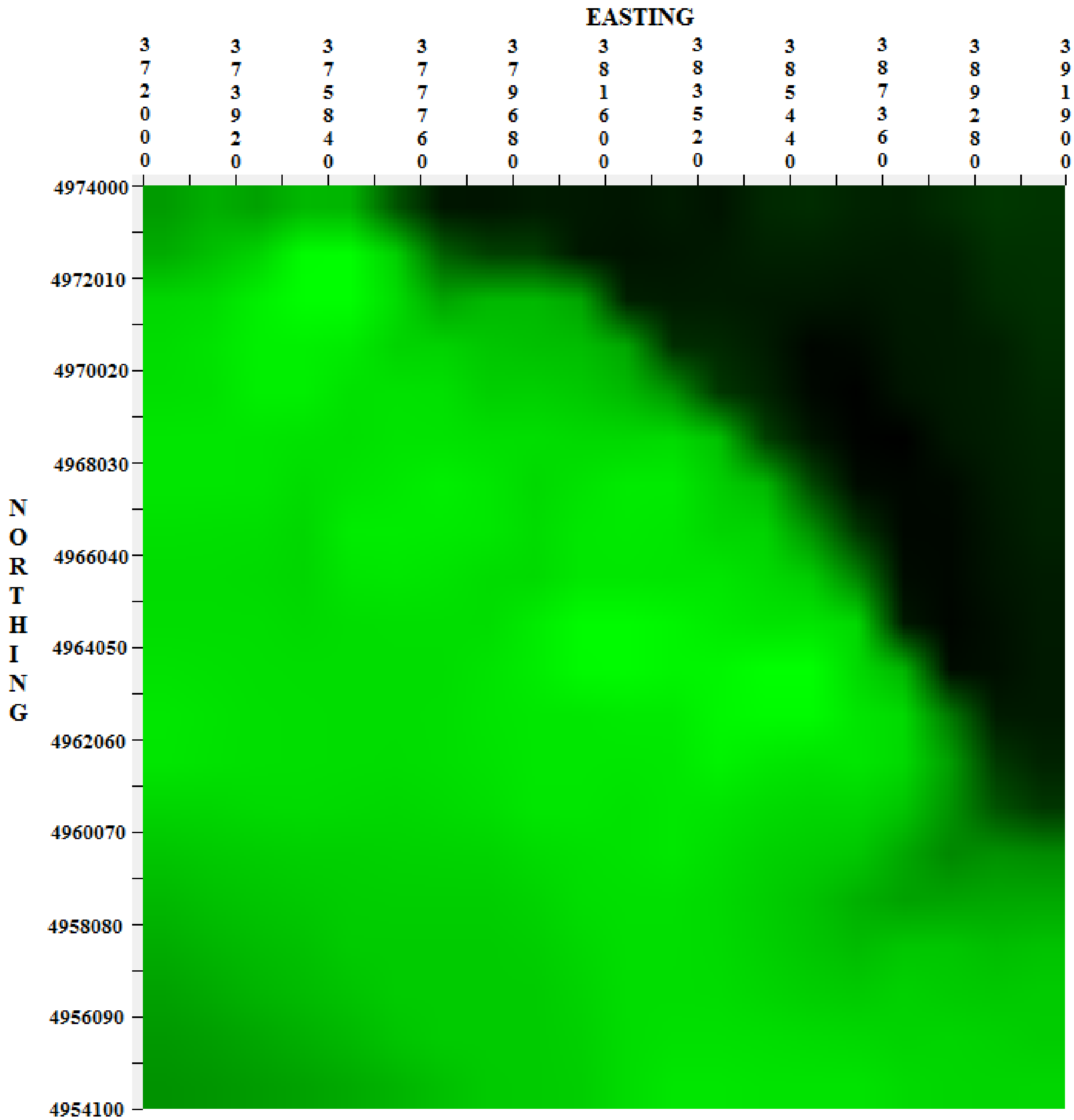


Figure 25: Area 1 Map with Coordinates (Map for Full Run)

Parameters and New Rules

Due to the fact that we were doing 16 runs for the full experiment, and because of the huge amount of time it takes to run one full 3,450 year simulation with one-year intervals, we are going to use five-year intervals for the full simulation. This means one full simulation now encompasses 690 generations rather than 3,450. To further reduce memory usage and computation time, we are condensing all the map points into a 200 x 200 grid (40,000 points in total) rather than the 250 x 250 grid (62,500 points in total) that we used in the proof of concept.

Also, after seeing the results of the proof-of-concept run, we decided to implement the additional rule that the hunting blinds cannot be *directly adjacent* to one another. Any two hunting blinds must now have at least one empty square between them. This seems a reasonable measure to prevent the severe clumping that we saw during some of the earlier years during the proof of concept run, where large numbers of blinds would form a solid "block" around a desirable area. Although a severe score penalty for being too close to another hunting blind did eventually convince them not to clump directly adjacent to one another anymore, the fact still is that such close clumping would never occur in real life, which is why it is being completely disallowed for the main experiment. Note that all other aspects of the regular "closeness" penalty still apply, a hunting blind that is only just a few (but more than one) squares away from another, although this is still allowed, still receives a hefty penalty for being too close to another blind, and the blinds eventually figure out that they must keep a reasonably healthy distance between themselves, just as they did during the proof-of-concept run.

Also, we decided to implement a "forgetting" ability for the belief space. If a belief space square has not met the requirements for an update for a certain number of generations (excepting generations when the land bridge is flooded) either by having a hunting blind directly in it or somewhere within the proper Moore radius of a hunting blind, then that square's knowledge gets completely forgotten. (In our full experiment, we set this forgetting threshold to 10 generations with no update.) This had the effect of preventing stragglers from choosing squares with obsolete knowledge, and generally it made the algorithm learn faster.

Also, we decided to change the fourth term of the weight function, the "water fear" factor, to be logarithmic. This is because if a hunting blind is already very close to water, the risk of being swamped by rising water is much greater than if the hunting blind is quite far away. In other words, the net benefit of moving, say, 50 feet away from rising water when one is currently only 10 feet away from the water is much greater than moving 50 feet away when one is already miles away. In the former case, the benefit is crucial, in the latter, it is nugatory. The new weight function for a given tile T is thus:

$$W(T) = W_1B_1 - W_2B_2 - W_3B_3 - Log_{10}(W_4B_4) \mid (W_1, W_2, W_3, W_4 \geq 0)$$

Equation 5: Modified Weight Function Used in Full Experiment

Overall, we have found that this change seems to have made the algorithm a lot smarter and quicker to learn.

Finally, for the full experiment, the population number of hunting blinds is 50, and the number of caribou per generation is 48. We will now show a variety of year frames from Run 1 of 16 to demonstrate our full experiment in action.

9793 BCE

9793 BCE is when the glacier initially recedes from the land bridge and reveals it as a crossable path. In each of these frames, as before, the white curve designates the caribou path, the black dots designate the AI hunting blinds, and an orange rectangle surrounds the highest-scoring hunting blind. Since this is the full run with the full map, a purple rectangle now designates where Dr. O'Shea found the Funnel Drive Structure. As with the proof-of-concept run, during the first generation, hunting blinds are simply placed in random non-water squares.

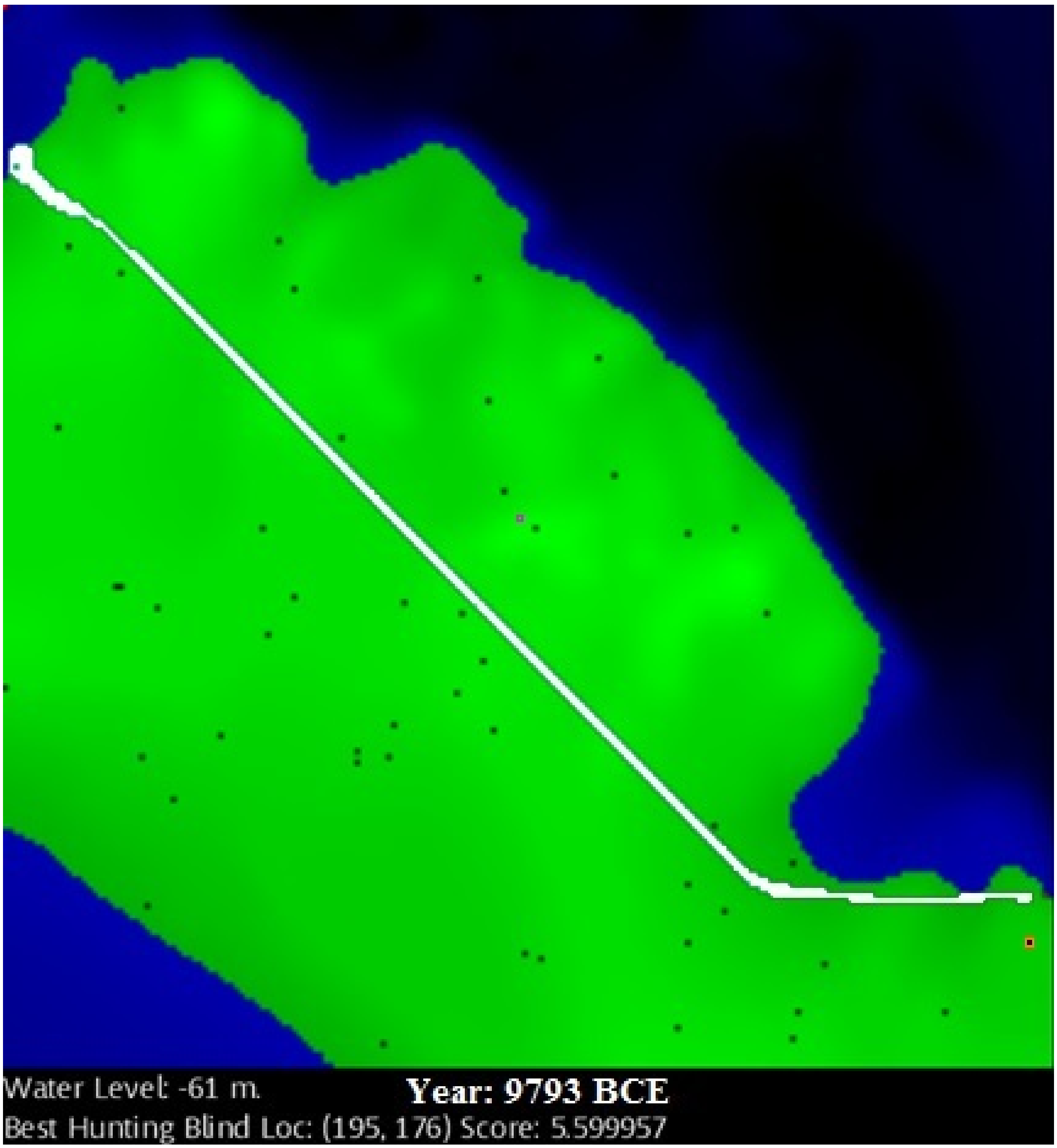


Figure 26: 9793 BCE (Full Run)

9743 BCE

As mentioned before, the algorithm learns much quicker this time around. After just 10 generations (50 years), it has learned to have the hunting blinds tightly track the caribou trail. However, it has not yet learned to keep the hunting blinds at a healthy distance from one another, and hence many of the hunting blinds are still losing a lot of points from being too close to another.

