January 2019

Capso: A Multi-Objective Cultural Algorithm System To Predict Locations Of Ancient Sites

Samuel Dustin Stanley
Wayne State University, sdstanley467@yahoo.com

CAPSO: A MULTI-OBJECTIVE CULTURAL ALGORITHM SYSTEM TO PREDICT LOCATIONS OF ANCIENT SITES

by

SAMUEL DUSTIN STANLEY

DISSERTATION

Submitted to the Graduate School

of Wayne State University,

Detroit, Michigan

in partial fulfillment of the requirements

for the degree of

DOCTOR OF PHILOSOPHY

2019

MAJOR: COMPUTER SCIENCE

Approved by:

Advisor Date

DEDICATION

I hereby dedicate this work to the memory of Gerald Keith Larsen, a former Wayne State Land Bridge Team member and one of the kindest and most intelligent individuals I have ever had the pleasure of knowing. Gerald regrettably died in early 2013, but his memory lives on, and the rest of us would not be where we are today without the early groundwork that he helped to lay.

ACKNOWLEDGEMENTS

I'd like to thank first of all the Archaeological Anthropology portion of the Land Bridge Team. These team members include Dr. John O'Shea and Dr. Ashley Lemke from University of Michigan. Secondly, I'd like to thank all other past and present members of the Computer Science portion of the Land Bridge Team here at Wayne State University. These include my advisor Dr. Robert Reynolds, as well as Jin Jin, Kevin Vitale, James Fogarty, Tiffany MacLean, David Warnke, Angela Allen, Daniel Reinheimer, Areej Salaymeh, Bailey Walker, Thomas Palazzolo, and Gerald Keith Larsen. This work would not have been possible without the efforts of all of you.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

CHAPTER 1: INTRODUCTION

Researchers have always been interested in the formation of the Great Lakes, a series of five lakes sandwiched between the United States and Canada which contain about 20% of the world's surface freshwater and are among the world's most major freshwater systems [Herdendorf, 1990]. For some time now it has been known that Lake Huron, the third-largest lake in the system, was divided into two disjointed lakes at certain times during the early Holocene by a now-underwater feature called the Alpena-Amberley Ridge [Dyke, 1987]. During those times, this ridge was a long, narrow isthmus referred to by geologists as the Alpena-Amberley Land Bridge which connected what is now Alpena, Michigan in the USA to what is now Amberley, Ontario in Canada. Recently, Dr. John O'Shea of the University of Michigan has been interested in the tantalizing possibility that caribou used this Land Bridge as a corridor for migration between northern Michigan and southern Ontario during the Paleolithic and that they were hunted by prehistoric Paleoindian tribes.

1.1 Initial Project

There are a couple of crucial factors have heightened the importance of this overall project from the very beginning. First, there is the fact that the Alpena-Amberley Ridge would have been a crossable isthmus during certain times in the early Holocene, which would have made it a geographic bottleneck for migrating caribou. This would have been noticed by human hunters, who would have decided to take advantage of this unique geography. Said hunters would logically have built various occupational structures (i.e., hunting blinds, drive lines, etc.) to facilitate their hunting activities [O'Shea, 2013].

Additionally, there is the fact that any extant Paleolithic sites that are still relatively undisturbed and intact are exceedingly rare [O'Shea, 2009]. Lake Huron's water accomplishes two protective purposes: Firstly, the water has for the most part physically blocked modern humans from destroying or building over any ancient sites that may lie underneath. Secondly, since Huron is a freshwater lake, any ancient material remains, especially those made from biodegradable materials such as wood, benefit from freshwater's preserving effects.

These reasons are why Dr. O'Shea was so adamant early on about taking advantage of the potential for finding intact sites underneath what is now Lake Huron. That is why in 2008, Dr. O'Shea applied for, and received, a grant from the National Science Foundation to pursue his research goals regarding the Alpena-Amberley Ridge region. Over the following years, Dr. O'Shea and his expeditionary team used sonar, underwater autonomous vehicles (UAV's), and human scuba divers to investigate various portions of the region. O'Shea's hypothesis has paid off as his team has found various prehistoric occupational structures such as hunting blinds, caribou drive lanes, logistical camps, and caches in this region [O'Shea, 2009, 2013]. His findings were also picked up by the popular press and named one of the top 100 scientific discoveries of the year 2009 by the popular science magazine *Discover* [Barth, 2010].

1.2 Cultural Algorithm Team Involvement in the Project

In 2009, the Cultural Algorithm team from the Artificial Intelligence Laboratory at Wayne State University under the direction of Dr. Robert G. Reynolds became interested in collaborating with Dr. O'Shea on the project. The WSU team's main contribution has been the "Land Bridge GUI", a program which assists Dr. O'Shea's underwater expedition team in a number of ways. The Land Bridge GUI system is the result of the collaborative work of multiple Wayne State University graduate students including Kevin Vitale, James Fogarty, Thomas Palazzolo, Jin Jin, Gerald Larsen, David Warnke, Areej Salaymeh, and myself under the direction of Dr. Robert G. Reynolds. Originally the system was simply designed to simulate the crossing of a herd of AI caribou over a landscape created from NOAA height-map data and to provide a realistic-looking visualization of their crossing. However as described in [Stanley, 2013], in 2011 we developed a "time engine" for the Land Bridge GUI that takes time series data on various environmental variables such as water level and temperature and provides a rich visualization of the changing environment over time as well as the ability to run experiments during specific time periods. Around the end of 2011, we also began developing a hunting blind artifact finder system that used input from the constantly changing simulated environment to produce dynamic influence maps. These influence maps retain and expand upon relevant knowledge about the environment while discarding irrelevant knowledge. These maps provided environmental knowledge to (and receive environmental knowledge from) AI “hunting blind teams” loaded with a Cultural Algorithm which scored them according to a fitness function determining the most effective hunting blind locations. The program then combined these results into heatmaps predicting where archaeological expeditions are most likely to find actual prehistoric hunting blinds. In the period that followed the publication of [Stanley, 2013], the artifact finder algorithm was overhauled twice and the program was revamped to be able to handle other artifact types. It was also reengineered to enable the fast implementation of rules and the variables that influence them as suggested by the anthropological archaeology research community. The rules and variables were derived from the works of several anthropological archaeologists including Lewis Binford [Binford 1978a, 1978b, 1980, 1982, 1991], John O'Shea [O'Shea, 2013], and Ashley Lemke [Lemke, 2016]. The main interface of our program is shown in Figure 1.

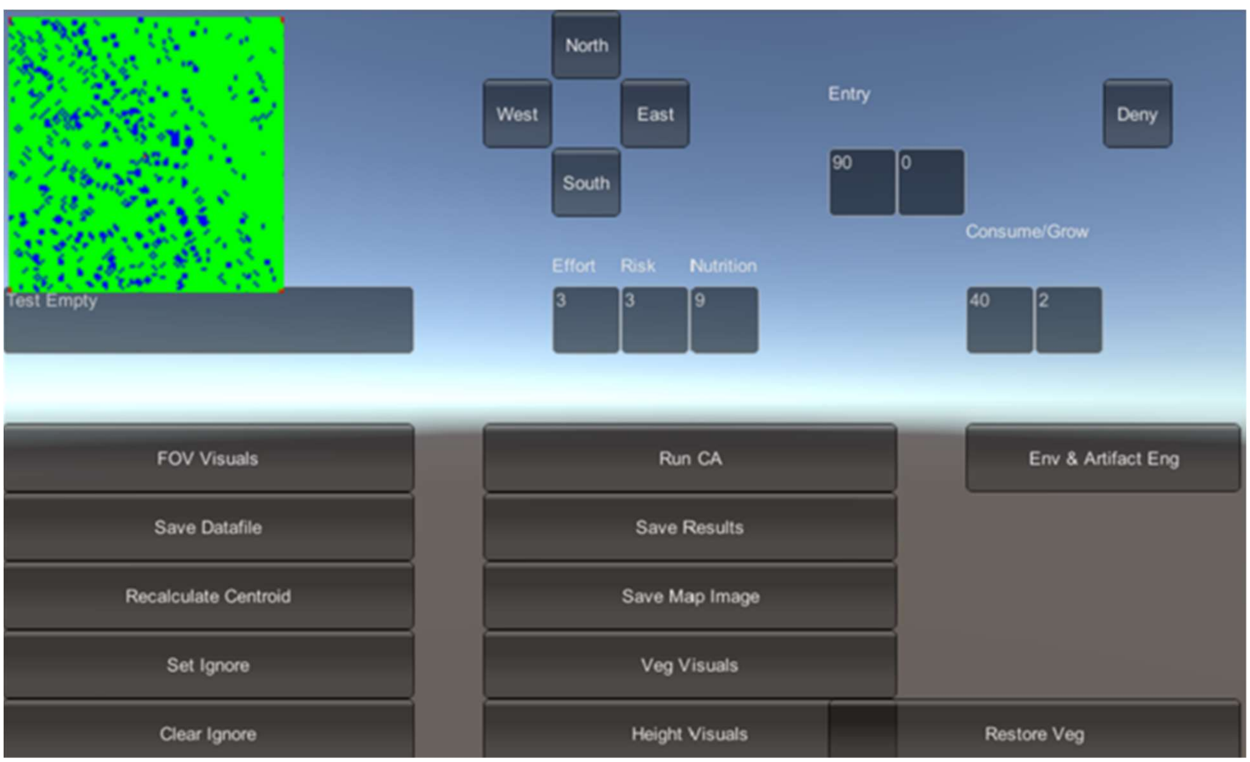


Figure 1: Land Bridge GUI Main Screen

1.3 A Cross-Disciplinary Effort

This work would not have been possible were it not for the groundwork having been laid across several widely disparate disciplines, namely geology, anthropological archaeology, information theory, and computer science.

The artifact heatmaps supplied to Dr. O'Shea are produced from artifact algorithms which require compiled environment maps, a generated set of caribou paths, and either functions that describe AI agent behavior (for the agent-based approach) or rules that directly describe AI artifact behavior (for the rule-based approach). The compiled environment maps are produced by Thomas Palazzolo's program from a combination of heightmap data from the National Oceanic and Atmospheric Administration [NOAA, 2012], prehistoric yearly water level data from Dr. Mike Lewis [Lewis, 2016], and vegetation data produced by Palazzolo's vegetation simulation algorithm.

In order to generate the caribou paths, up until 2014 we used an A* algorithm implemented by James Fogarty and Jin Jin [Fogarty, 2011] [Jin, 2011] [Stanley, 2013]. In 2014 we replaced that algorithm with a CA-equipped A* algorithm designed by Thomas Palazzolo [Stanley, 2014]. In 2017, that algorithm was replaced with an improved version also designed by Thomas Palazzolo which also included an algorithm to simulate the consumption of landscape vegetation. In mid-2018, that algorithm was replaced with a CA-equipped version of the A*mbush algorithm, also designed by Thomas Palazzolo and which also works with his new vegetation algorithm.

Prior to 2015, the process of determining the artifact predictions was governed by a CA-equipped agent-based algorithm described in [Stanley, 2013, 2014]. In 2015, that algorithm was improved by being loaded with a Lamarckian "look-ahead effect" further described in Chapter 5. In February 2017, that algorithm was replaced with an entirely different algorithm using a rule-based approach rather than an agent-based approach. All three algorithms incorporate work from Dr. O'Shea in the form of functions and rules, however as time went on, more and more effort was made to incorporate more work from different archaeologists (such as Lewis Binford and Ashley Lemke) and to streamline this process. This led to the development and completion in April 2017 of a rule engine that allows for very fast incorporation and implementation of rules, factors that go into those rules, and even whole new artifact types, coming from the work of anthropological archaeologists.

1.4 Artifact Finder Motivation

In practice, archaological expeditions face time and money constraints. According to Dr. O'Shea, it costs an average of $1,000 per day to work out on the research vessel out on Lake Huron. Thus it is not possible to do a detailed archaeological survey of every single location on the Alpena-Amberley Ridge. The original motivation of our artifact finder was to supply the locations most likely to turn up an artifact in the form of heatmaps. Dr. O'Shea can then use those heatmaps in conjunction with his own intuition in order to decide which locations to send the research vessel out to, and which to ignore. We eventually added functionality to obtain predictions for eight different artifact types discussed in the anthropological-archaeological literature and that might logically have been constructed by prehistoric hunter gatherers living on the Alpena-Amberley Land Bridge. They are the following: Residential Camps, Logistical Camps, Fishing Field Camps, Observation Stands, Large Game Hunting Structures, Small Game Trapping Structures, and Caches [Binford 1978a, 1978b, 1980, 1982, 1991] [O'Shea, 2013] [Lemke, 2016]. (Specific details for each of these artifact types will be discussed in Chapter 2.)

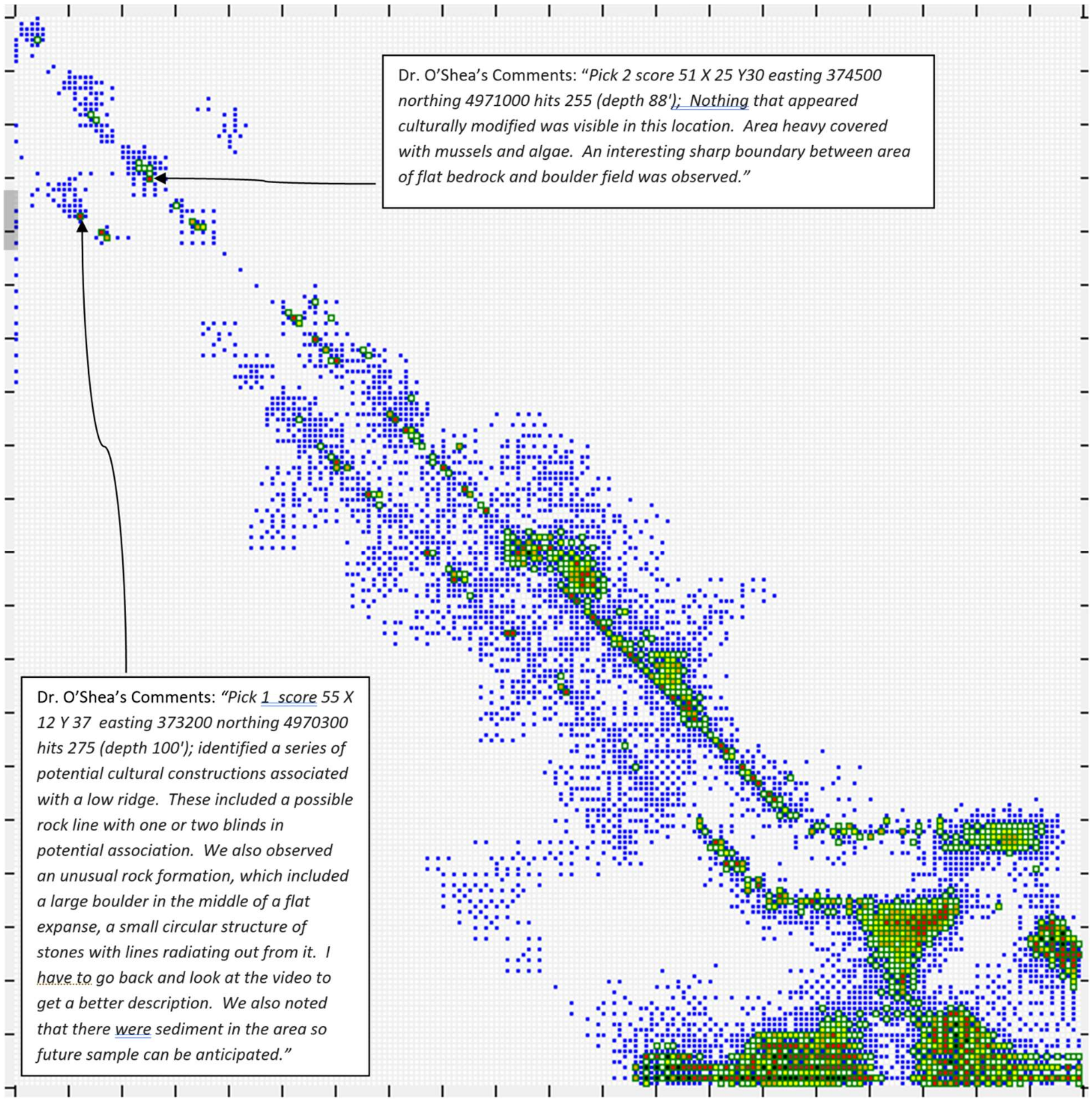


Figure 2: Example Observation Stand Heatmap (11800-8350BP Area 1)

Figure 2 is shown here as an example of a heatmap produced by our system for Area 1 of the Alpena-Amberley Ridge (whose location is denoted by the larger rectangle in Figure 2 on page 8). Area 1 is a place of great research interest to Dr. O'Shea, and during his 2016 expedition to Area 1 in October, he used the heatmap in Figure 2 to locate two occupational structures, one of which was a large game hunting structure containing a hunting blind and drive line, the second was a ring of stones with lines radiating out from it which is currently unclassified, but may have been an Observation Stand similar to the one in Figure 19, which is also mainly comprised of a ring of stones.