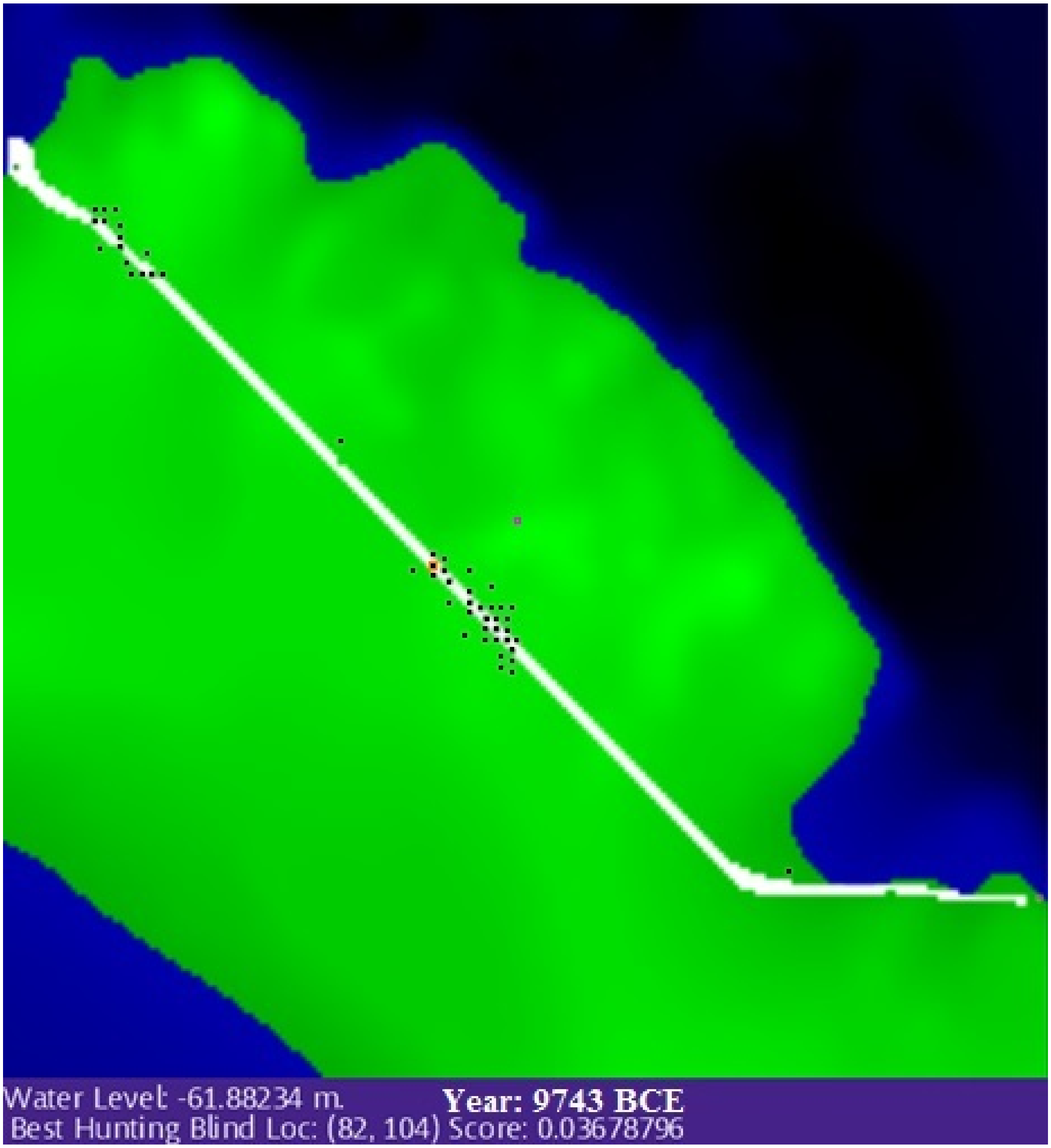


Figure 27: 9743 BCE (Full Run)

9693 BCE

Now the hunting blinds have learned to space out adequately from one another, as well as to stay close to where the caribou path is most likely to be. You can see a few also seeking out high ground in order to gain extra points for having a vantage point above the caribou.

Most of the results from the individual generations from here on out look more or less similar to this picture.

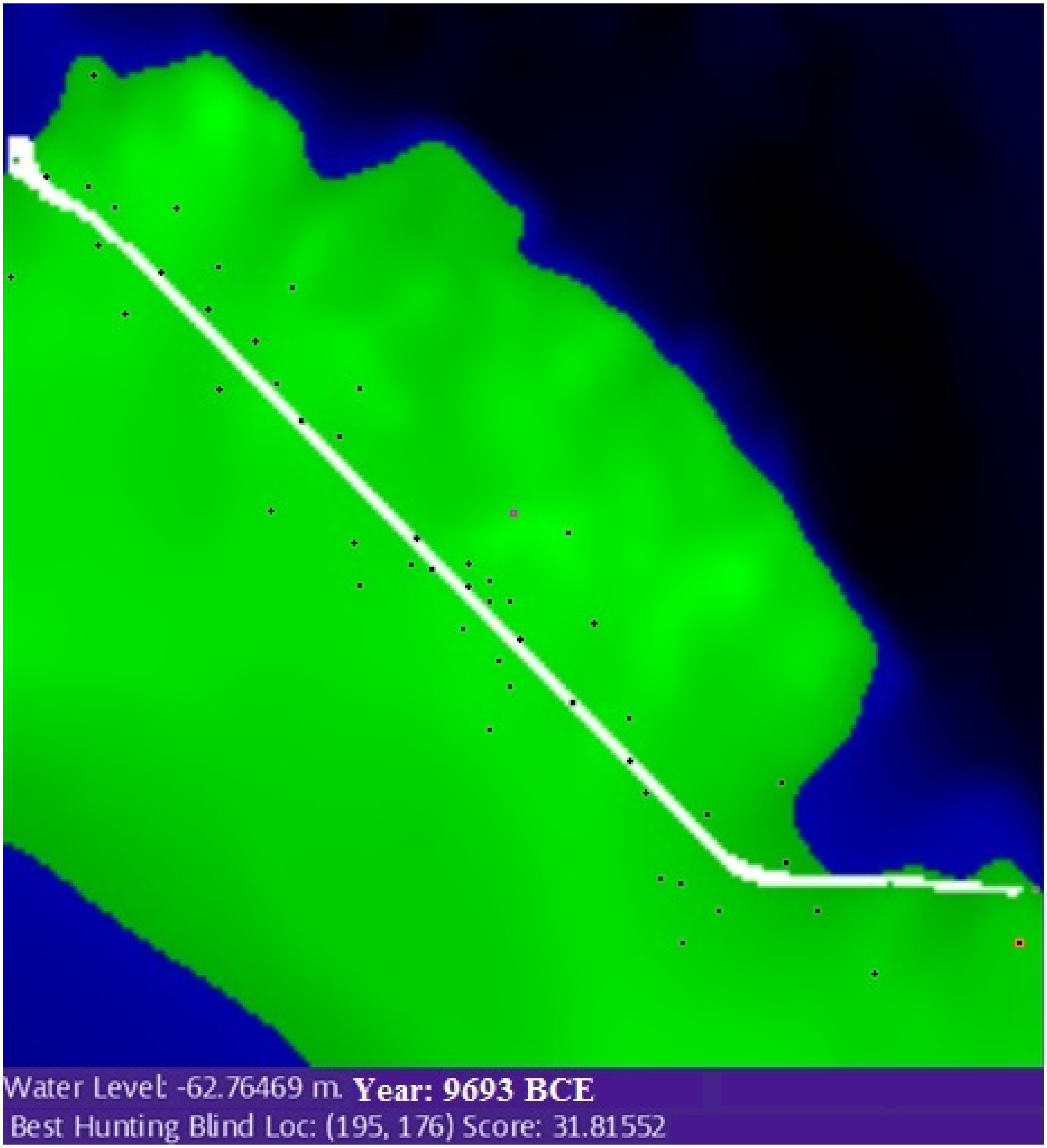


Figure 28: 9693 BCE (Full Run)

9363 BCE

When the simulated hunting blinds choose the spot where Dr. O'Shea found the Funnel Drive Structure, it is almost always during the Early Stanley and Mid Stanley lowstand periods, which run from about 9423 to 7993 BCE. That is when the water level is the lowest, and the caribou path responds by running very close to where the Funnel Drive Structure was found. The actual caribou path seldom actually runs through the spot, but there is a Y-shaped hill very near it, and the hunting blinds often choose this area in order to gain a vantage point above the caribou. Also, the hunting blinds are trying to space themselves out adequately to gain points for doing that, so as a consequence, a hunting blind will often choose the exact spot where Dr. O'Shea actually found one during the Early and Mid Stanley lowstand periods. Already we can see that four of them have chosen the hill just a few generations into the Early and Mid Stanley periods.