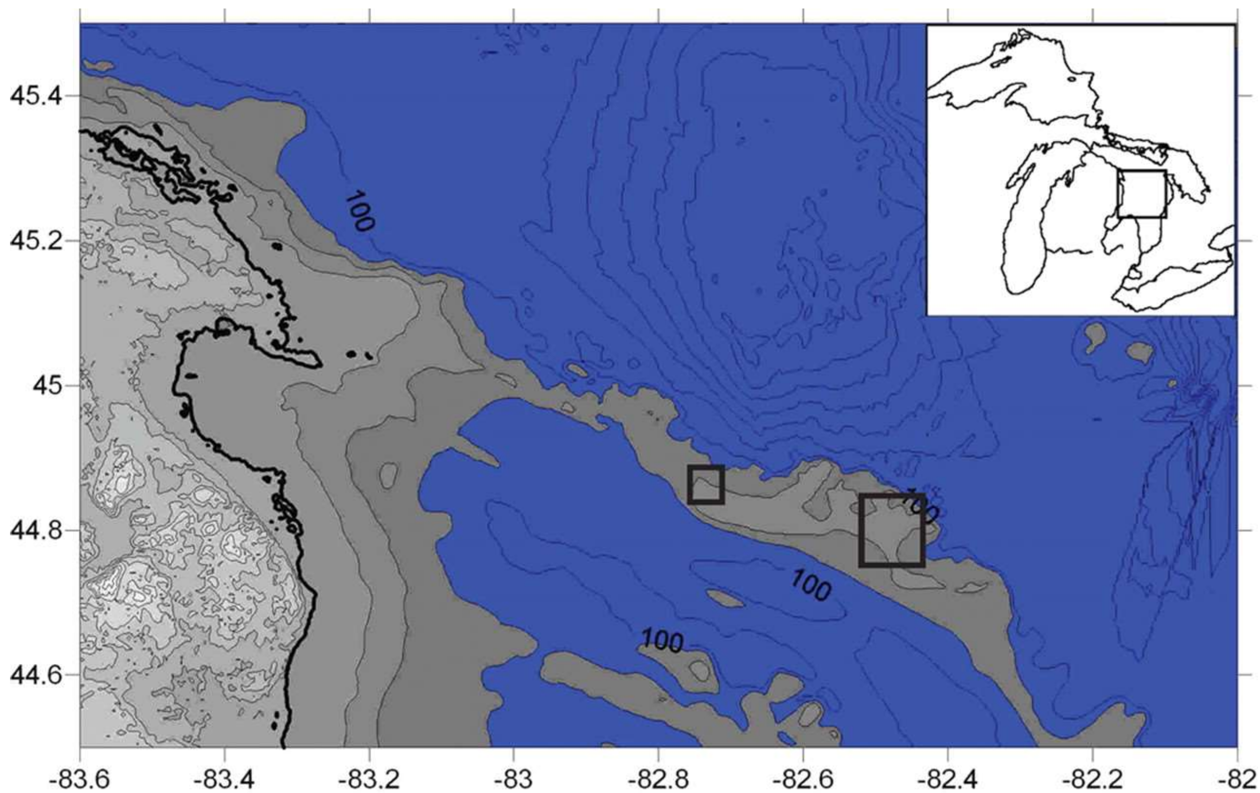


Figure 3: Alpena-Amberley Land Bridge. The larger rectangle denotes Area 1. [O'Shea, 2009]

Heatmaps such as the one in Figure 2 are designed as a tool for archaeologists such as Dr. O'Shea to use in conjunction with their own intuition in order to plan sorties out to sites most likely to bear artifacts of interest. Additionally, if a known artifact is difficult to classify, comparing artifact type heatmaps can help to classify it into one (or more) of the preexisting artifact types, or into a new artifact type whose properties are specified by the archaeologists.

1.5 Boosting the Land Bridge Project with Multi-Objective Optimization

In 2018, we realized that rather than merely generating individual heatmaps for each structure type, it would be better to predicate the construction of heatmaps directly upon the two objectives that most concern archaeological projects such as the one Dr. O'Shea is undertaking: Minimizing the number of locations that the archaeological expeditionary team has to search, and maximizing the number of culturally-modified structures found. Because these are two directly countervailing objectives (the first relates to effort, while the second relates to payout), this can be formulated in terms of a bi-objective optimization problem. Additionally, it can dovetail with the rule-based approach. Evolving parameters for the rules that predict locations within our system can be directly tied in to evolving a Pareto-optimal set of (number of locations predicted, number of structures found) ordered pairs. We can produce a Pareto-optimal set for each of the eight structure types individually and/or a full combined composite Pareto-optimal set for all eight structure types combined. From each point in each of these Pareto-optimal set, the system can produce a structure heatmap (like the one in Figure 2, for instance, and the archaeologists can choose the one that looks the most promising.

Additionally, because these sets are Pareto-optimal, there are only three ways to improve on them and the heatmaps created from them: The first way would be to obtain better paleoenvironmental data such as better caribou behavior information, better water level data, or reliable temperature data for Alpena-Amberley Ridge for this time period, the last of these which has yet to be forthcoming at all. The WSU Land Bridge Team is already using the latest data regarding caribou behavior and the latest water level data for the Huron basin [Lewis, 2016]. The second way would be to obtain better and/or more specification forms of rules, something which would require more specialized knowledge from expert archaeologists. The WSU Land Bridge Team consults regularly with the anthropological-archaeology professors Drs. O'Shea and Lemke and as far as we are aware, our specification forms of our rules are up-to-date. The third way would be to improve the training set, which happens whenever the archaeological team finds more structures on the Alpena-Amberley Ridge. The WSU Land Bridge Team currently has the most up-to-date training set, provided as of April 2018. Because none of these three aforementioned things have to do with Computer Science, at least not per se, the Computer Science side of the structure-finder project can be considered complete once the aforementioned Pareto-optimal sets have been found and documented.

1.6 Workflow Diagram

Once again, this project would not be possible without the combined efforts from a number of different people working across a number of different disciplines. We have thus produced a workflow diagram in Figure 4 to summarize their individual contributions previously discussed and to show how those contributions fit together in order to create this project.

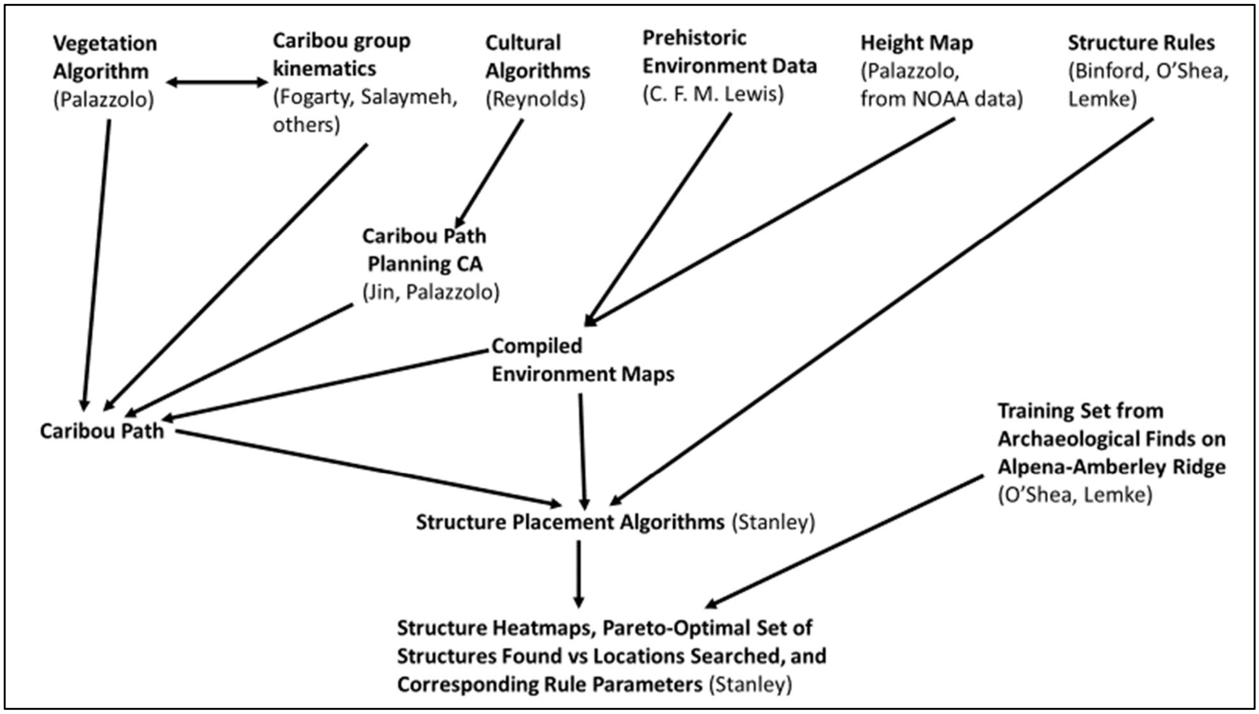


Figure 4: Workflow Diagram for the Land Bridge Project

1.7 Component Diagram

We now provide a component diagram to show how these various subsystems within the Land Bridge Project fit together and interact with one another.

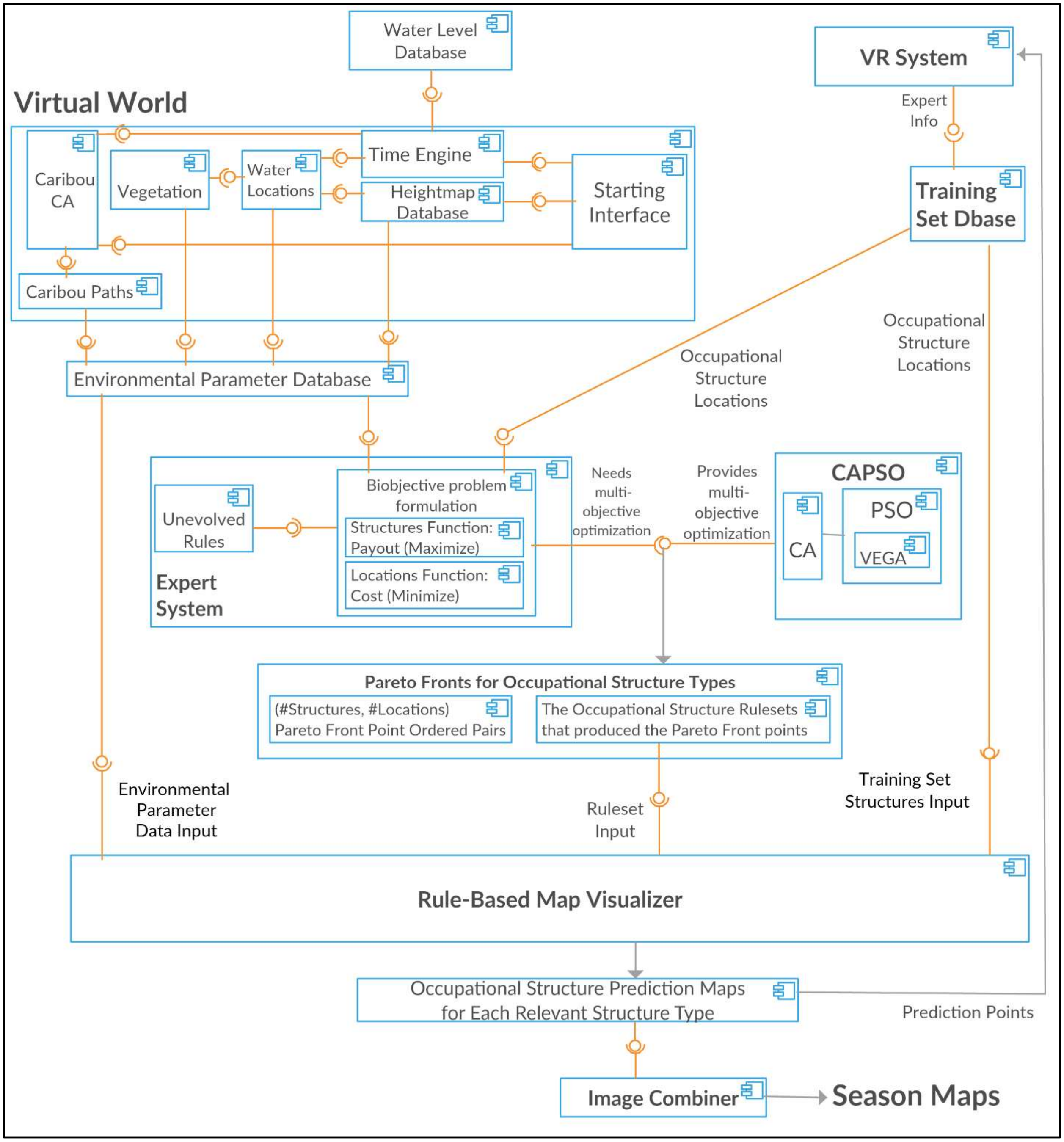


Figure 5: Overall System Component Diagram

1.8 The Accelerating Cost Hypothesis

We predict that the Pareto Front discussed in the previous section will be logarithmic in shape. In other words, once the first several structures are found, loosening the rules in order to include more and more locations in the prediction heatmaps will have diminishing returns. Another way to look at this hypothesis is in terms of accelerating cost: If predicting a certain number of structures is at the cost of flagging a certain number of locations, then predicting a slightly greater number of structures will be at the cost of flagging a much greater number of locations. We thus name our hypothesis *The Accelerating Cost Hypothesis.* Statistical validation of the Accelerating Cost Hypothesis is provided in Chapter 7, and implications are discussed in Chapter 9.