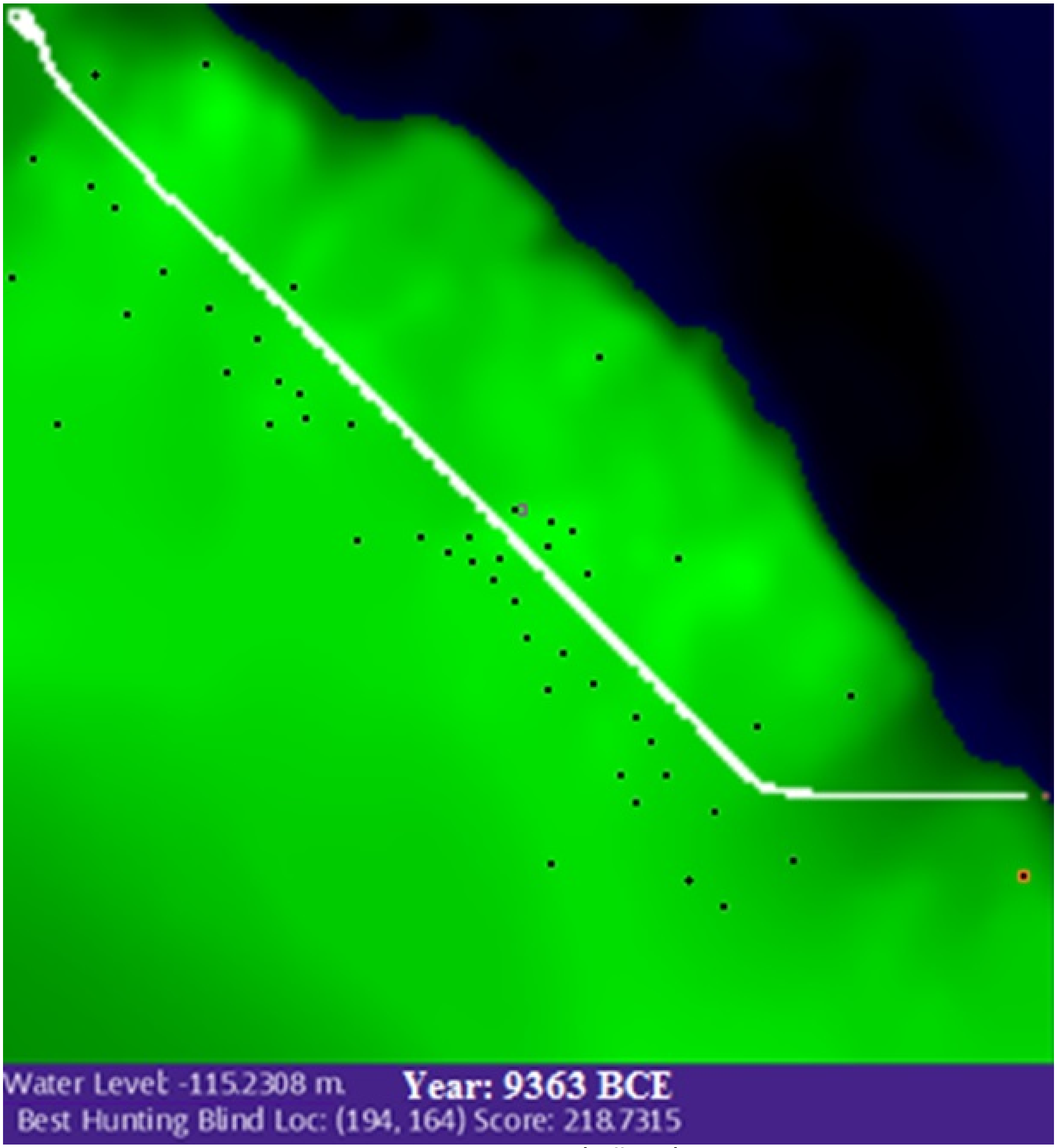


Figure 29: 9363 BCE (Full Run)

7393 BCE

We are now thoroughly out of the Early and Mid Stanley phases and well into Mid-Late Stanley. The frame for this year is typical of how the algorithm acts during Mid-Late Stanley, when lake levels are quite high. The caribou path is now significantly far to the southwest of the spot where Dr. O'Shea found the Funnel Drive Structure, so the AI hunting blinds now no longer have any incentive to go near that spot again (they would lose a huge number of points if they actually did so at this time).

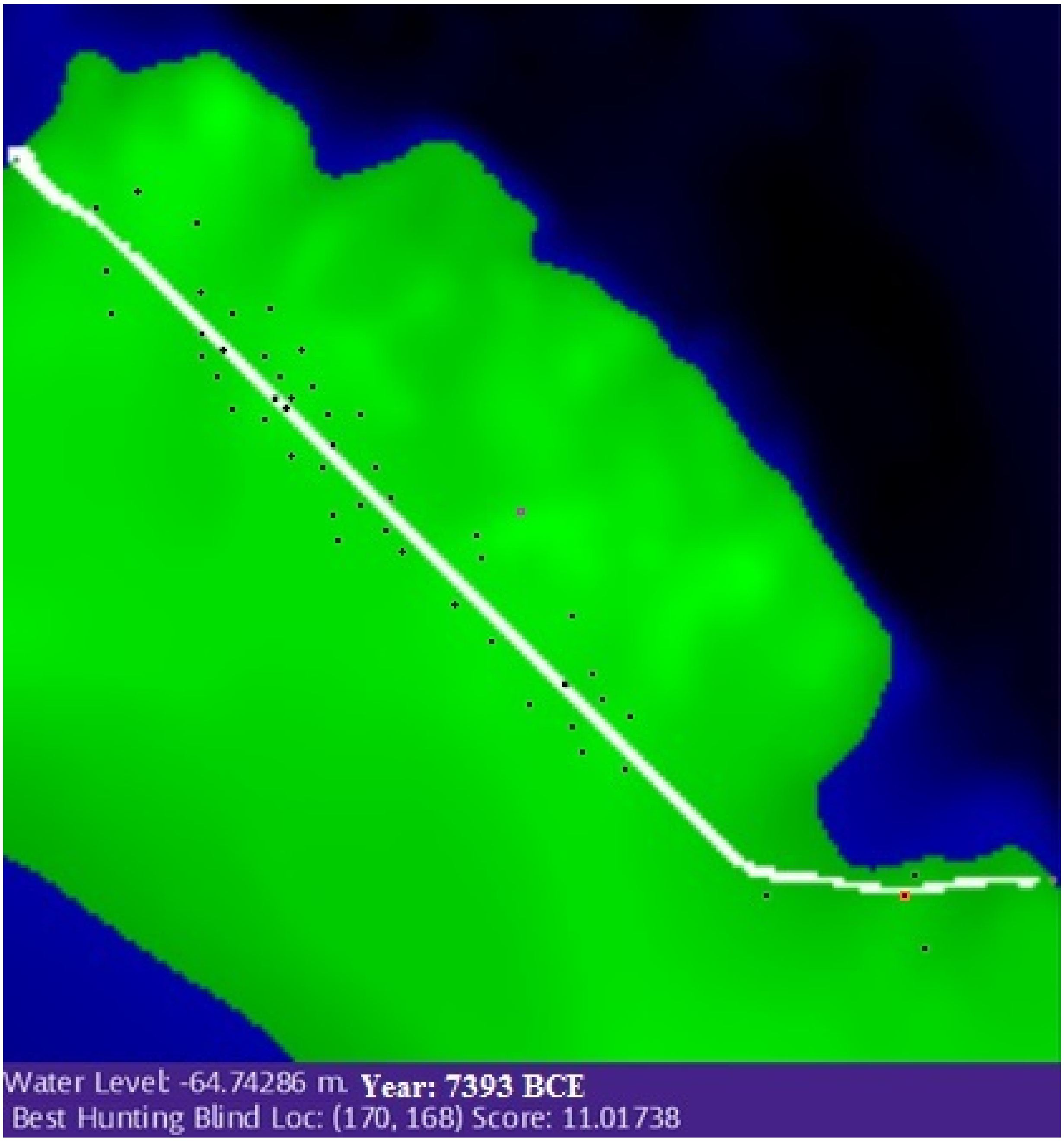


Figure 30: 7393 BCE (Full Run)

6353 BCE

We are now reaching the "final hours" of the Late Stanley phase, and therefore of the land bridge, since the flooding at the end of the Late Stanley period, unlike earlier instances of flooding, will be permanent. A good deal of the land area has been submerged already, and the land bridge as a whole is destined to enter the "island phase" in about 10 years (two generations). Once this happens, caribou will no longer be able to use the Alpena-Amberley Land Bridge as a crossing point, and it will thus cease to be an attractive caribou hunting location. Eventually, even the "island" left in the center will disappear beneath the rising lake.

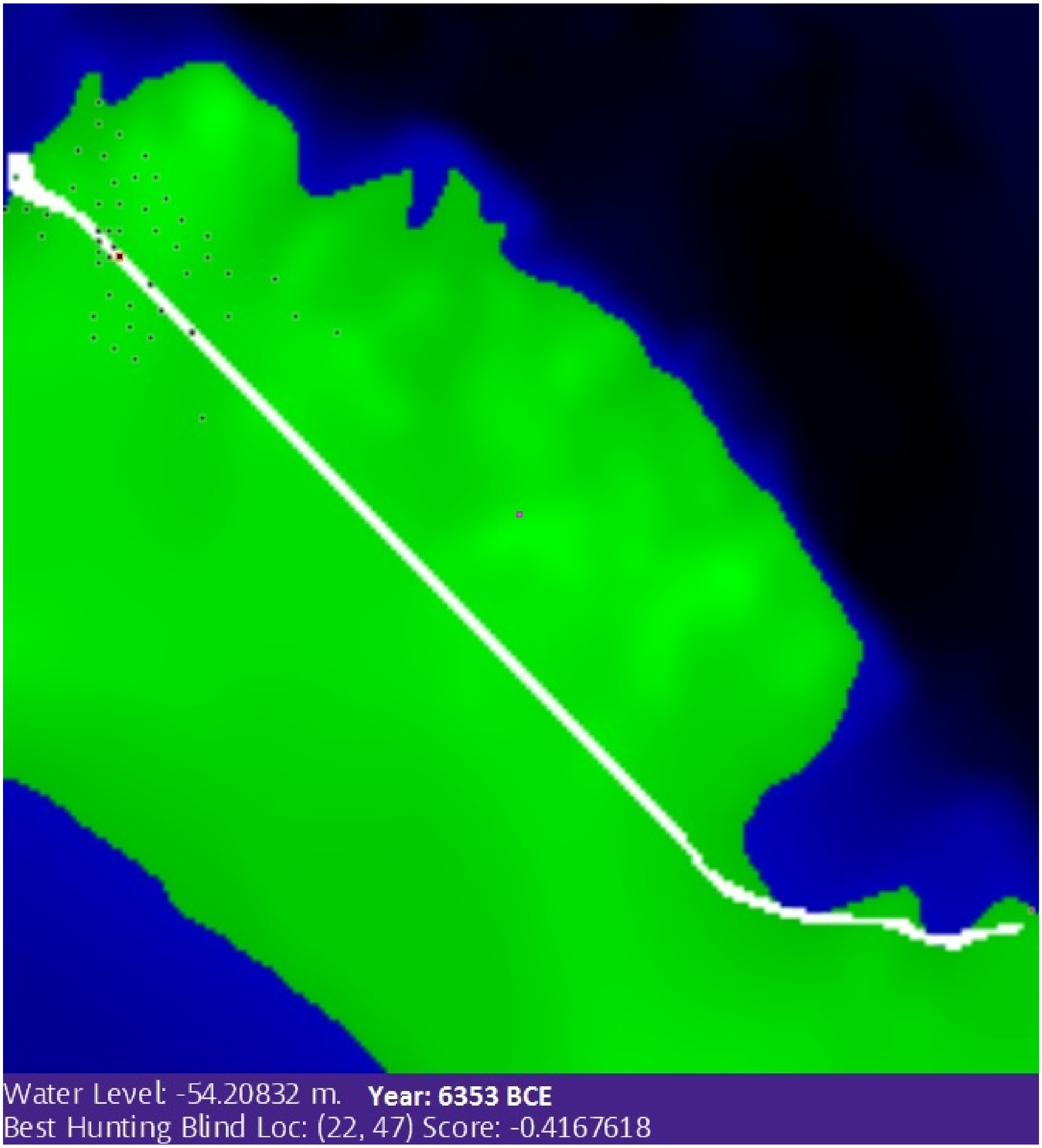


Figure 31: 6353 BCE

Learning Curve

To demonstrate our CA's learning process, we now provide a learning curve for Run 1/8.