1.9 The Low Initial Cost Hypothesis

Supposing that the Accelerating Cost Hypothesis is true, the cost-to-benefit ratio will always increase at an increasing rate for each of our Pareto fronts. The question is: How big is this ratio at the bottom end of the Pareto curve? Does it start out small enough so that the lower end of the cost curve is low enough such that archaeological teams of more limited means can still afford the lower end of the cost curve? Or is even the lower end of the cost curve very expensive and thus out of the reach of teams of more modest means? For the purpose of this hypothesis, which we are terming the "Low Initial Cost Hypothesis", we are assuming the former. We will provide more discussion of the Low Initial Cost Hypothesis at the end of Chapter 8, after providing hypothetical scenarios of how different archaeological teams might use our system.

1.10 The Ruleset Size vs. Problem Complexity Hypothesis

One of the hardest things about this project is that it is a “worst of both worlds” situation that somehow manages to combine the challenges involved with both “Big Data” and paucity of data. With regard to the latter, it is very difficult to reconstruct the Early Holocene environment due to its extreme antiquity and there are considerably few categories which we can reliably model. This leads to our rulesets being necessarily small. On the other hand, we are simulating over a timespan of 3,400 years, so even a small number of data categories becomes rapidly multiplied into many millions of data entries. The question becomes, which factor will win? Will paucity of data categories ensure that this is a simple problem, or will the sheer volume of data produced because of the temporal component ensure that this is a complex problem? For the purpose of this hypothesis, which we are calling the “Ruleset Size vs. Problem Complexity Hypothesis”, we are supposing the latter. We will provide more discussion about this hypothesis at the end of Chapter 7.

1.11 Overview of this Dissertation

The rest of this dissertation is arranged as follows: Chapter 2 contains a discussion of previous work done in the study of the Alpena-Amberley Land Bridge. Chapter 3 provides discussion of the paleoarchaeological background of the Alpena-Amberley Ridge region and the prehistoric artifacts that are expected to be found there. Chapter 4 discusses how a virtual world of the prehistoric environment, including prehistoric topography, water levels, and vegetation, is modeled. Chapter 5 contains a formal specification of the various structure types and the parameters and rules that pertain to each of them, as well as a formal specification of these in terms of biobjective optimization problems. Chapter 6 contains a discussion of Cultural Algorithms and the CAPSO (Cultural Algorithm / Particle Swarm Optimizer) system that will create Pareto Fronts for each structure type out of the biobjective optimization problems specified in Chapter 5. Chapter 7 contains said Pareto Fronts along with the evolved rulesets that generated each Pareto-optimal point, along with visual maps resulting from applying these rulesets into the pertinent area of archaeological study. Chapter 7 also contains a statistical validation of the Accelerating Cost Hypothesis. Chapter 8 explores possible ways in which hypothetical archaeological teams with different research aims might each choose to composite the results in Chapter 7 for the purpose of planning expedition seasons in order to achieve their research aims. In Chapter 9, final conclusions, including the implications of the Accelerating Cost Hypothesis, are discussed.

CHAPTER 2: PREVIOUS WORK ON THE LAND BRIDGE PROJECT

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 was an uninterrupted land corridor with two lakes on either side during part of the melt phase of the Laurentide ice sheet [Dyke, 1987] [Lewis, 1989] [Lewis, 1994]. In 2008, the University of Michigan Museum of Archaeology along with the University of Michigan Department of Oceanographic and Atmospheric Engineering and Wayne State University received an NSF High Risk Grant to begin the exploration of the Alpena Amberley Land Bridge in search of Paleo-indian occupational remains. The resultant expedition to Lake Huron was carried out using side-scanning sonar, underwater autonomous vehicles (UAV's) launched from surface boats, remotely operated vehicles (ROVs), and finally human divers. During the expedition, the research team found the remains of prehistoric hunting blinds and caribou drive lanes [O'Shea, 2009]. When these results were published, there was a surge of interest within the Anthropological Archaeology community concerning the Alpena-Amberley Ridge, as this is a pristine region largely undisturbed by the activities of modern humans that also benefits from the preserving effects of freshwater on normally perishable materials such as wood that prehistoric peoples often used to fabricate structures. This surge of interest also carried over to the Artificial Intelligence community and before long there were researchers who wanted to create computer models of the behavior of the caribou and hunters that roamed the Alpena-Amberley Ridge when it was dry land in prehistoric times.

2.2 Learning Group Behavior in Games Using Cultural Algorithms and the Land Bridge Simulation Example.

The first computer models of caribou behavior on the land bridge were implemented by Kevin Vitale and Dr. Robert Reynolds in 2009, discussed in the paper "Learning Group Behavior in Games Using Cultural Algorithms and the Land Bridge Simulation Example" [Vitale, 2009]. Vitale's program used a Cultural Algorithm (CA) simply to teach caribou agents (represented as yellow triangles in Figure 6) how to successfully migrate as a herd across the land bridge.

Figure 6: Vitale's Land Bridge model with caribou forming a herd to migrate across the Land Bridge. [Vitale, 2009]

Vitale's CA 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 kinematic wander behavior is determined by three values: The *wander target position*, the *wander circle radius*, and the *projection distance*. Vitale's pseudo-code for his wander behavior mechanism is given below: [Vitale, 2009]

```
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
}
```

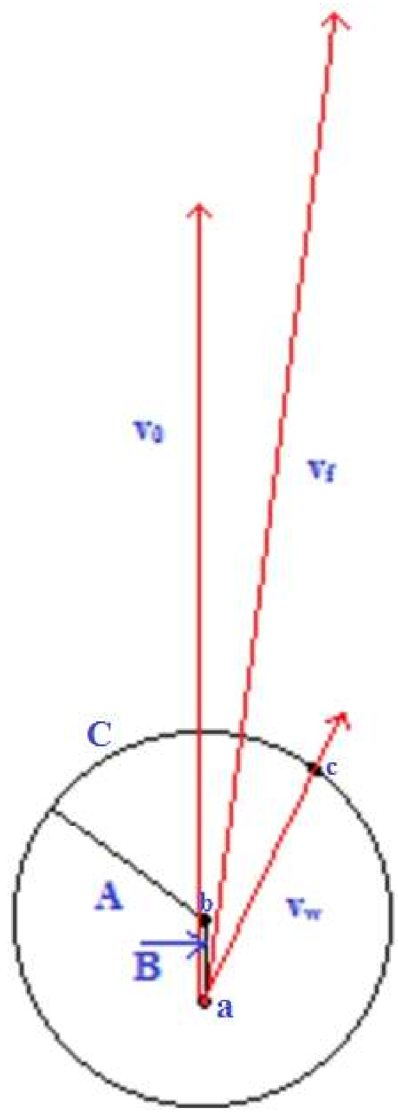


Figure 7: Schemata of Vitale's Wander Mechanism

The diagram given in Figure 7 above details how Vitale's wander behavior kinematic 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 parameters, the wander circle radius and projection distance, are hardcoded into the program. v_0, the initial velocity, is also hardcoded into the program. It combines with v_w, the velocity produced by the wander mechanism described above, to produce v_f, which is the final velocity for an individual caribou until v_w changes, which happens whenever getSteering gets called.

Vitale and Reynolds' CA ultimately learned in a statistical sense that the most successful caribou herds, i.e. those herds who succeeded in getting the largest number of caribou safely across the Land Bridge within Vitale's virtual environment, have initial wander targets located about 5° North of North-East and jitter values close to zero, which produce caribou that wander very little from the herd. Vitale's program has no separate algorithm controlling caribou group kinematics on top of his CA controlling caribou individual kinematics. However, group kinematics are implicitly learned through the CA since the caribou implicitly learn that straying from the group as little as possible vastly increases their chance of survival.

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 masters thesis "Serious Game Modeling Of Caribou Behavior Across Lake Huron Using Cultural Algorithms And Influence Maps" [Fogarty, 2011]. Fogarty contributed by providing a herd-level path-planning

A* algorithm designed to take caribou from one end of the Alpena-Amberley Land Bridge to the other.

2.3.1 Influence Map

As a resource for his Cultural Algorithm, Fogarty proposed an agent-based system which creates a composite influence map from three influence map “layers” each containing one of the three following factors: The availability of food within each square (topographic knowledge), the caribou deaths within each square (situational knowledge), and the difficulty of the square's terrain (topographic knowledge -- peaks and valleys are considered "difficult terrain", as opposed to level ground which is considered "easy terrain"). These three influence map “layers” of the complex system are combined together to produce the composite influence map containing the final vertex weights used in Fogarty’s A* algorithm.

2.3.2 A* Algorithm

In Fogarty's program, the land bridge map is a navigation map composed of grid cells. A waypoint in the graph is the center of each cell. The path to be produced is a connected sequence of waypoints. The program uses the A* algorithm to create a path from a given start location to a given finish location [Fogarty, 2011].

The A* algorithm itself is an extension of Dijkstra's algorithm with a heuristic 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 all other reachable points as taken from Graph Theory

[Dijkstra, 1959]. A* is a heuristic extension of Dijkstra's Algorithm that finds the optimal path between two points. "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*. The path is guaranteed to be optimal if the heuristic used is determined to be "admissible". In other words, it is always a conservation estimate of the distance from one point to another. Euclidean distances, for example, are admissible [Yao, 2010].

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 a representative portion of the land bridge based on these weights. When Fogarty's A* algorithm actually generates the path, it is visualized as a series of blue diamonds projected onto the program's GUI display, as shown in Figure 8.

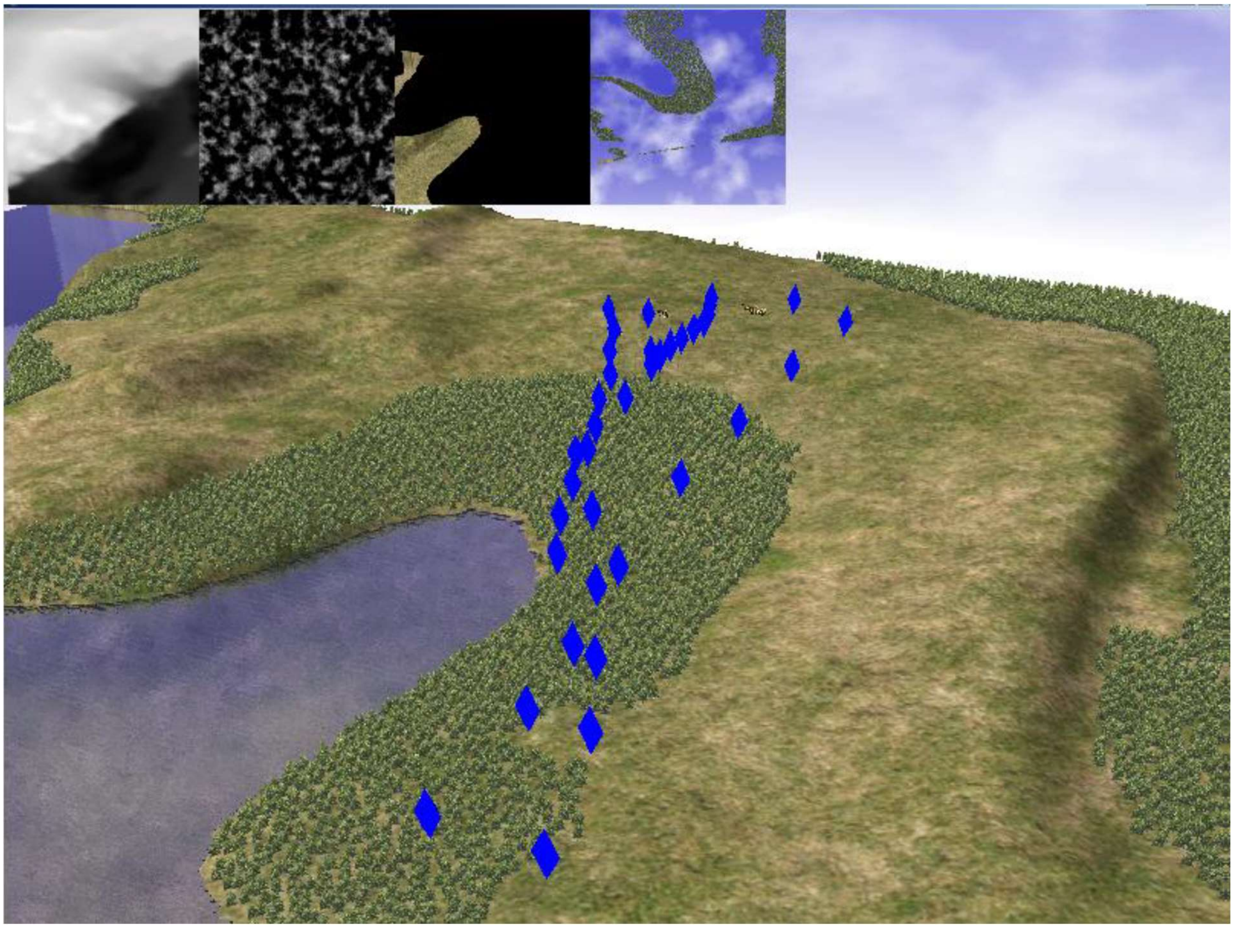


Figure 8: An Optimal Path Produced by Fogarty's CA Over a Portion of the Land Bridge

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

Jin Jin, in his thesis entitled "Path Planning in Reality Games Using Cultural Algorithm: The Land Bridge Example", provided an extended 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 factors that determine the raw value of an individual square [Jin, 2011]. 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 hard coded into the program (as they were in the 2012 version of Jin's program within Palazzolo's framework), or they can be learned using a Cultural Algorithm.

Jin's algorithm was used as the caribou path-planning algorithm in [Stanley, 2013]. In [Stanley, 2014] a multi-path variant was devised by David Warnke. Further discussion of this variant can be found in Section 2.7.

2.4.1 Geometry Value

In Jin's approach, the geometry value of a given square (g_s) is determined by the terrain that the square 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 2012 version of Jin's program effectively contained only two terrain types: those with water and those without. Water squares were given a geometry value of 255, whereas non-water squares were given a geometry value of 0.

2.4.2 Distance Value

The "distance value" (d_s) of a given square in Jin's model is the Euclidean distance from the center of that square to the center of the terminal square. The greater this Euclidean distance, the greater the distance value of the square.

2.4.3 Food Value

The "food value" (f_s) of a given square in Jin's model is the same as the vegetation value in that square. This value was taken from Palazzolo's program which provided the framework for Jin's program. 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. The greater the food value, the more desirable the square.

2.4.4 Total Value of a Square

In this model 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 equation 1, V_s is the overall value of the square, W_g, W_d, and W_f are the geometry, distance, and food weights, respectively, and g_s, d_s, and f_s are the square's geometry, distance, and food values, respectively.

2.4.5 Finding the Minimal Value-Sum Path

Jin's program finds the path from a given starting location to a given ending location 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 yields the minimal quantity for T(P), or the total combined value of all squares within path P, via 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

2.4.6 Learning Curve Diagram

The learning curve diagram for a sample run of Jin's program using his Cultural Algorithm is given in Figure 9 while Figure 10 shows the example terrain upon which he performed his experiment. The learning curve shows how the system is able to learn an improved path over time.

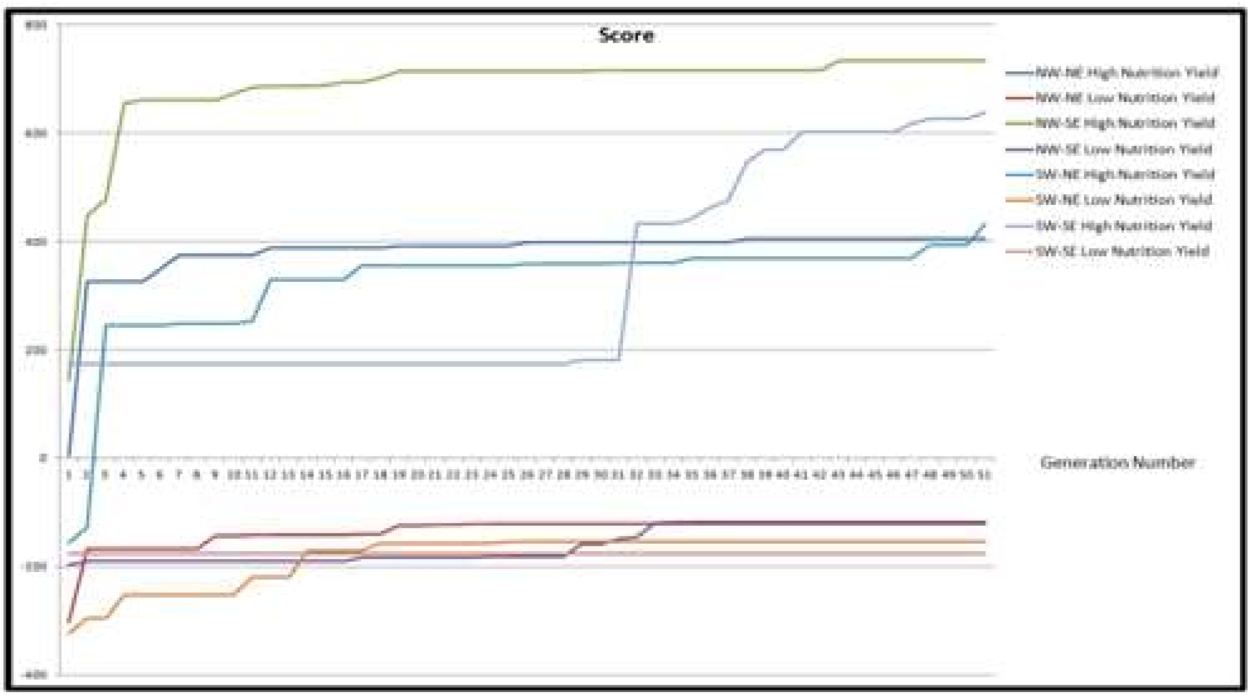


Figure 9: Jin's CA Learning Curve (Total Score vs. Generation) [Jin, 2011]

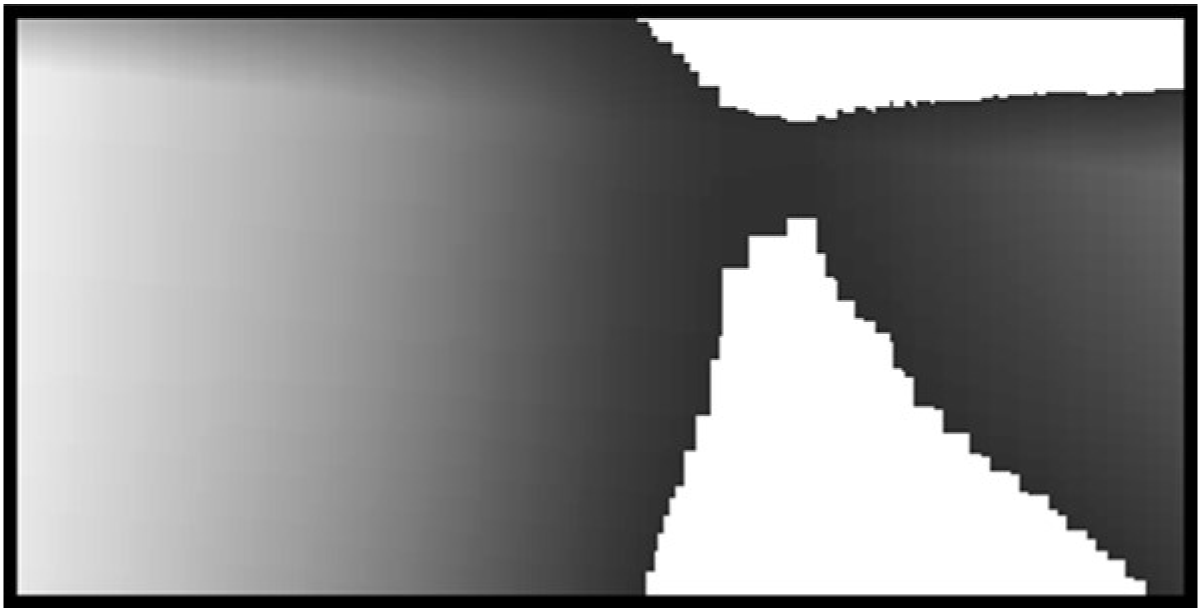


Figure 10: Jin's Experiment's Terrain Model [Jin, 2011]. The darker the color the lower the elevation.

2.5 Conclusions Regarding Previous Work

As of the time of this writing, none of the components discussed in this chapter are still in use today. They have all been replaced by something more advanced. Vitale's kinematic system discussed in Section 2.2 has long since been replaced by more advanced kinematics, the subject of which is outside the scope of this dissertation. Also, Fogarty's and Jin's caribou path-planners, discussed in Sections 2.3 and 2.4 respectively, have been replaced by a more advanced caribou path-planner developed by Thomas Palazzolo (shown in *Figure 5: Overall System Component Diagram* as "Caribou CA" and discussed later in this dissertation in *Section 4.5 Caribou Path-Planning CA*). Nonetheless, there is value in revisiting this previous work as it did lay the initial foundations for the work we have done since.

CHAPTER 3: ARCHAEOLOGICAL BACKGROUND

3.1 Alpena Amberley Ridge Phases

As the Laurentide Ice Sheet melted, the Alpena-Amberley Ridge underwent three main phases as described in the geological literature: The Algonquin Phase, the Lake Stanley Phase, and the Nipissing Phase. Of these, the Lake Stanley Phase and Early Nipissing phases could potentially have produced artifacts constructed by Paleoindian hunter-gatherers. Here we have done some work on the Early Nipissing Phase, however the main focus is on the Lake Stanley Phase since this is the phase most likely to yield discoverable artifacts. It is the phase when caribou would have been able to use the Alpena-Amberley Ridge as a crossable land corridor. The other phases can be addressed by the system in the future.

3.1.1 Algonquin Phase

When the Laurentide Ice Sheet initially receded from the Huron region, what will eventually become Lakes Huron and Michigan was a single huge body of freshwater called the Lake Algonquin (shown in Figure 11). Due to continuing meltwater inflow from the Superior Lobe and overall Laurentide Ice Sheet, the Lake Algonquin's water level continued to rise until ca. 12600 BP, reaching a level of 150m above sea level at its maximal extent [Lewis, 2007]. The massive amount of water within Lake Algonquin was held in by an unnamed lobe of the Laurentide Ice Sheet (see Figure 11) that separated Lake Algonquin from the Champlain Sea and hence the Atlantic Ocean [Dyke, 1987].

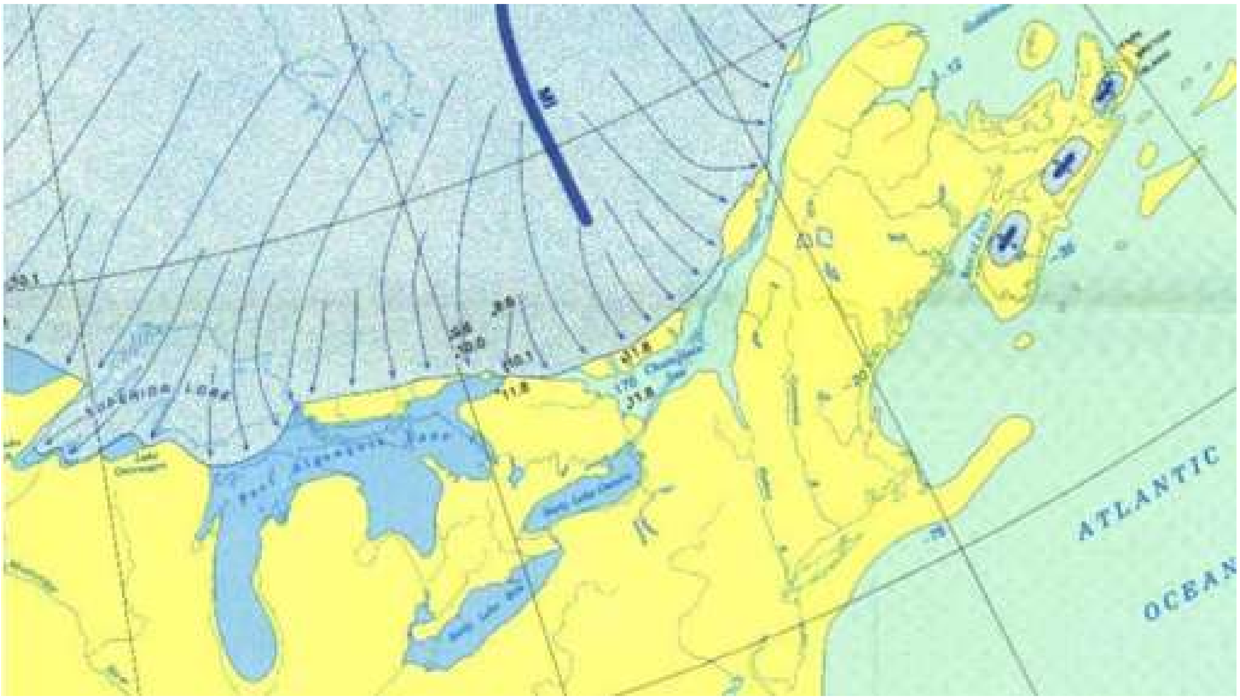


Figure 11: Algonquin Phase: Map 1703A Sheet 2 of 3 from [Dyke, 1987]

3.1.2 Lake Stanley Phase

Ca. 12,600 years BP [Lewis, 2007], the small ice lobe dividing Lake Algonquin from the Champlain Sea and the Atlantic Ocean melted. The lobe not only separated the lakewater from the ocean, it depressed the land directly underlying it [Dyke, 1987], meaning that when it melted, the force of gravity started propelling Lake Algonquin's lakewater out through the North Bay Outlet out into the Champlain Sea and hence to the Atlantic Ocean. The amount of meltwater flowing into Lake Algonquin from the Laurentide Ice Sheet was outstripped by the amount of lakewater flowing out of Lake Algonquin through the North Bay Outlet, thus Lake Algonquin's water level began gradually declining. By around 800 years later (ca. 11800 BP) [Lewis, 2007], what was once the Lake Algonquin became divided into four much smaller lakes as shown in Figure 12: Lake Chippewa in the Southwest, Lake Hough in the Northeast, Lake

Stanley, and a smaller unnamed lake that was separated from Lake Stanley by the emergent Alpena-Amberley Land Bridge, which was formerly simply a tall ridge under Lake Algonquin [Dyke, 1987].

The Lake Stanley Phase is the phase of greatest interest for Alpena-Amberley research. All of the artifacts that Dr. O'Shea has currently found are most likely to have come from this phase. During this phase, the Alpena-Amberley Land Bridge was a geographic bottleneck. Herds of migrating animals moving through it would have been relatively constrained, and Paleoindian hunter-gatherers could take advantage of that [O'Shea, 2013] [Lemke, 2016].

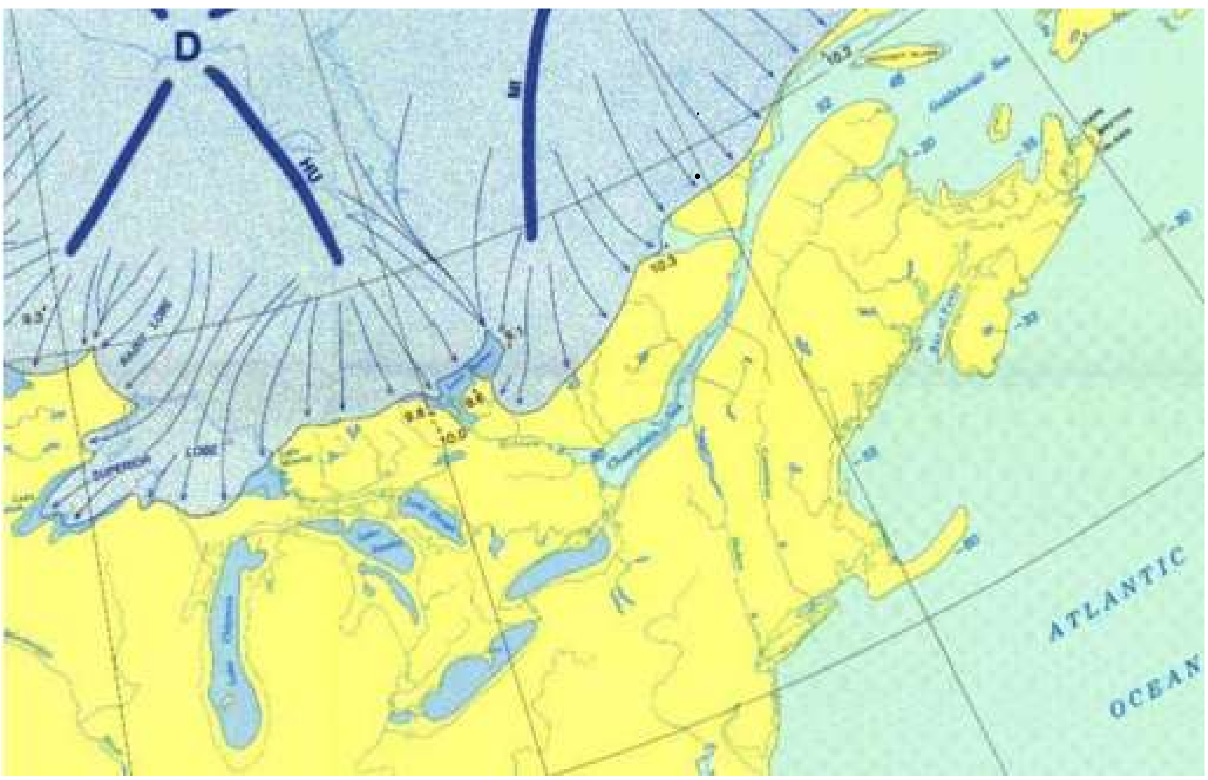


Figure 12: Lake Stanley Phase: Map 1703A Sheet 2 of 3 from [Dyke, 1987]

3.1.3 Early Nipissing Phase

Eventually, the pace of the draining of the lakewater from the Great Lakes System into the Atlantic Ocean through the North Bay Outlet started to slow due to the postglacial rebound of the area [Dyke, 1987]. In other words, the North Bay Outlet begins to "spring back up" after the ice which once compressed it has melted away, meaning that gravity is now pushing the lakewater out through the Outlet at a slower rate. After ca. 11200 BP, the water flowing from Lake Agassiz into the Huron Basin begins to exceed the water flowing out of the Huron Basin through the North Bay Outlet. the water level in the Huron Basin is able to rise. The Alpena-Amberley Land Bridge was hence overrun at a low point in the center-East by the rising water, hence Lake Stanley and the smaller unnamed lake that were formerly separated by the Land Bridge coalesced into a single larger lake as shown in Figure 13. The only remnants of the Land Bridge are two peninsulas with a smattering of small islands between them which used to be high points during the Lake Stanley Phase.