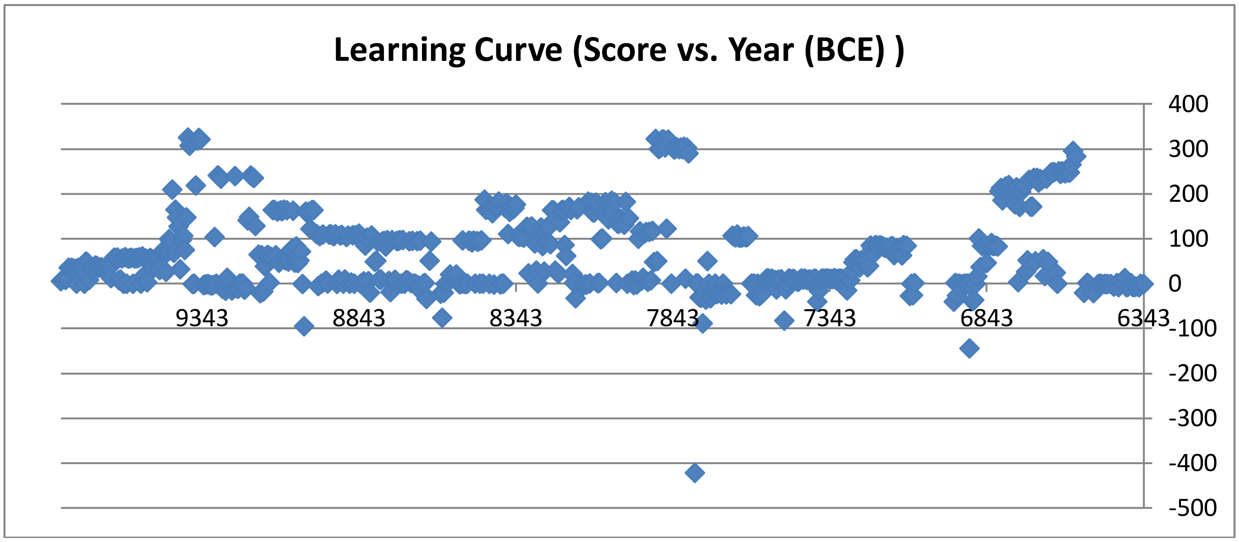


Figure 32: Learning Curve for Our CA

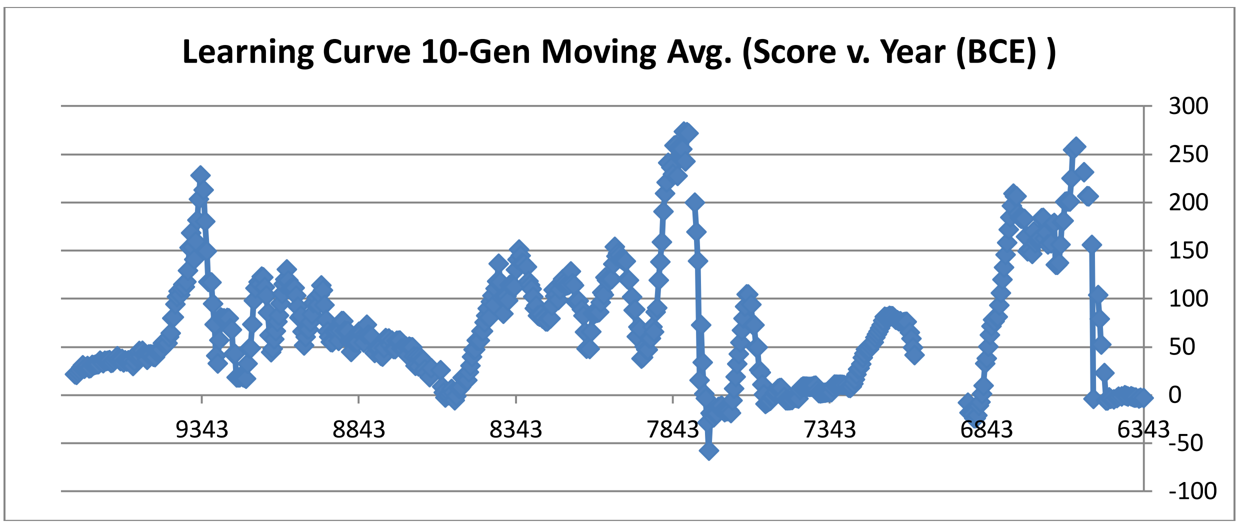


Figure 33: Learning Curve (10-Generation Moving Avg.)

The learning curve seen here is unlike most other CA learning curves, however there are important reasons for that, the most important being that our objectives are not static. Caribou paths, and most importantly water levels, are subject to sudden and unpredictable change. What had been an excellent hunting spot for a few or even many generations may not be so good, or may be completely unavailable, the next generation. In addition, the four major catastrophic water rises which befall the land bridge are major hampers on learning because they create significant periods in which the caribou do not even attempt to cross the land bridge, creating a major disruption for the hunters. Nevertheless, we can see that the algorithm is indeed learning. Notice how the 10-generation moving average reaches its overall peak during Mid Stanley, even though the water level is lower (and hence more hunting spots are available) during Early Stanley. Notice how even the Late Stanley peak for the 10-generation moving average is higher than for the Early Stanley period, even though the water level is significantly higher in Late Stanley than Early Stanley. It is only during Mid-Late Stanley, when the water level is extremely high and there are many fewer good hunting spots available than in the other periods, that the peak fails to exceed that of the Early Stanley period.

Final Results

To fully demonstrate the final results, we have created another program which outputs different kinds of heatmaps, including one that shows the average number of hunting blinds in a square over the 16 simulation runs vs. the 690 generations (3,450 years) that the land bridge is crossable. The program also places a square cyan overlay around the location where Dr. O'Shea found the Funnel Drive Structure (4964407.461N, 0381773.819E are the exact coordinates, which are rounded very slightly to 4964400N, 0381800E in our model due to the way the grid system works in ours and Palazzolo's programs and computer memory constraints). This heatmap is shown below.

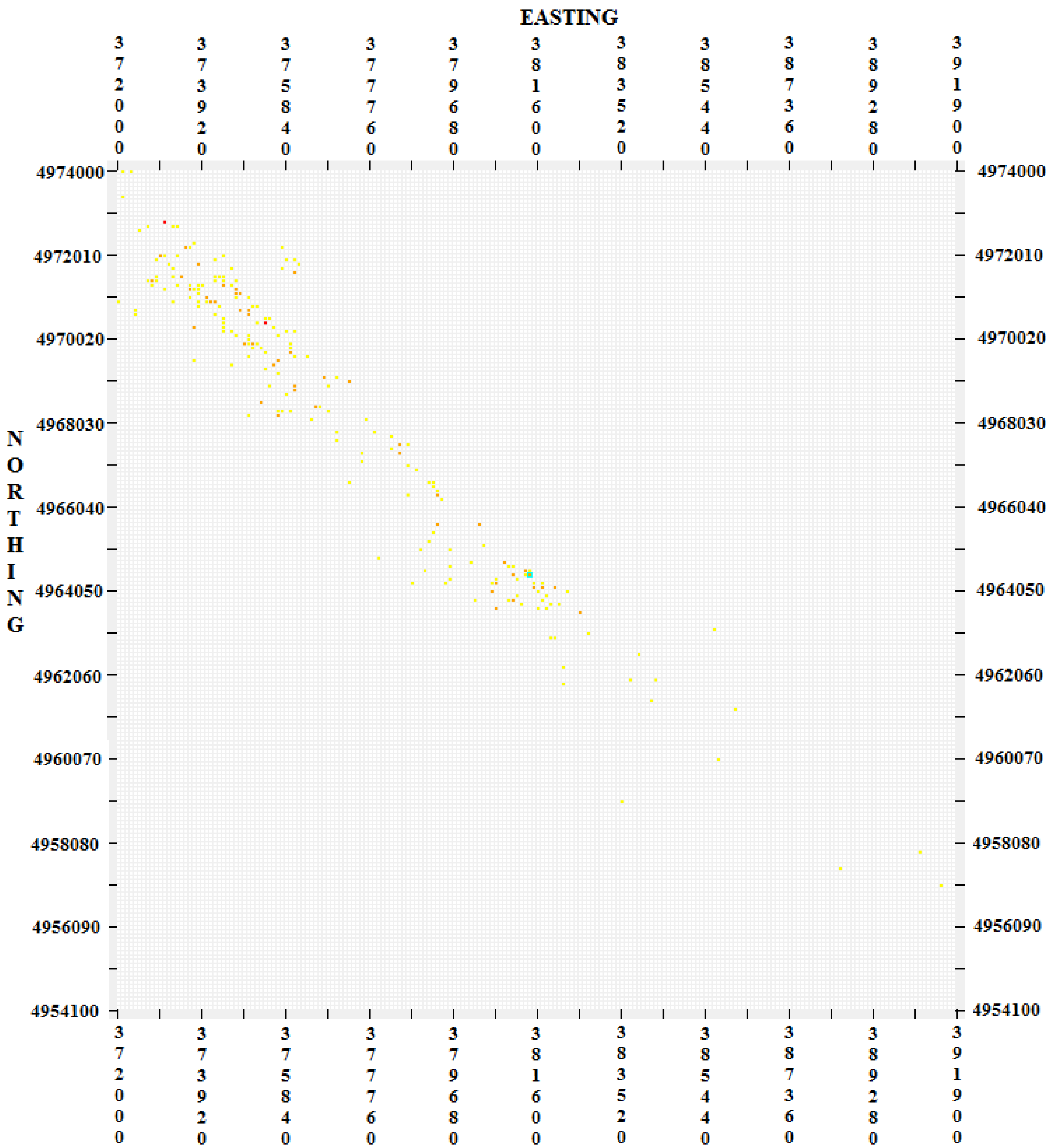


Figure 34: Full Sim Heatmap: Avg Hits for Each Square Over 16 Runs vs. 690 Gens (3,450 yrs)

Color	16-run avg # of generations that this square contained a hunting blind as a percent of 690 generations = 3,450 years (i.e., the simulation period).
Red	5%-10%
Orange	3%-5%
Yellow	2%-3%

Table 1: Map Key For Full Sim Heatmap

For the sake of convenience, we will also show close-ups of each of the four map quadrants.