When this phase arrives, caribou were no longer able to use the Alpena-Amberley Ridge as a corridor to and from what is now Alpena, USA and what is now Amberley, Canada. However, there may still have been very choice fishing spots on the peninsulas and islands during this phase that Paleoindian hunter-gatherers could have used.

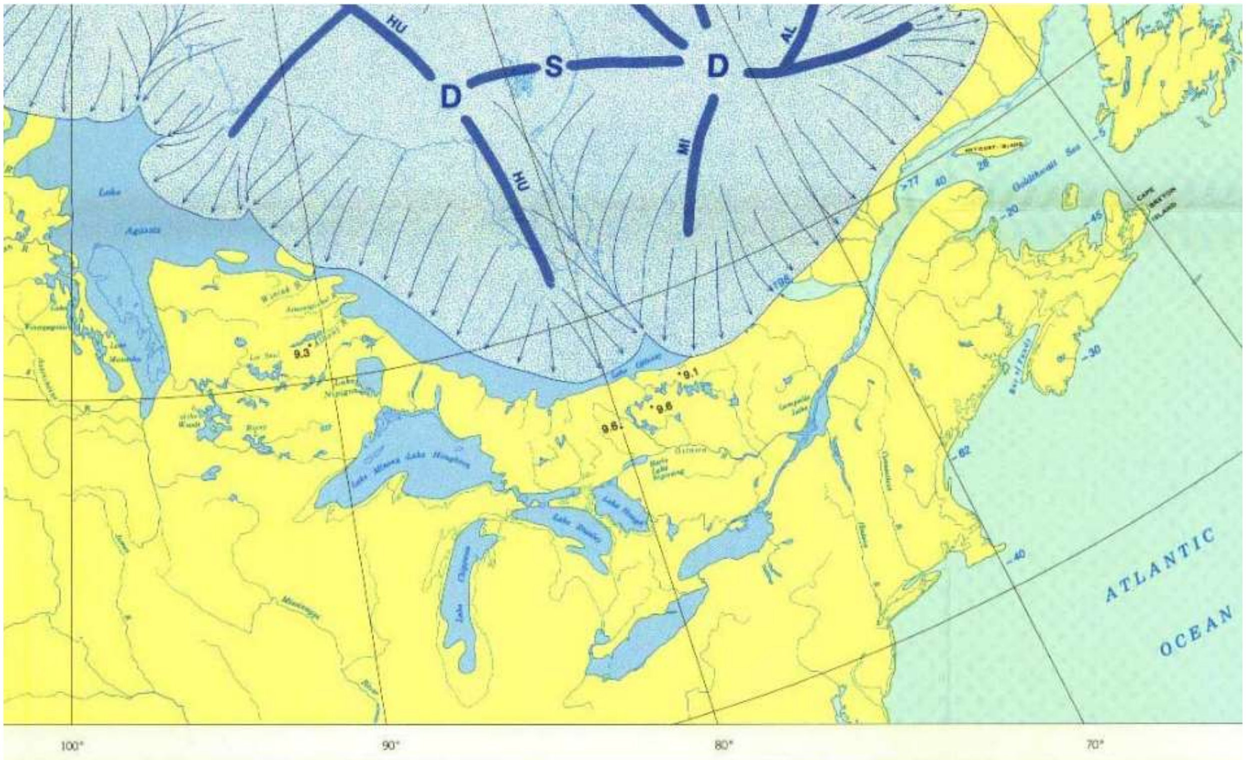


Figure 13: Early Nipissing : Map 1703A Sheet 2 of 3 from [Dyke, 1987]

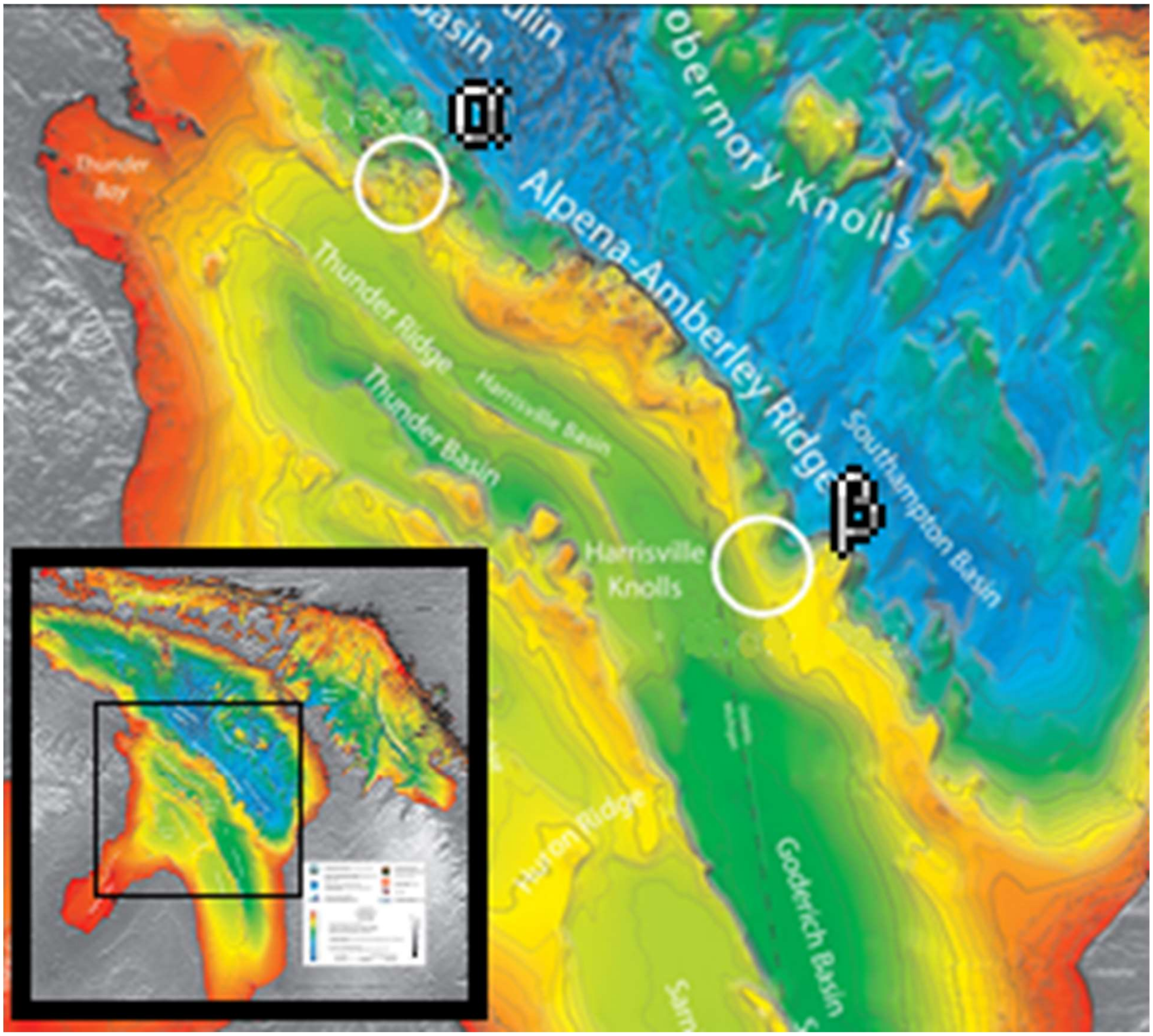


Figure 14: Alpena-Amberley Ridge Choke Points [NOAA, 2012]

Referring to Figure 14, around the time of transition from the Lake Stanley Phase to the Nipissing Phase ca. 8400 BP, point β is overrun by rising water, and the Alpena-Amberley Ridge becomes no longer a land bridge but rather two peninsulas. Afterwards when point α is overrun by rising water, the Alpena-Amberley Ridge becomes two peninsulas with knolls dotting various places in the lake between them.

3.1.4 Later Nipissing Phase

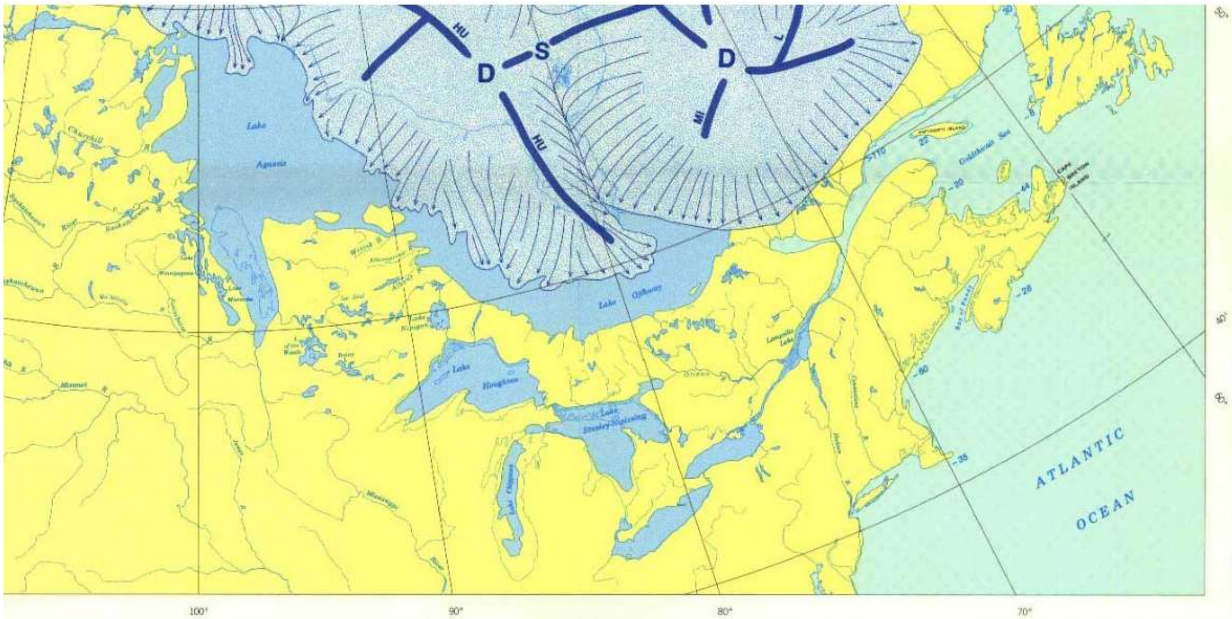


Figure 15: Later Phase: Map 1703A Sheet 2 of 3 from [Dyke, 1987]

Eventually, the inexorably rising water flowing into the Great Lakes system from Lake Agassiz causes lakes Stanley and Hough to coalesce into a single large lake called Lake Stanley-Nipissing as shown in Figure 15. At this point, all remnants of the Alpena-Amberley Land Bridge have been submerged beneath lakewater, where they remain to this day.

3.2 C. F. M. Lewis's Interpretation

In 2016, C. F. M. Lewis published "Understanding the Closed-Basin Phases (Lowstands) of the Laurentian Great Lakes and their Significance," in which he argued that the lowstand lake levels of the Early Holocene Great Lakes were significantly lower than what was surmised by earlier geologists such as Dyke and Prest. Lewis argued that outflow through avenues such as the North Bay Outlet was not the only way that the Early Holocene Great Lakes lost water during these lowstands. The lakes also lost water through unusually intense evaporation caused by an arid climate that was produced by the atmospheric effects of the remnants of the Laurentide Ice Sheet [Lewis, 2016]. We believe Lewis's arguments to be plausable, and indeed the water levels published in [Lewis, 2016] are what we are using for this dissertation work. (See Figure 25 and Figure 28 in the next chapter.)

3.3 Locating Occupational Structures

This dissertation project is mainly interested in locating occupational structures constructed by Paleoindian hunter-gatherers while the Alpena-Amberley Ridge was a crossable land corridor (ca. 11800 BP - 8400 BP). Since that region's climate at that time was semi-arctic, one would expect to see structures similar to those produced by modern-day sub-arctic hunter-gatherers such as the Nunamiut studied by Lewis Binford [Binford, 1978b]. Using this previous work as a guide, we provide a list of several types of occupational structures associated with sub-arctic hunter-gatherer communities that we expect to find using our system. The structure types listed here are by no means an exhaustive list of every structure type that could conceivably be left by a Paleoindian hunter-gatherer group. However, in creating this list we have to take into account the ability of each occupational structure to have been both preserved and also be identifiable by modern-day archaeologists. Although freshwater is an excellent preservant, it is still the case that some types of structures will have fared worse than others in withstanding thousands of years of the ravages of time. Thus, the selected occupational structures are generally bigger, heavier structures such as camps, hunting blinds, and drive lines. However, it should be recalled that smaller artifacts are often found near larger artifacts which they are relevant to.

3.3.1 Hunting Blind

Hunting blinds of the type found on the Alpena-Amberley Ridge are structures made of several large stones that form a rough enclosure for a particular space. Their most obvious purpose was to keep the animals from seeing the hunters, so the animals would wander into spear or atlatl range where they could be killed. Hunting blinds may be either circular (as shown in Figure 16) or V-shaped (as shown in Figure 17). The V-shaped blinds would be useful only during a particular season, depending upon the predominant direction of game movement. V-shaped blinds facing north would have been used in the fall, whereas V-shaped blinds facing south would have been used in the spring.

Figure 16: Photo taken in June 2011 of the "Dragon Blind", a feature found in Area 1 which is thought to be a prehistoric hunting blind [Sonnenburg, 2015].

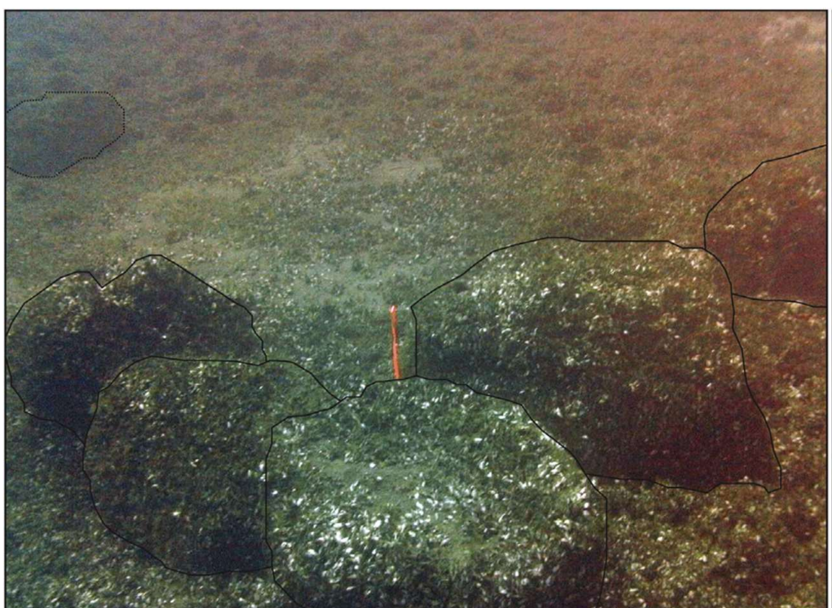


Figure 17: "V-shaped" hunting blind found in Area 3 [O'Shea, 2013].

3.3.2 Drive Lane

According to O'Shea, drive lines, also referred to in the literature as "drive lanes" are "a feature designed to channel movement toward a predictable kill zone". Because "the occupants of the AAR [Alpena-Amberley Ridge] were not interested in creating a lot of extra work for themselves" [Sonnenburg, 2015], drive lanes on the Alpena-Amberley Ridge were often augmented by straight-edges within the natural terrain [O'Shea, 2013] as well as other "natural alignments and barriers that this post-glacial landscape offered" [Sonnenburg, 2015]. In environments such as the AAR, other structures such as hunting blinds are often found in association with drive lines because both can work as a single system in order to maximize the potential for killing caribou [O'Shea, 2013] [Sonnenburg, 2015].

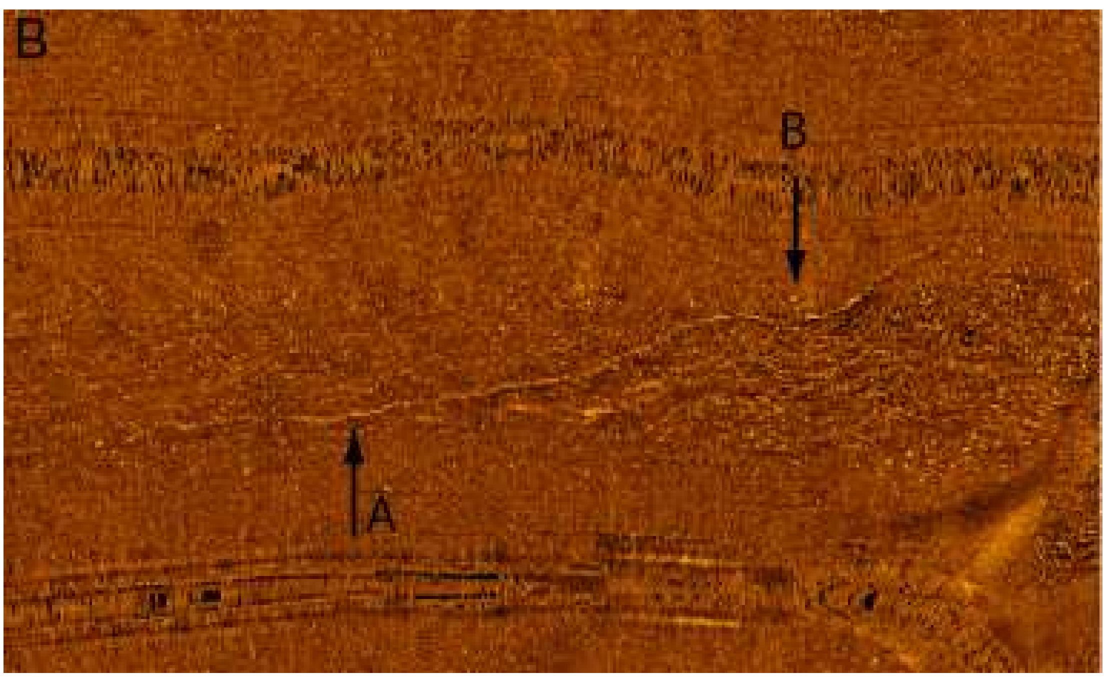


Figure 18: Acoustic Image of Dragon Drive Lane (A) in Assoc. with Dragon Blind (B) [O'Shea, 2009]

3.3.3 Observation Stand

Figure 19: Observation Stand at Kollutuk. Fig 7.37 in [Binford, 1979]

Lewis Binford describes an observation stand, also referred to in the literature as an "observation site", as "a station […] which is occupied and used basically for collecting information on game presence or movement" [Binford, 1980]. Once prey is found, the observer would signal to hunters in waiting that the prey has arrived. Since there were of course no telephones or radio signals in prehistory, ancient observers would probably have lit a signal fire to indicate the presence of prey. Typically, observation stands are located on high points overlooking lower points [Binford, 1980]. If archaeologists were to find a fire ring on a high point, they might surmise that this was a prehistoric observation stand. Figure 19 is an example of an observation stand described by Binford from his studies of the Nunamiut [Binford, 1979].

3.3.4 Residential Camp

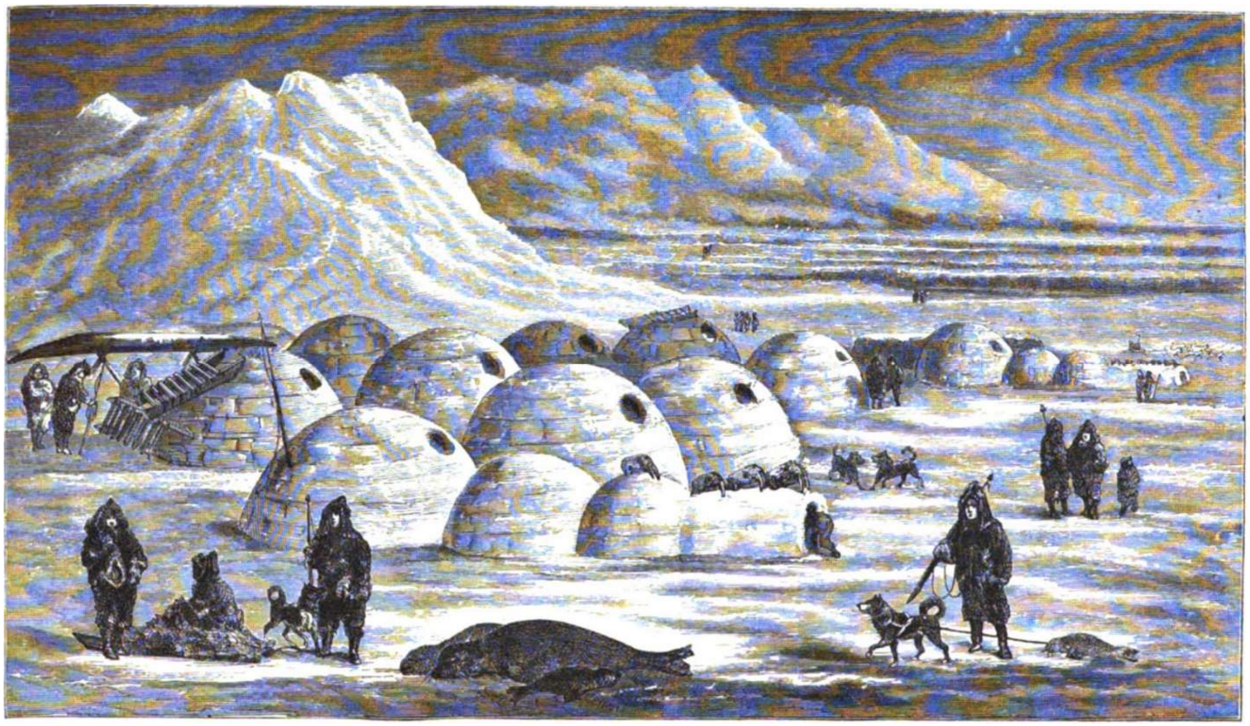


Figure 20: Igloos of Snow Village at Oo-Pung-Ne-Wing [Hall, 1865]

According to Binford, a residential camp, also referred to as a “residential base”, “residential village”, or “village”, is “the hub of subsistence activities, the locus out of which foraging parties originate and where most processing, manufacturing, and maintenance activities take place” [Binford, 1980]. It also provides the central living quarters for the hunter-gatherer tribe. According to Dr. John O’Shea, a successful find of a residential campsite is considered the "Holy Grail" of the project due to their central position within the hunter-gatherer economy. Figure 20 shows an example of a residential village located on Oo-Pung-Ne-Wing Island in the modern-day province of Nunavut located in the Canadian High Arctic.

3.3.5 Logistical Camp

Figure 21: Model of Alaska Eskimo House Group [Gleason, 1915]

Binford describes a "logistical zone" as "the zone which is exploited by task groups who stay away from the residential camp at least one night before returning" [Binford, 1982]. This zone begins to be exploited as the area immediately around the residential camp begins to become less productive due to overexploitation. These logistical zone task groups mentioned by Binford also often build a "logistical camp" so they will have somewhere to sleep since they are going to be away from the main residential camp for one night or more. If archaeologists find a camp that resembles a mini-version of a residential camp, they might take this to be a prehistoric logistic camp. For the purposes of the Alpena-Amberley Land Bridge research project, a logistical camp would probably be the second-most valuable find besides a main residential camp. Figure 21 provides a diorama model of a logistical camp.

3.3.6 Fishing Field Camp

Figure 22: Fishing Field Camp [UWLSC]
University of Washington Libraries, Special Collections, AWC6362

Binford describes a fishing field camp is a small camp where fishermen base their fishing operations [Binford, 1980]. In our particular case, a cache of prehistoric fishhooks might be indicative of a prehistoric fishing field camp. Also, a location containing multiple prehistoric fishhooks wedged in rocks and logs might indicate that a prehistoric fishing field camp was located nearby. However, the most telling finds would be the remnants of a fish trap or fish drying racks, such as those displayed in Figure 22, which is a photograph taken in the early 20th century of a small fishing field camp built by modern-day Inuit near the native village of Ekuk, located on the Nushagak River in Alaska. A fishing field camp would be an excellent find since it would be interesting to find out what type of fishhook and fish trap technology the Alpena-Amberley hunter-gatherers might actually have had. Multiple fishing field camp finds from various time periods would be even better because archaeologists could see the progression of fishhook evolution among hunter-gatherers in the region.

3.3.7 Small Game Trapping Structure

Figure 23: Inuit Fox Trap [Stopp, 2002]

Figure 23 is an example of a small game trapping structure, specifically a fox trap, built by ancient Inuit. These prehistoric structures had no moving parts, rather they were cleverly designed with openings just large enough so that a small animal could enter into them, but once inside, the animal could not get out.

3.3.8 Cache

Figure 24: Inuit Meat Cache [LAC, 1930]

The purpose of a cache (example shown in Figure 24) is to store recently killed meat for later consumption. Thus, caches are typically found near kill sites [Binford, 1980]. Prehistoric caches were essentially a crude, pre-modern version of today's electric freezers which only worked when temperatures were low enough for meat to naturally freeze: In order to prevent spoilage, the meat would have had to be cached during the fall or winter.

3.4 Conclusions

This concludes our review of the geological and archaeological literature regarding Early Holocene conditions pertinent to the Alpena-Amberley Land Bridge and what structures may have existed there. We now discuss how we create a virtual model of this ancient environment.

CHAPTER 4: MODELING THE PREHISTORIC ENVIRONMENT

4.1 Introduction to Virtual World System

In the previous chapter, we provided an overview of the paleogeology of the Alpena-Amberley Ridge Region during the Early Holocene along with the occupational structure types that might be expected to have existed there during that time. The next step is to create an actual computer model of the Alpena-Amberley Ridge Region during the Early Holocene. To that end, the Wayne State University Land Bridge Team co-created the Virtual World System. A component diagram for the Virtual World System can be found in Figure 25.

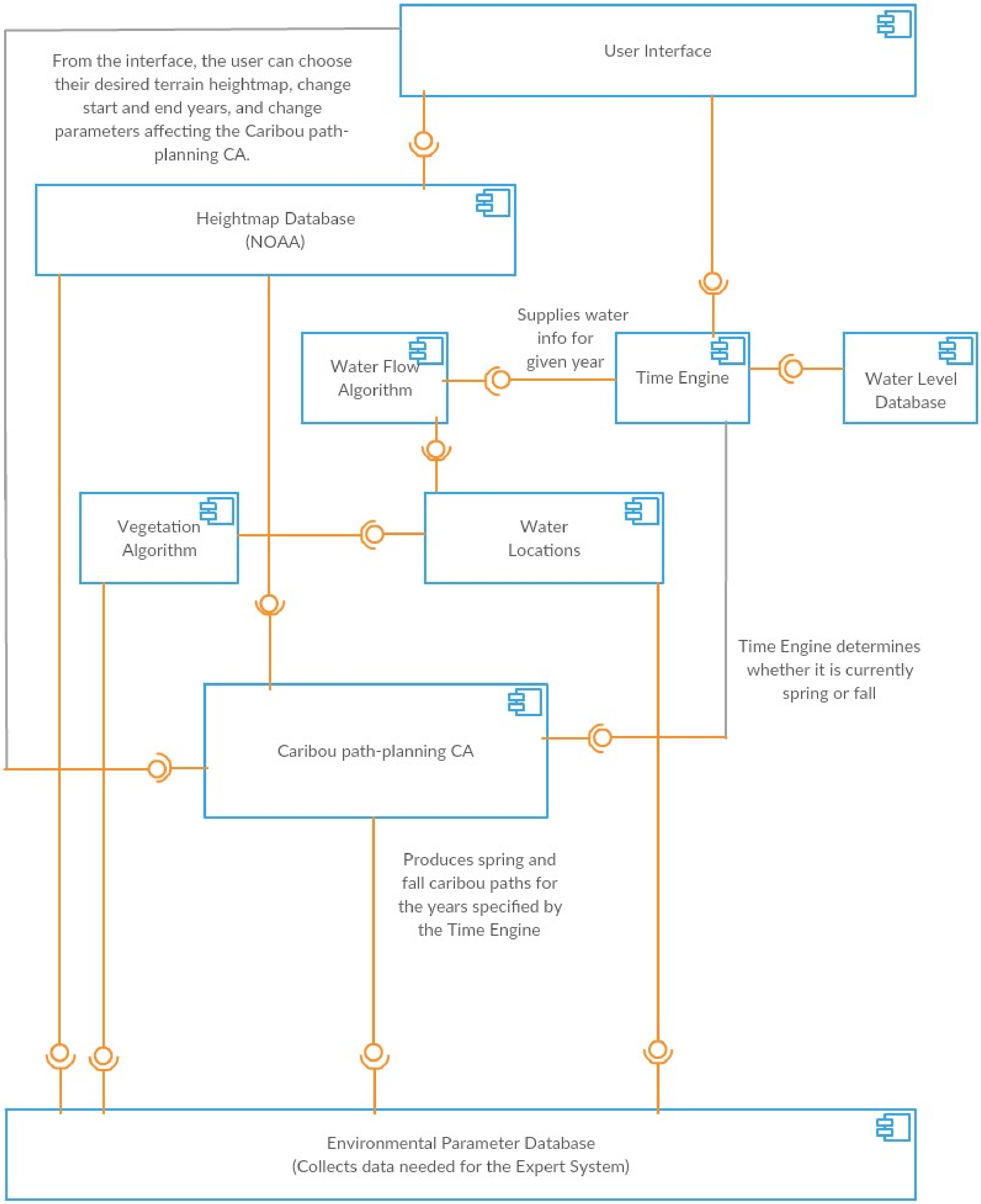
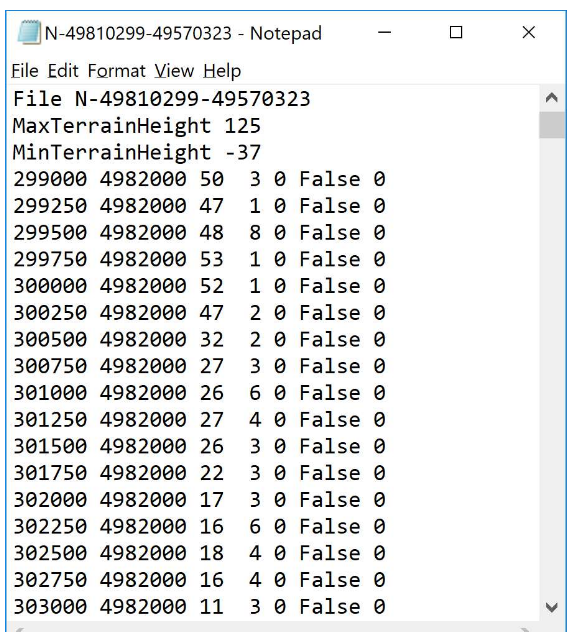


Figure 25: Virtual World Subsystem Component Diagram

4.1 Experimental Area and Heightmap

Topographical data from the National Oceanic and Atmospheric Administration (NOAA) was used as the basis for our Virtual World model. We decided to divide the Alpena-Amberley Ridge Region into 14 regional data files, each of which represents regions of 25km x 25km in dimension, each of which is further divided up into 10,000 data points. Each of those data points itself is 250m x 250m in dimension. Figure 26 contains an example segment from one of these region data files. Each data point contains initial information on latitude, longitude, water flow direction, terrain height, vegetation level, and whether the point is contained within a standing water body. Given this information, a simulated topography for the Land Bridge can be automatically constructed, as seen in Figure 27.