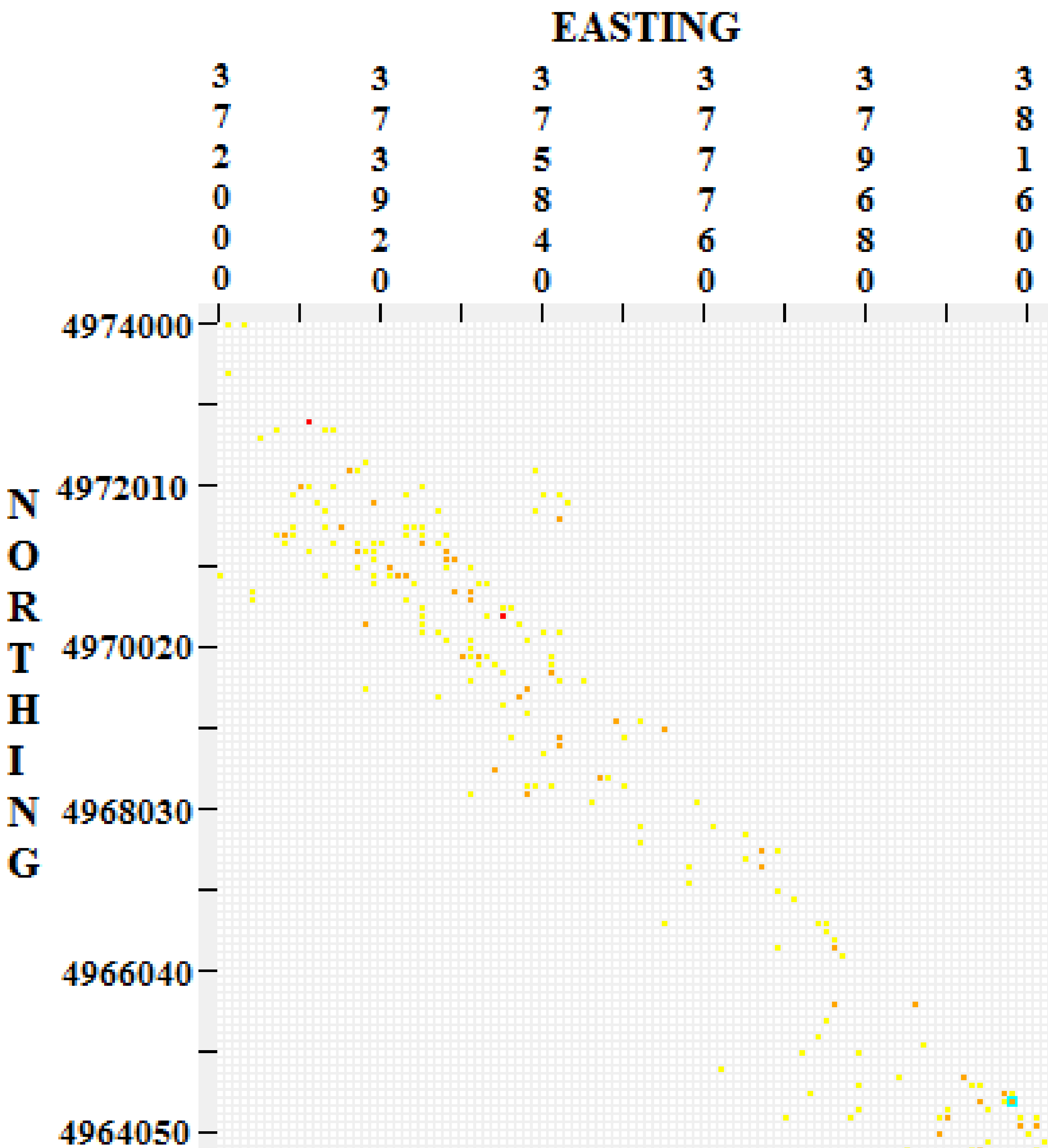


Figure 35: Heatmap Quadrant 1

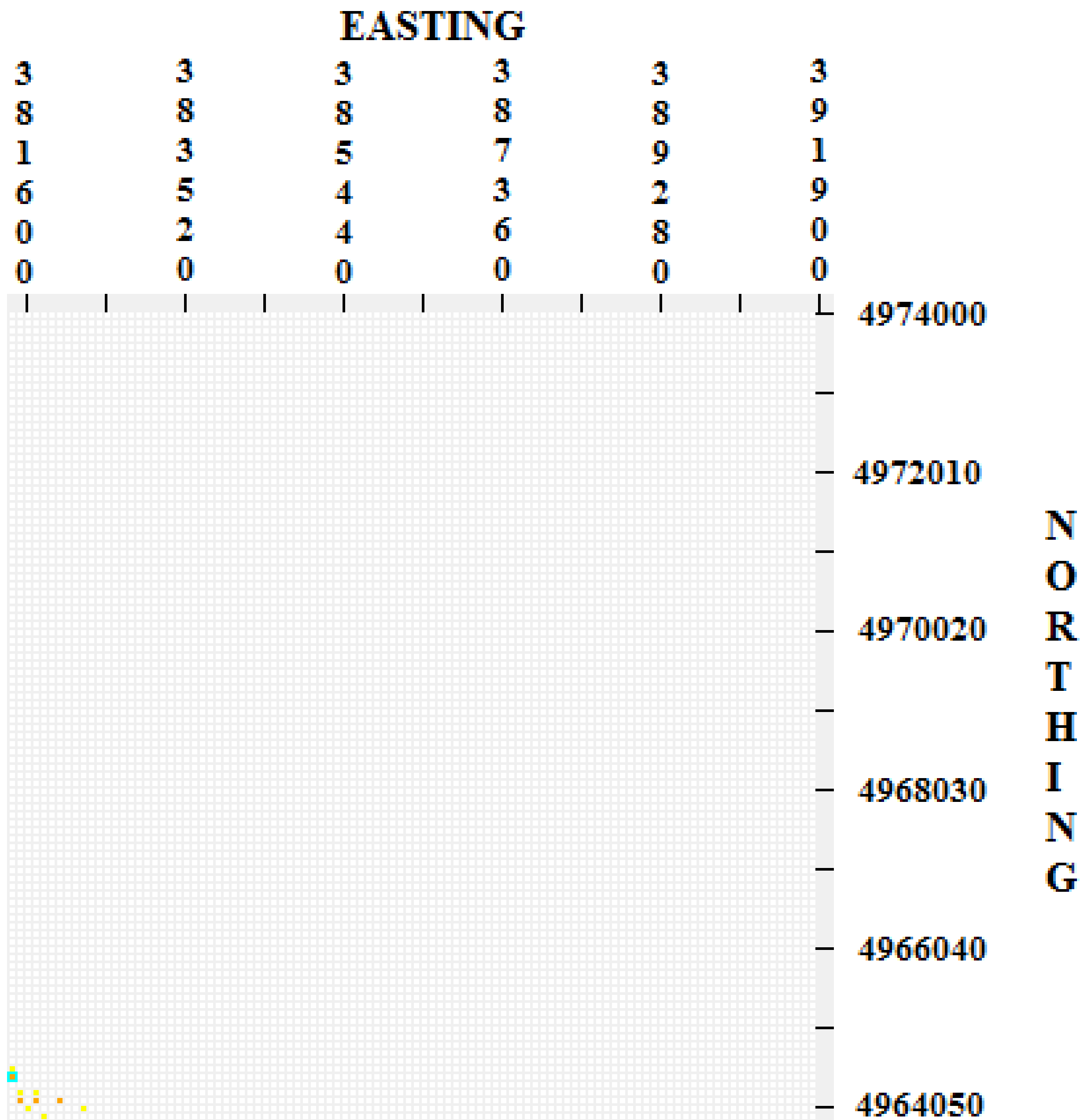


Figure 36: Heatmap Quadrant 2

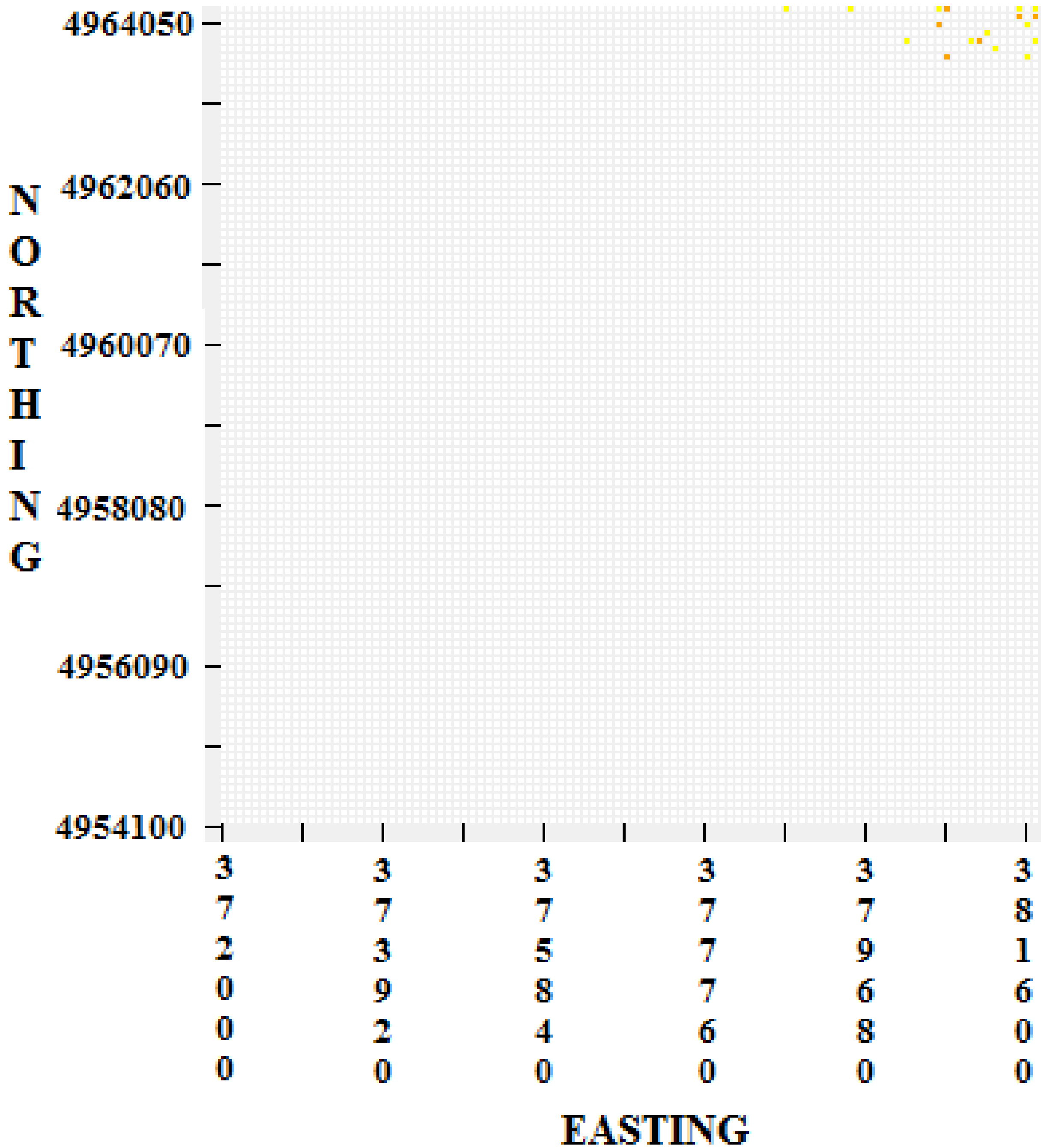


Figure 37: Heatmap Quadrant 3

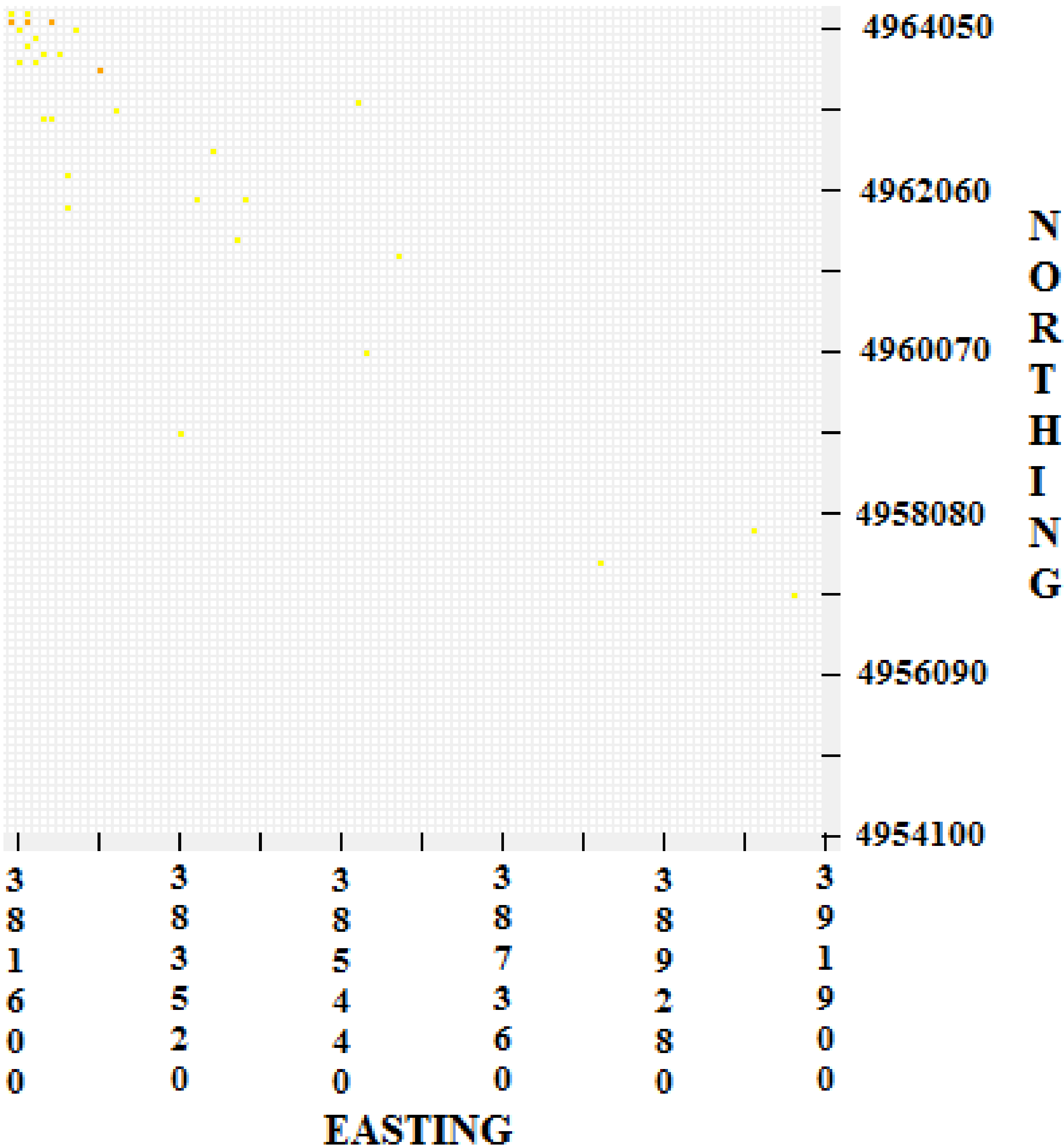


Figure 38: Heatmap Quadrant 4

As one can see from the heatmap, the cyan rectangle, denoting the location where Dr. O'Shea's expedition found the Funnel Drive Structure, is overtop a square that is colored orange, meaning that a hunting blind was in that square an average of 3-5% of the time (actual percentage: 3.804%, or 26.25 hits on average per run). Although this seems like a tiny percentage, it really is much more significant than it appears at first glance, as the vast majority of squares did not make it past the 2% threshold, the lowest threshold for receiving a color at all on the map. Looking at the squares that did receive colors, it is clear that they form a somewhat jagged "hockey stick" pattern. This makes sense, since the most crucial factor for gaining a high score in the cultural algorithm was being stationed close to the caribou. Generally, the caribou took an easy, straight diagonal path across Area 1, but often had to modify their path to avoid a low-lying valley in the southeast portion of Area 1 which would often fill up with water, blocking any path straight through, hence the caribou path is mostly a hockey stick itself. As the hunting blinds are closely tracking the caribou, they generally form this same shape on average. Where the hunting blind "hockey stick" is jagged, it is usually because of the influence of the other factors included in the cultural algorithm design: hunting blinds often choose hills and open space away from other hunting blinds even if they aren't immediately adjacent to the direct path of the caribou, as long as they are still somewhat close, to take advantage of the point rewards provided for having a vantage point above the caribou and not being too close to another hunting blind. Therefore, especially where there are hills, one can see deviations from a strict hockey stick shape.

Model Validation and Statistical Analysis