N-49810299-49570323 - Notepad

File Edit Format View Help

```
File N-49810299-49570323
MaxTerrainHeight 125
MinTerrainHeight -37
299000 4982000 50  3 0 False 0
299250 4982000 47  1 0 False 0
299500 4982000 48  8 0 False 0
299750 4982000 53  1 0 False 0
300000 4982000 52  1 0 False 0
300250 4982000 47  2 0 False 0
300500 4982000 32  2 0 False 0
300750 4982000 27  3 0 False 0
301000 4982000 26  6 0 False 0
301250 4982000 27  4 0 False 0
301500 4982000 26  3 0 False 0
301750 4982000 22  3 0 False 0
302000 4982000 17  3 0 False 0
302250 4982000 16  6 0 False 0
302500 4982000 18  4 0 False 0
302750 4982000 16  4 0 False 0
303000 4982000 11  3 0 False 0
```

Figure 26: A Segment of a Region Data File

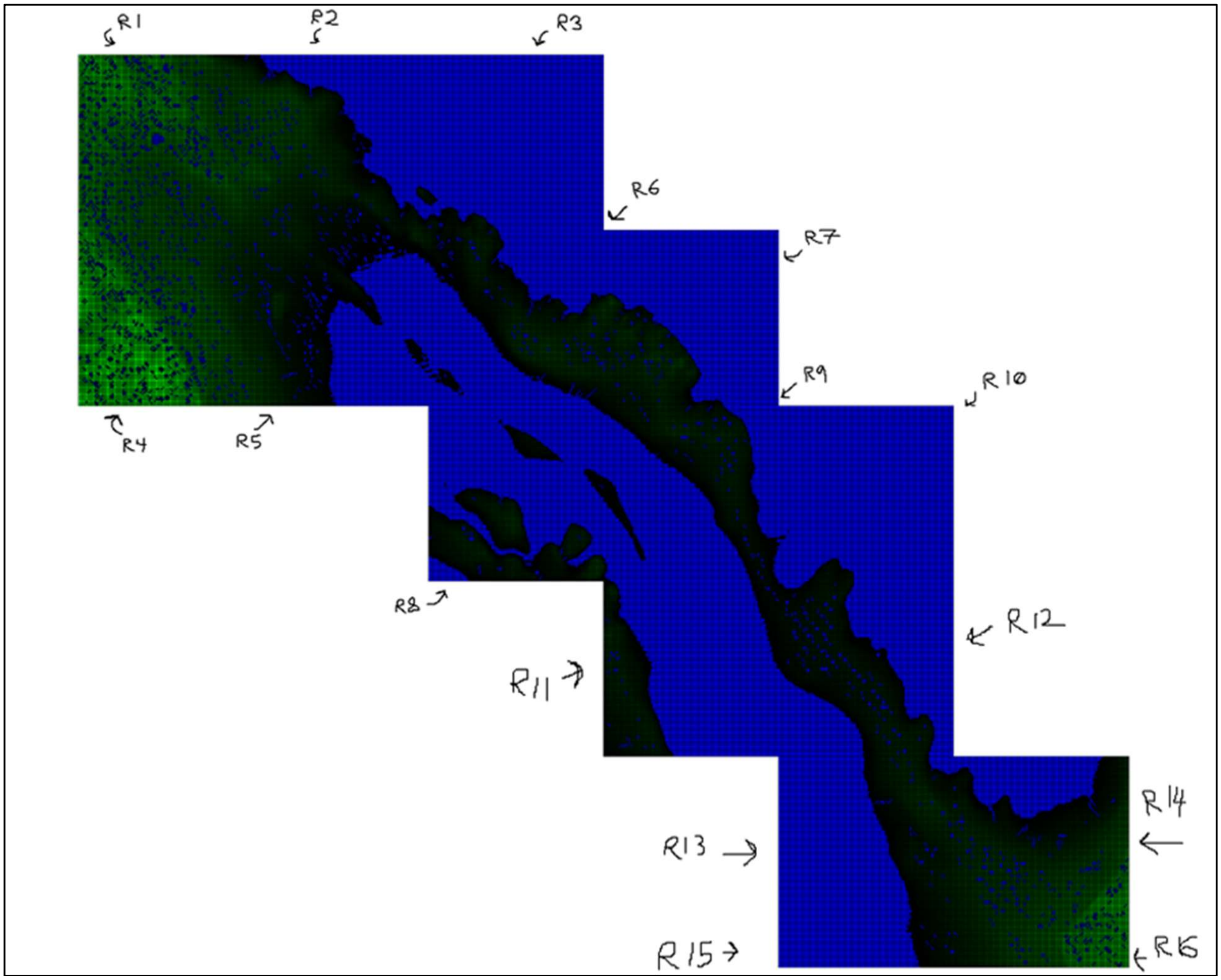


Figure 27: Regions Key

4.2 Topographic (Elevation) Modeling

The topography itself plays an important role in any accurate model of the environment. It is an important factor for both path selection by caribou and for the selection of structure locations by ancient hunters. The topography is determined by elevation data for each of the points. This data has been obtained from the National Oceanic and Atmospheric Administration

[NOAA, 2012]. For the purposes of the model, we are not taking land erosion into account in our simulation.

4.3 Water Level Modeling

The change of water levels is the single greatest source of environmental variability for the period of the simulation. This is because it determines which portions of the landscape are underwater, and hence which parts of the map are available for hunters to place their artifacts. In [Lewis, 2016], C. F. M. Lewis has provided up-to-date estimates of Huron basin water level data from 11,800 BP to 7,600 BP. These estimates came from the radiocarbon dating of the remains of various prehistoric organisms such as, tree stumps, driftwood, etc. Lewis's latest water level data [Lewis, 2016] will be used in order to produce water levels for our simulation period (11800-8400BP). This water level data is shown visually in Figure 28.

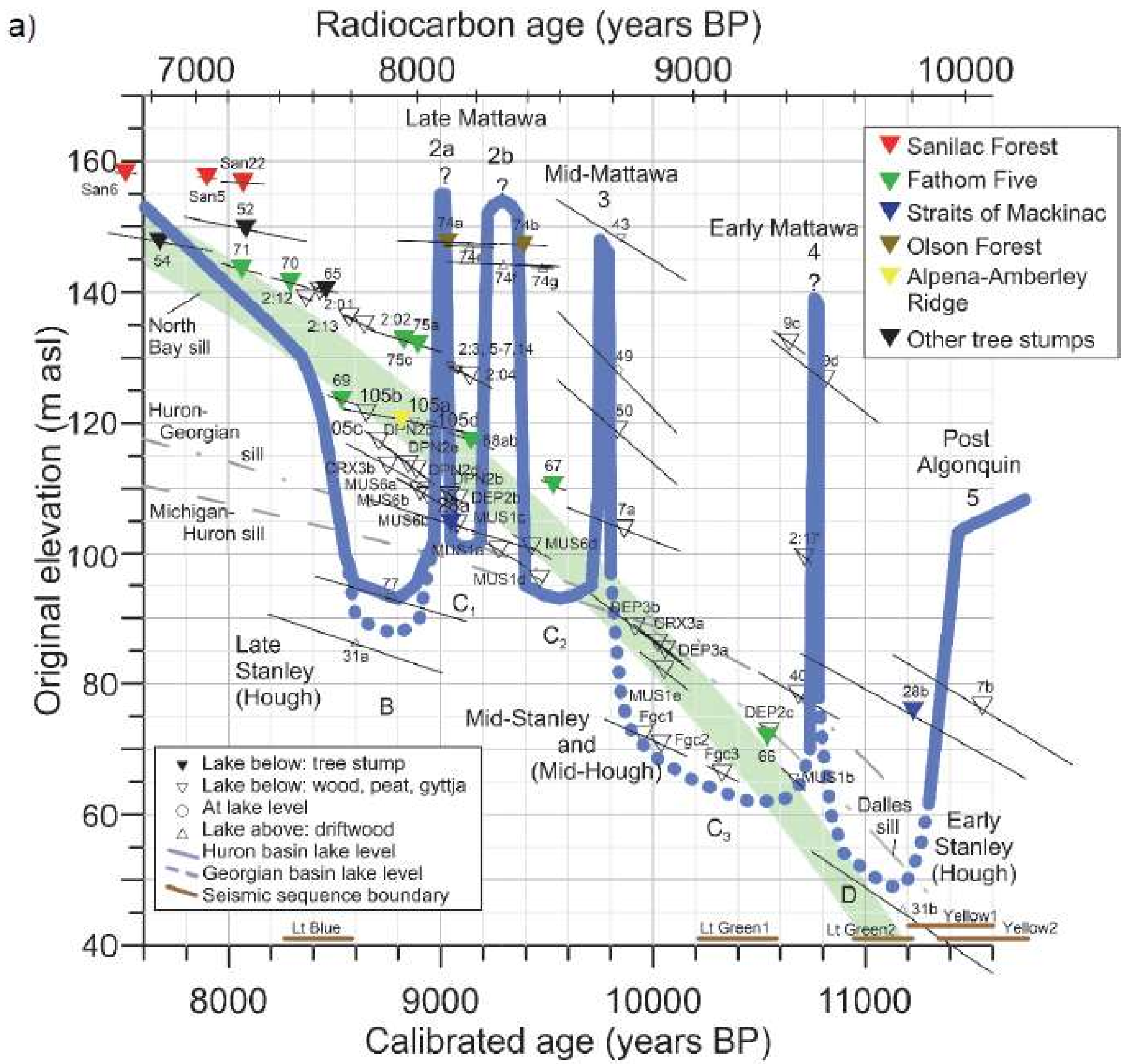


Figure 28: Lewis Water Level Data [Lewis, 2016]

Note that we did not include in our water level data the highstands indicated with labels "2a", "2b", "3", and "4" in Lewis's water level diagram. Dr. O'Shea believes that these were in fact local phenomena rather than phenomena affecting the entirety of the prehistoric Huron Basin.

4.4 Vegetation Modeling

The vegetational component of the prehistoric Alpena-Amberley Land Bridge would logically have been similar to that in modern postglacial tundra environments today. It would have consisted mostly of lichens, mosses, small shrubs, with perhaps a small smattering of fir trees.

The vegetation model in our system has been created in consultation with Anthropological Archaeologist Dr. Ashley Lemke. The model was created on the precept that for vegetation it is much more desirable to be on a slope that faces south than a north-faced slope. The amount of Vegetation *V* at point *p* is:

$$V(p) = \left[\left| \frac{1}{2} \sin\alpha + |\cos\alpha| - \frac{1}{2} \right| \cdot \sin\theta + (1 - \sin\theta) \right] \cdot (1 - \sin\theta)$$

Equation 3: Vegetation Equation

where θ is the angle of the slope of a point of land deviated from horizontal and α is the deviation of that slope from due East. *V(p)* is in the range [0.0, 1.0], where a value of 0.0 means that point p is completely bare of vegetation, while a value of 1.0 means that point p is completely covered with vegetation.

Shown in Figure 29 is a diagram of Area 1 of the Alpena-Amberley Land Bridge with the vegetation filled in. The darker areas are places with heavier vegetation, while the lighter areas contain less vegetation. The white areas are vegetation-free.

Figure 29: Area 1 Vegetation Example

4.5 Caribou Path-Planning CA

The Virtual World system needs a way to create migration routes for the virtual caribou. To this end, Thomas Palazzolo designed and implemented a path-planning CA based on the A* algorithm. This caribou path-planning CA takes an entry point and an exit direction as inputs, and then plots waypoints across the landscape, which the virtual herds follow to their destination. (See “Caribou path-planning CA” in Figure 25.)

4.6 Time Engine

In Chapter 3, we detailed the drastic environmental changes that befall the Great Lakes Region throughout the Early Holocene. During this time, the Alpena-Amberley Ridge Region was clearly a very tenuous, volatile environment. Because environmental change over time played such a major role, it became clear that our Virtual World program would have to have a temporal component. The “Time Engine” was thus created for this purpose. Using the Time Engine, the user or the system itself can choose a year and a season, and then the Time Engine supplies the relevant environmental data from its time series databases to the Virtual World’s environmental generation engine so that it generates an accurate reconstruction of the actual Land Bridge environment corresponding to that given point in time. The Time Engine was designed to handle time series data for water levels, temperature, and any number of other environmental variables in a time series format. For the purposes of [Stanley, 2013], the team decided at the time to use only water level data in the temporal engine for sake of simplicity. However, for the purposes of the new work done in this dissertation, the Time Engine also supplies the Caribou CA with the current season so that the Caribou CA generates north-to-south migrations are generated during Fall, and south-to-north migrations during Spring.

4.6.1 Time Engine Algorithm

1. A component that needs information about a temporally-dependent quantity sends a request to the Time Engine.

2. The Time Engine picks out the database associated with the requested quantity.

   a.) The Time Engine determines the current time (Year BP, Season).

   b.) The Time Engine determines if there is a database entry in the relevant quantity database associated with the current time.

      i.) If there is, then the time engine returns the entry to the requester.
      ii.) If there isn't, then the time engine uses linear interpolation to approximate the requested quantity, then returns the approximation to the requester.