Unfortunately, given that the Alpena-Amberley Land Bridge research project spearheaded by Dr. O'Shea is still in a very early phase, and given that there are no other validated models of this kind to compare this one against, there is very little actual validation of the model that can be done, other than to confirm that its behavior is consistent with its high level design (a task that the previous portion of this chapter has basically accomplished).

Although a full and formal statistical validation of this model's effectiveness in predicting locations of actual hunting blinds would be ideal, the fact that very few hunting blinds have been actually found to date unfortunately makes it impossible at this early stage to perform such a validation. We are, however, determined to get a small start. Given that we have the coordinates of the Funnel Drive Structure (which contains 2 to 3 hunting blinds) from Thomas Palazzolo, we can design a statistical test where our model is matched against another in the task of predicting the existence of a hunting blind in the Funnel Drive Structure's location. If a formal statistical test confirms that our model is more likely to predict a hunting blind in the Funnel Drive Structure's actual location, it will be an encouraging result despite the fact that it falls short of a truly full validation. Once again, no other such model exists at this point, however we can easily create one that simply places hunting blinds in random land squares every generation.

Formal statements of the models and statistical hypotheses are as follows:

M0: "Each generation, hunting blinds are placed in random land locations."

M1: "Each generation, hunting blinds are placed according to the cultural algorithm described in this paper".

H0: "M1 predicts the existence of a hunting blind in the location that Dr. O'Shea actually found the Funnel Drive Structure blinds no more often than M0".

H1: "M1 predicts the existence of a hunting blind in the location that Dr. O'Shea actually found the Funnel Drive Structure blinds more often than M0".

Heatmap for M0 ("Random" Model)

We now instruct our system to predict hunting blinds in random land locations and produce 16 runs using that setup in order to create our dataset for M0 (our "random" model). We then provide a heatmap for M0 similar to the one we provided for M1 in Figure 34.

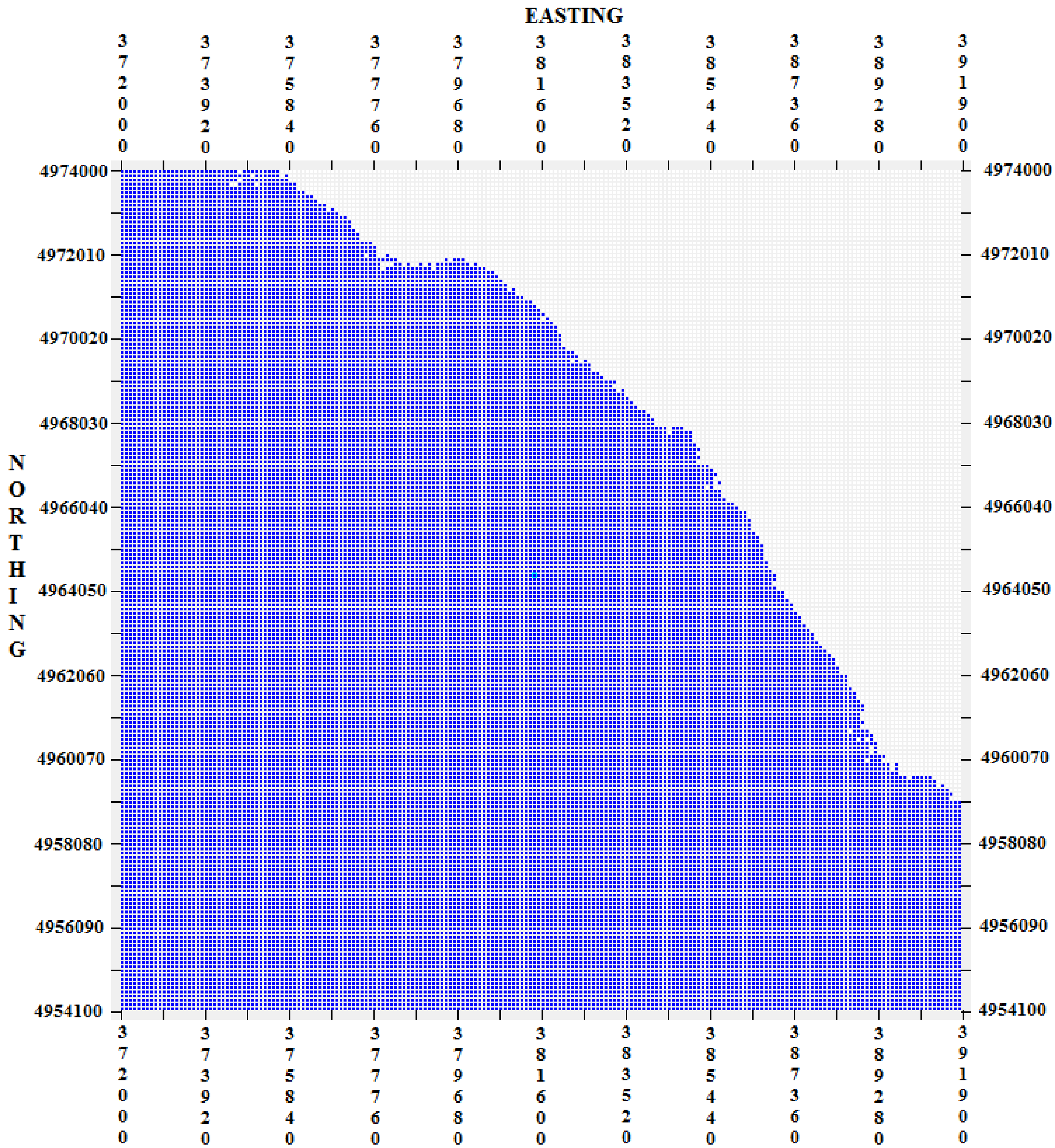


Figure 39: Heatmap for "Random" Model

The small cyan square outline marks where Dr. O'Shea found the Funnel Drive Structure.

The map key for M0's heatmap is as follows:

Color	**8-run avg # of generations that this square contained a hunting blind as a percent of 690 generations = 3,450 years (the period that the land bridge was crossable).**
Blue	Less than 2%, but greater than 0%

Table 2: Map Key for Model M0's Heatmap

For M0's heatmap, we had to provide a new color, blue, since none of the squares met even the lowest threshold for a color (2%) according to the key for our heatmap for M1. As one can plainly see, almost every square has *some* hits in it, albeit an extremely low percentage of hits. This is exactly what we would expect from a model which simply predicts hits randomly.

Testing Our Statistical Hypotheses

To test our alternative hypothesis (H1) against our null hypothesis (H0), we will use the Mann-Whitney rank-sum test. This test is very powerful because it can handle low sample sizes and makes no assumptions about the shape of the sample distribution. Essentially, the Mann-Whitney test gives each experimental unit in each of the two categories a rank based on its value and uses the sum of the ranks to determine whether one category's values are statistically greater than the other's. In our particular case, the two categories are the two models, and an experimental unit is defined as the number of generations that a hunting blind was predicted during a given run by a given model within the location that Dr. O'Shea actually found the Funnel Drive Structure. Our test will be a one-tailed test, given that all we are interested in knowing is whether M1 is *better* at predicting the existence of the Funnel Drive Structure than M0. Lastly, we wish to confirm this at an α-level of 0.05. In other words, we want to be 95% confident that M1 predicts the existence of a hunting blind in the spot where Dr. O'Shea actually found the Funnel Drive Structure more often than M0.

We proceed by first listing out the categories and values of the experimental units, and then determining the ranks of these values. In the Mann-Whitney ranking procedure, higher values get higher-numbered ranks, whereas lower values get lower-numbered ranks. In the case of a tie, all tied units receive a rank which is the average of the ranks that they would otherwise have received.

Run #	M0 unit value	M1 unit value	M0 unit value rank	M1 unit value rank
1	1	0	12.5	4.5
2	0	0	4.5	4.5
3	2	1	19	12.5
4	0	2	4.5	19
5	2	52	19	30
6	1	101	12.5	32
7	3	37	22.5	29
8	0	17	4.5	26
9	1	26	12.5	27
10	2	0	19	4.5
11	1	0	12.5	4.5
12	2	1	19	12.5
13	3	73	22.5	31
14	1	4	12.5	24
15	1	6	12.5	25
16	0	29	4.5	28
Sum	20	349	214	314

Table 3: Mann-Whitney Test Values and Ranks

We then find the U-value, which is computed through the formula

$$U = n_0 n_1 + \ n_1 \left(\frac{n_1 + \ 1}{2}\right) - \sum R_{n_1}$$

Equation 6: Mann-Whitney Formula for Calculating U-Value

In the above formula, n_0 and n_1 are the number of experimental units in the M0 and M1 categories, respectively, and $\sum R_{n_1}$ is the sum total of the ranks of all the units in the M1 category.

Plugging the proper values into the formula, we have

$$U = 16 \cdot 16 + \ 16\left(\frac{16 + 1}{2}\right) - \ 314 = 78$$

Equation 7: Mann-Whitney U-Value Calculation

Given that ours is a one-tailed test and we wish to test our hypotheses at an α-value of 0.05, the critical U value is 83 in our particular case [16]. In order to reject H0 and accept H1, our U value needs to be less than the critical U value. Since 78 < 83, we are indeed able to reject H0 and conclude that the model described in this paper predicts the location of a hunting blind in the location where Dr. O'Shea actually found the Funnel Drive Structure blinds significantly more often than a model which predicts hunting blinds in random squares for each generation.

CHAPTER 7: CONCLUSION

As far as we are aware, our cultural algorithm system is the first in the world to provide a solution to a problem where the objective is non-static and changes unpredictably ("How do hunters hunt caribou when their paths change unpredictably from year to year due to changing environmental conditions, and when the feasible solution space changes unpredictably from year to year due to changing water levels?") We have successfully produced a platform to simulate such unpredictable environmental change, and our algorithm has produced results that are reasonable and in line with what we would expect from common sense.

It should be noted once again that our core method is by no means limited to just prehistoric hunting blinds. Given a set of rules about where any type of artifact can generally be found with respect to various conditions and features within its environment, our cultural algorithm system can incorporate those rules, our time engine can simulate the changing environment during the relevant period, and heatmaps of object locations over the time period according to the simulation can be produced, just as was the case in the hunting blind experiment featured in this paper. As also mentioned before, even these existing results of this experiment regarding hunting blind locations may be useful in finding other kinds of artifacts (e.g., inukshuks and drive lanes), that would logically seem to be located near hunting blinds.

Once our colleagues are finished producing a truly full set of Land Bridge maps, the next step in our research will be to run our system over the entire Alpena-Amberley Land Bridge in order to produce a probability map encompassing the entire land bridge that archaeologists can use to decide which areas they should spend their time in searching for hunting blinds and other related artifacts, and which to ignore.

APPENDIX: 40 MOST OFTEN PREDICTED HUNTING BLIND SITES

Below is a table followed by a map of the 40 sites that were predicted most often on average by our model over the 16 simulation runs of the full experiment. (Note that X and Y map coordinates are from 0 to 199.)

16-run avg # of generations that this square contained a hunting blind	**16-run avg # of generations that this square contained a hunting blind as a percent of 690 generations = 3,450 years (i.e., the simulation period).**	X	Y	Easting	Northing
36.25	0.052536	11	12	373100	4972800
36.25	0.052536	35	36	375500	4970400
32.8125	0.047554	25	27	374500	4971300
32.4375	0.047011	32	41	375200	4969900
31.5	0.045652	37	46	375700	4969400
31.4375	0.045562	23	31	374300	4970900
31.375	0.045471	29	33	374900	4970700
31.0625	0.045018	30	41	375000	4969900
30.875	0.044746	41	43	376100	4969700
29.6875	0.043025	29	29	374900	4971100
29.5625	0.042844	10	20	373000	4972000
29.5625	0.042844	49	49	376900	4969100
29.375	0.042572	17	28	373700	4971200
29.0625	0.04212	16	18	373600	4972200
28.875	0.041848	42	24	376200	4971600
27.9375	0.040489	110	105	383000	4963500
27.5625	0.039946	28	29	374800	4971100
27.4375	0.039764	8	26	372800	4971400
26.9375	0.03904	55	50	377500	4969000
25.375	0.036775	18	37	373800	4970300
24.875	0.036051	34	55	375400	4968500
24.875	0.036051	76	84	379600	4965600
24.6875	0.035779	94	96	381400	4964400
24.625	0.035688	15	25	373500	4971500
23.8125	0.034511	67	65	378700	4967500
23.625	0.034239	89	100	380900	4964000
23.5	0.034058	38	45	375800	4969500
23.4375	0.033967	86	84	380600	4965600
23.3125	0.033786	19	22	373900	4971800
23.125	0.033514	92	93	381200	4964700

22.875	0.033152	28	28	374800	4971200
22.875	0.033152	104	99	382400	4964100
22.375	0.032428	31	33	375100	4970700
22.1875	0.032156	42	52	376200	4968800
22.0625	0.031975	90	98	381000	4964200
21.9375	0.031793	97	95	381700	4964500
21.875	0.031703	67	67	378700	4967300
21.8125	0.031612	47	56	376700	4968400
21.8125	0.031612	94	102	381400	4963800
21.8125	0.031612	98	96	381800	4964400

Table 4: Most Highly Predicted Hunting Blind Sites

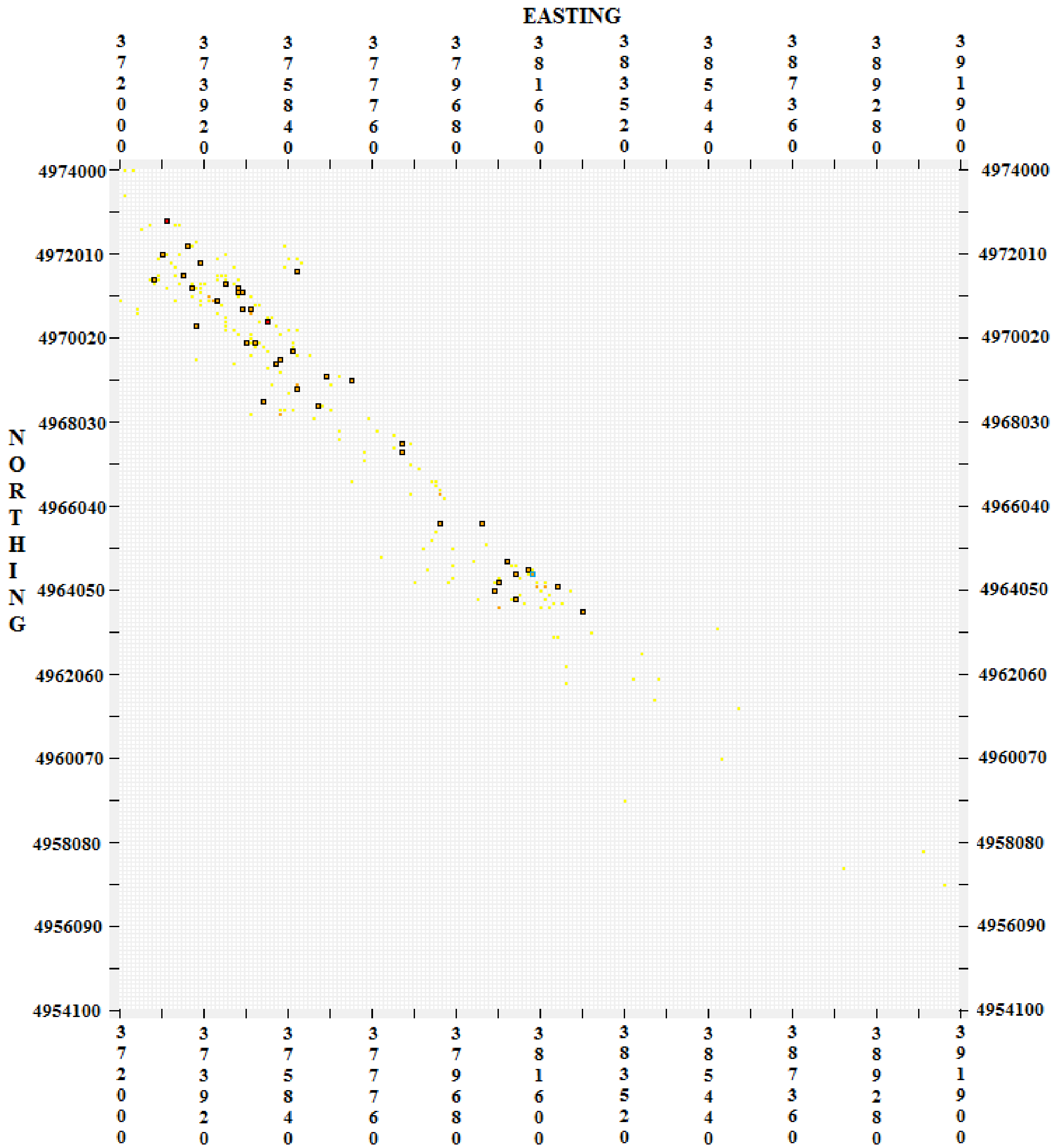


Figure 40: 40 Highest Scoring Locations Map

Color	16-run avg # of generations that square a hunting blind as a % of 690 generations = 3450 years (i.e., the simulation period).
Brown	10% or over
Red	5%-10%
Orange	3%-5%
Yellow	2%-3%

Table 5: Key for 40 Highest Scoring Locations Map

On the map, the 40 best locations have black square outlines, except for the location in which the Funnel Drive Structure was actually found by Dr. O'Shea's team. This made the list at 40th out of 40, however its outline is light blue (for easy identification). For convenience, we also provide a close-up of the map portion containing the 20 best-scoring locations.

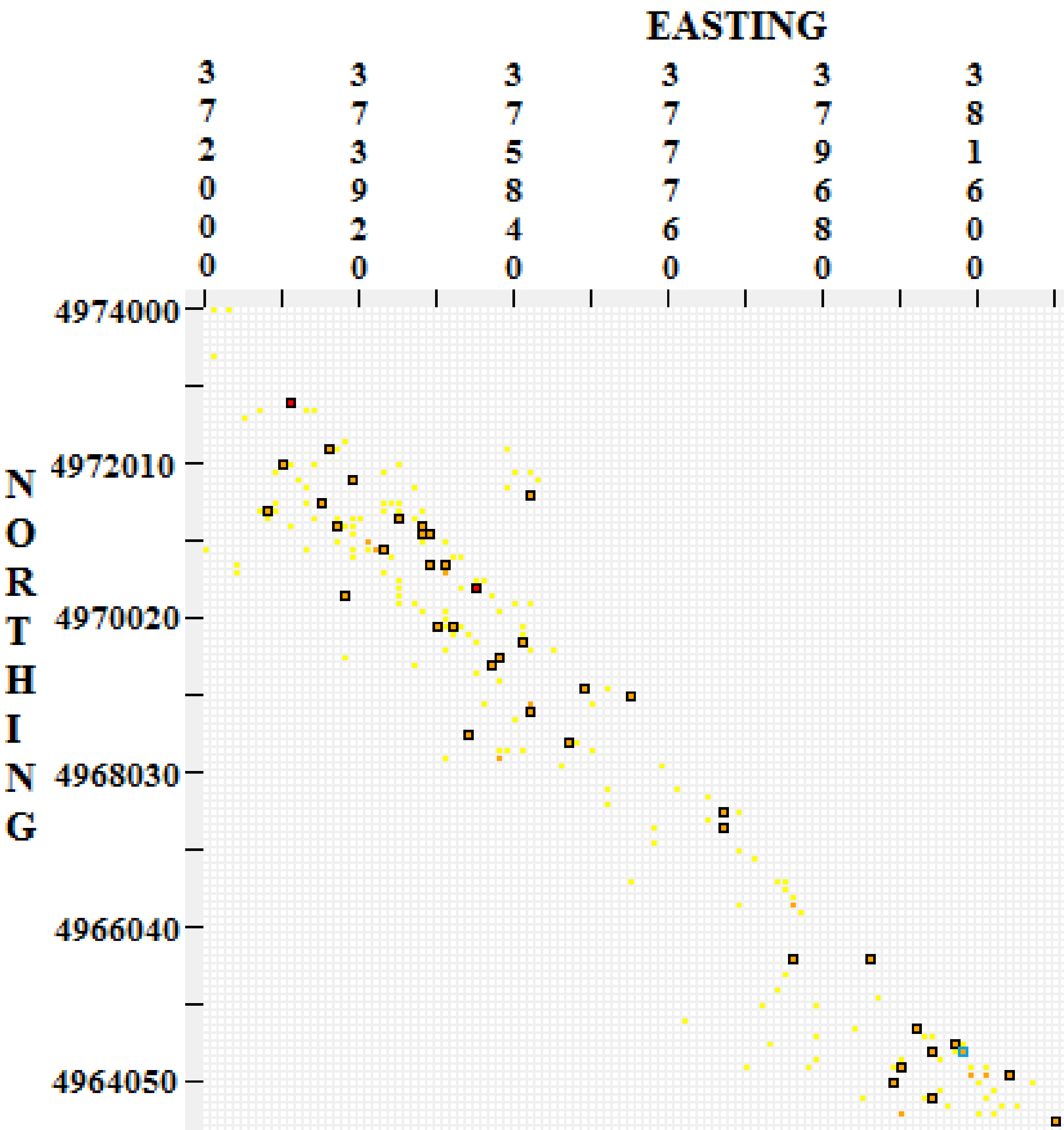


Figure 41: 40 Highest Scoring Locations Map Closeup

BIBLIOGRAPHY

[1] C. F. M. Lewis, C. W. Heil, J. B. Hubeny and J. W. M. T. C. a. R. D. K. King, "The Stanley Unconformity in Lake Huron Basin: Evidence for a Climate-Driven Closed Lowstand About 7900 14CBP, With Similar Implications for the Chippewa Lowstand in Lake Michigan Basin," *Journal of Paleolimnology,* vol. 37, no. 3, pp. 435-452, 2007.

[2] A. S. a. P. V. K. Dyke, "Late Wisconsinan and Holocene History of the Laurentide Ice Sheet," *Géographie Physique et Quatrenaire,* vol. 41, no. 2, pp. 237-263, 1987.

[3] C. F. M. Lewis and T. W. Anderson, "Oscillations of Levels and Cool Phases of Laurentian Great Lakes Caused by Inflows from Glacial Lakes Agassiz and Barlow-Ojibway," *Journal of Paleolimnology,* vol. 2, pp. 99-146, 1989.

[4] J. M. O'Shea and G. A. Meadows, "Evidence for Early Hunters Beneath the Great Lakes," *National Acaemy of Sciences Proceedings,* vol. 106, pp. 10120-10123, 2009.

[5] K. Vitale, "Learning Group Behavior in Games Using Cultural Algorithms and the Land Bridge Simulation Example," ETD Collection for Wayne State University, Detroit, 2009.

[6] J. Fogarty, "Serious Game Modeling of Caribou Behavior Across Lake Huron Using Cultural Algorithms and Influence Maps," Wayne State University, Detroit, 2011.

[7] E. W. Dijkstra, "A Note on Two Problems in Connexion with Graphs," *Numerische Mathematik,* no. 1, pp. 269-271, 1959.

[8] J. Jin, "Path Planning in Reality Games Using Cultural Algorithm: The Land Bridge Example," 2011.

[9] S. Saleem and R. Reynolds, "Cultural Algorithms in Dynamic Environments," in *IEEE Congress on Evolutionary Computation*, San Diego, 2000.

[10] National Oceanic and Atmospheric Administration, "Great Lakes Bathymetry," [Online]. Available: http://www.ngdc.noaa.gov/mgg/greatlakes/greatlakes.html. [Accessed 2012].

[11] C. F. M. Lewis, S. M. Blasco and P. L. Gareau, "Glacial Isostatic Adjustment of the Laurentian Great Lakes Basin: Using the Empirical Record of Strandline Deformation for Reconstruction of Early Holocene Paleo-Lakes and Discovery of a Hydrologically Closed Phase," *Géographie Physique et Quaternaire,* vol. 59, no. 2, pp. 187-210, 2005.

[12] R. G. Reynolds, "An Introduction to Cultural Algorithms," in *Proceedings of the 3rd Annual Conference on Evolutionary Programming*, 1994.

[13] C. Best, C. X, R. Reynolds and D. Liu, "Multi-Objective Cultural Algorithms," in *IEEE Congress on Evolutionary Computation*, 2010.

[14] C. Best, *Multi-Objective Cultural Algorithms,* Detroit: Wayne State University, 2009.

[15] X. Jin and R. Reynolds, "Using Knowledge-Based Evolutionary Computation to Solve Nonlinear Constraint Optimization Problems: a Cultural Algorithm Approach," in *Proceedings of the 1999 Congress on Evolutionary Computation*, 1999.

[16] R. C. Milton, "An Extended Table of Critical Values for the Mann-Whitney (Wilcoxon) Two-Sample Statistic," *Journal of the American Statistical Association,* vol. 59, no. 307, pp. 925-934, 1964.

ABSTRACT

ANALYZING ENVIRONMENTAL CHANGE AND PREHISTORIC HUNTER BEHAVIOR THROUGH A 3D TIME-LAPSED MODEL WITH LEVEL AUTO-GENERATION AND CULTURAL ALGORITHMS

by

SAMUEL DUSTIN STANLEY

August 2013

Advisor: Dr. Robert Reynolds

Major: Computer Science

Degree: Master of Science

This paper describes a system containing two portions whose purpose it is to help further the Alpena-Amberley Land Bridge research project and similar archaeological research. The first portion is a "time engine" which one can utilize to navigate through time in order to see how environmental conditions evolved as time passed, or to run experiments during a desired time period. The second portion is a hunting blind cultural algorithm, which is built on top of the time engine as well as Palazzolo's program. In this portion, the AI hunting blinds react to the goals that they are trying to achieve, and the goals themselves change as the environment changes over time. When the cultural algorithm is finished, the system also produces a frequency heatmap showing how often the system predicted a hunting blind within each location throughout the entire time period. Archaeologists can use this to determine which places in the area would be most worthy of sending an expedition.

AUTOBIOGRAPHICAL STATEMENT

Samuel Dustin Stanley received a B.A. from Kenyon College in 2008 with a major in mathematics with a concentration in statistics, a major in philosophy with distinction, an interdisciplinary concentration in scientific computing, and a minor in astronomy. He was the concertmaster of the Pointes Area Youth Orchestra from 2000 to 2002, and played in the Detroit Symphony Orchestra's Civic Sinfonia from 2002 to 2004. He has run several marathons, with a best time of 4 hours 24 minutes in the 2009 Detroit Marathon.