4.6.2 Time Engine and Water System

Right now, there are two components in the Virtual World Program that use the Time Engine. The first of these is the water system. When the program needs to know what the water level is for a given time, it calls the Time Engine, which provides either a direct value from the database containing the [Lewis, 2016] water level data or an interpolated value. That value tells the program where the water table is for the requested year and season. Then, the water flow component activates and uses Thomas Palazzolo's water flow algorithm to determine exactly where all the water is going to end up during the given year and season.

4.6.3 Time Engine and Caribou CA

The second usage of the Time Engine is regarding the Caribou CA. The caribou migration pattern is different depending upon the season: If it is fall, then the caribou start in the northwest and migrate to the southeast. If it is spring, then the caribou start in the southeast and migrate to the northwest. So when the Caribou CA is about to start, it sends a request to the Time Engine for a starting location and a direction based upon the current season. If the Time Engine determines that the current season is Fall, it provides the caribou with an entry point in the northwest of Region 6 and tells them to exit through the southeast of Region 9. On the other hand, if the Time Engine determines that the current season is Spring, it provides the caribou with an entry point in the southeast of Region 9 and tells them to exit through the northwest of Region 6.

4.6.4 Time Engine Example

Figure 30, Figure 31, and Figure 32 all show the same exact location (382310E, 4964730N, UTM-16) but at three different times in prehistory: 9888 BP, 7540 BP, and 7000 BP.

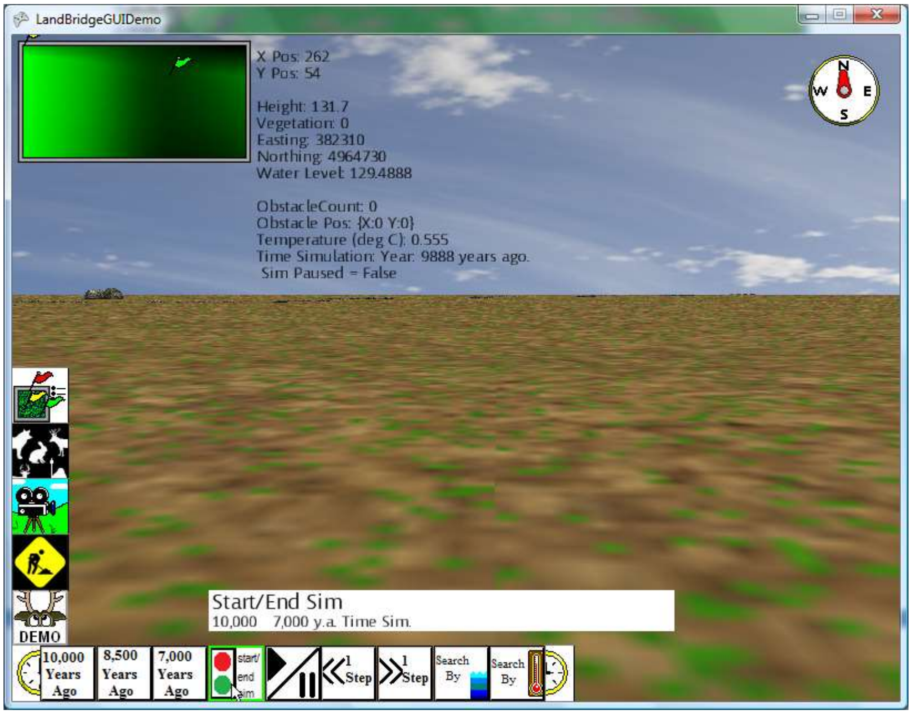


Figure 30: Test Environment 9888 YBP (382310 Easting, 4964730 Northing)

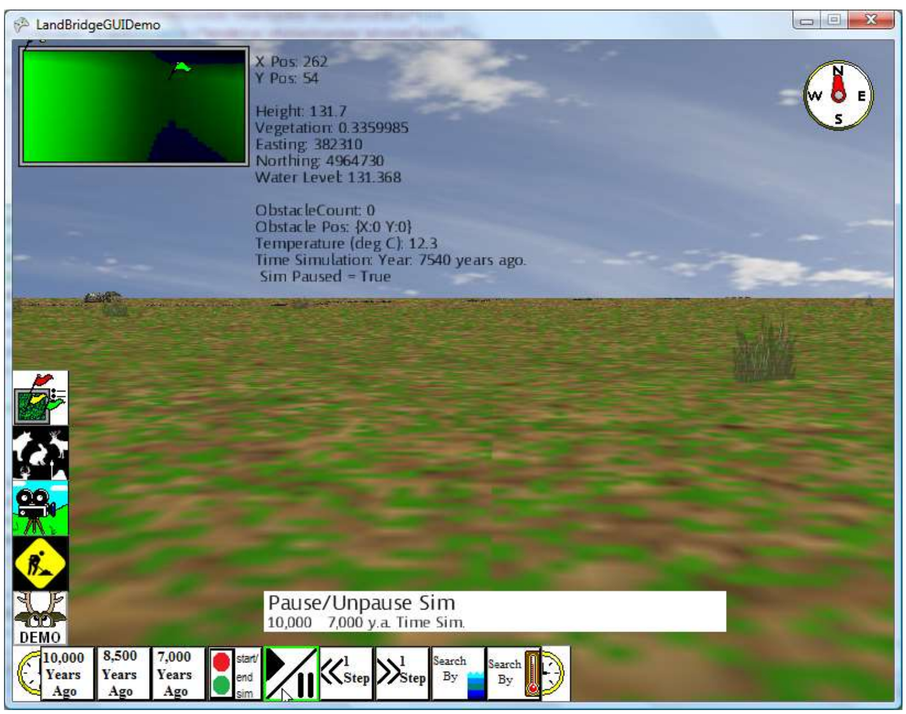


Figure 31: Test Environment 7540 YBP (382310 Easting, 4964730 Northing)

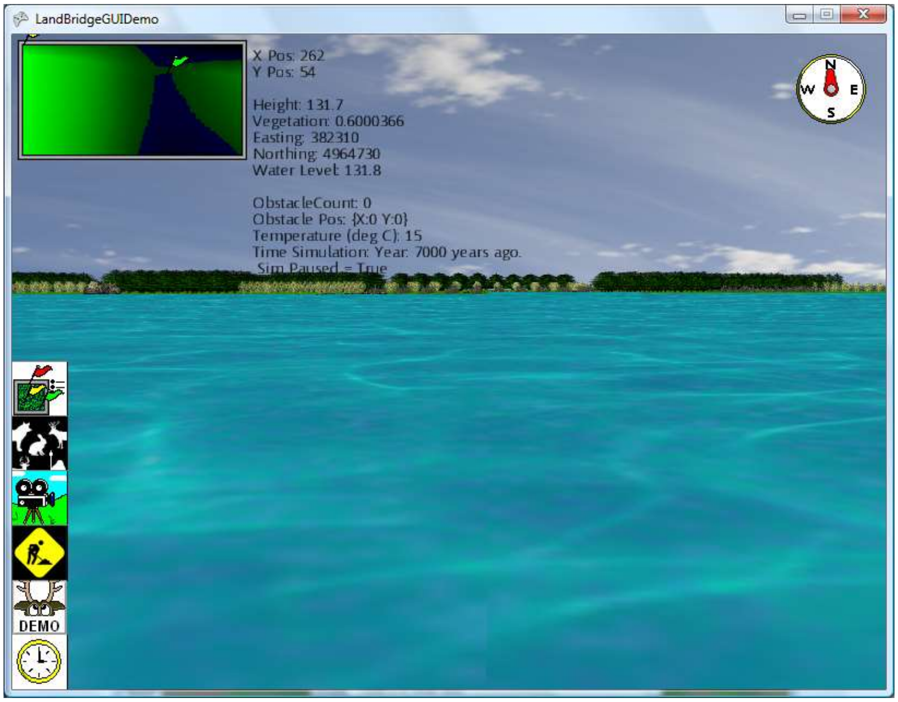


Figure 32: Test Environment 7000 YBP (382310 Easting, 4974730 Northing)

4.7 Environmental Parameter Database

With regard to the rest of this dissertation, the main purpose of the Virtual World system is to determine environmental parameter values for each of the relevant timesteps. Because the relevant time period is 11800BP-8400BP (inclusive), and we are using a timestep size of 200 years, there are 18 timesteps, each of for which 13 different pieces of data have to be collected (see Table 1 in Chapter 5) for each of the 40,000 locations in Regions 6-9. This gives a grand total of 9,360,000 pieces of data needed for the Expert System to be discussed in the next chapter. We store this data in a database (labeled "Environmental Parameter Database") in *Figure 25: Virtual World Subsystem Component Diagram*.

4.8 Conclusion Regarding the Virtual World System

The Virtual World System is an excellent tool in its own right, as it allows the user to "travel back in time" to see what the environment was like in the ancient past. As we have demonstrated in this chapter, it also serves as an excellent subsystem for this dissertation project; the collection of the data that we need for the rest of this project could not happen without it. Finally, the Virtual World System is by no means limited by the Alpena-Amberley Land Bridge metaproject. Given relevant heighmap data, water level data, and a time frame for another part of the world that archaeologists are interested in studying, the Virtual World System should be usable in other projects as well.

CHAPTER 5: AGENT-BASED APPROACH

5.1 Intro to Agent-Based Approach

The work done in [Stanley, 2013] and expanded upon in [Stanley, 2014] was the first attempt at creating a system for the prediction of sites potentially containing occupational structures. Our approach was based on two central premises. Firstly, if a location was used more frequently by ancient hunters, there is a greater chance that archaeologists will find artifacts there rather than in another location that was not used as often. Secondly, in deciding where to place their artifacts, ancient hunters were influenced by environmental conditions, their own intelligence, and the stored cultural knowledge of their society.

In this approach, referred to as the “agent-based approach”, there are a number of agents, each of which is responsible for placing an occupational structure of a given type somewhere in the landscape during each generation. Each agent has a set of beliefs about the relative importance of various factors within the environment, such as distance to the caribou trail, height above the caribou trail, and distance to the closest other occupational structure of the same type. The agents have only partial knowledge of the landscape. In other words, the agents are only able to choose certain portions of the landscape, i.e. those that have been recently explored or re-explored, to place their structures. Each agent “scores” each location within this knowledge bank according to its own personal beliefs about pertinent geographic categories, and places its structure in the highest-scoring location according to its beliefs.

There is then a single-objective fitness function which is used to calculate the “true” scores for each of the locations at the given timestep. Each of the agents is then ranked according to the “true” score of its location. The top 10% are admitted into the elite.

Meanwhile, each square within a specified radius of *any* agent becomes “discovered” (or rediscovered) and is admitted into the topographic knowledge base. Then each square in the topographic knowledge base that has not been “rediscovered” within a certain number of generations is “forgotten” from the topographic knowledge base.

Then, the elite reproduce. Genetic operators are used to create children with beliefs which are various recombinations of those of the parents. Then, the time engine moves to the next timestep, and the fitness function is made to calculate the “true scores” for each location once again since the dynamic environment has now changed. The entire process is started over again until a stop condition is reached.

Originally, the agent-based algorithm was used only to predict the locations of Hunting Blinds. However, it was eventually expanded to be able to generate prediction maps for the Observation Stand and Fishing Field Camp structure types as well.

5.2 Agent-Based Algorithm

On the following pages is a listing of the core algorithm used in the agent-based approach [Stanley, 2013]:

Definitions:

Let $\bar{B}$ be the belief space.
Let $\bar{P}$ be the population space.

Let a belief space entry $B_i \in \bar{B} : B_i = (B_{i_{lat}}, B_{i_{lon}}, B_{i_{dtc}}, B_{i_{hac}}, B_{i_{dco}}, B_{i_{dtw}})$ where $B_{i_{lat}}, B_{i_{lon}}, B_{i_{dtc}}, B_{i_{hac}}, B_{i_{dco}}$, and $B_{i_{dtw}}$ are respectively the values for the latitude, longitude, distance to the caribou trail, height above the caribou trail, distance to the closest hunting blind, and distance to water that are recorded in B_i.

Let a population member $P_i \in \bar{P} : P_i = (P_{i_{lat}}, P_{i_{lon}}, P_{i_{dtcw}}, P_{i_{hacw}}, P_{i_{dcow}}, P_{i_{dtww}})$ where $P_{i_{lat}}, P_{i_{lon}}, P_{i_{dtcw}}, P_{i_{hacw}}, P_{i_{dcow}}$, and $P_{i_{dtww}}$ are respectively the values for P_i's latitude, P_i's longitude, P_i's weight for the distance to caribou trail category, P_i's weight for the height above the caribou trail category, P_i's weight for the distance to the closest other hunting blind category, and P_i's weight for the distance to water category.

Define weight function $w : \bar{B} \times \bar{P} \rightarrow \mathbb{R}$, given for some $B_i \in \bar{B}$ and some $P_j \in \bar{P}$ by

$$w(B_i, P_j) = B_{i_{dtc}} P_{j_{dtc}} + B_{i_{hac}} P_{j_{hac}} + B_{i_{dco}} P_{j_{dco}} + Log_{10}(B_{i_{dtw}} P_{j_{dtw}})$$

Define the fitness function $f : \bar{P} \rightarrow \mathbb{R}$, given for some $P_i \in \bar{P}$ by:

$$f(P_i) = \begin{cases} -30P_{i_{dtcw}} + 50P_{i_{hacw}} + 8P_{i_{dcow}} & if\ (P_{i_{lat}}, P_{i_{lon}})\ is\ not\ underwater \\ -\infty & if\ \left(P_{i_{lat}}, P_{i_{lon}}\right)\ is\ underwater \end{cases}$$

Algorithm:

Initialize t = start year before present.

Loop While (t = t – timestepSize) ≥ end year:

(Update Environment Variables): Use the caribou algorithm and the environment engine to update dtc (distance to caribou), hac (height above caribou), dto (distance to closest other hunting blind), and dtw (distance to water) for all of the locations in the search area for time t.

(Population Placement Phase): For each P_i in $\bar{P}$, find the B_x in $\bar{B}$ that maximizes $g(P_i, B_x)$. Then, set $P_{i_{lat}} = B_{i_{lat}}$ and $P_{i_{lon}} = B_{i_{lon}}$. Then, set $P_{i_{dtc}}, P_{i_{ha}}, P_{i_{dto}}$, and $P_{i_{dtw}}$ to the corresponding dtc, hac, dto, and dtw values supplied at the location corresponding to ($P_{i_{lat}}, P_{i_{lon}}$)for time t by the environment engine. Record ($P_{i_{lat}}, P_{i_{lon}}$) in heatmap.

(Population Fitness Evaluation/Evolution Phase): $\forall P_i \in \bar{P},\ evaluate\ f(P_i)$. The bottom 90% of performers become mutated versions of the top 10%, then undergo crossover. The top 10% of performers remain unaltered.

(Belief Space Expansion Phase): Then, for each $P_i \in \bar{P}$, for each location L such that L_{lat}, L_{lon} is within a 3-square Moore radius of P_{lat}, P_{lon}, B_L *in* $B = (L_{lat}, L_{lon}, L_{dtc}, L_{hac}, L_{dto}, L_{dtw})$.

(Belief Space Culling Phase) $\forall B_i \in \bar{B}$ if B_i was not updated for ≥10 timesteps, remove B_i from $\bar{B}$.

5.3 Component Diagram for Agent-Based System

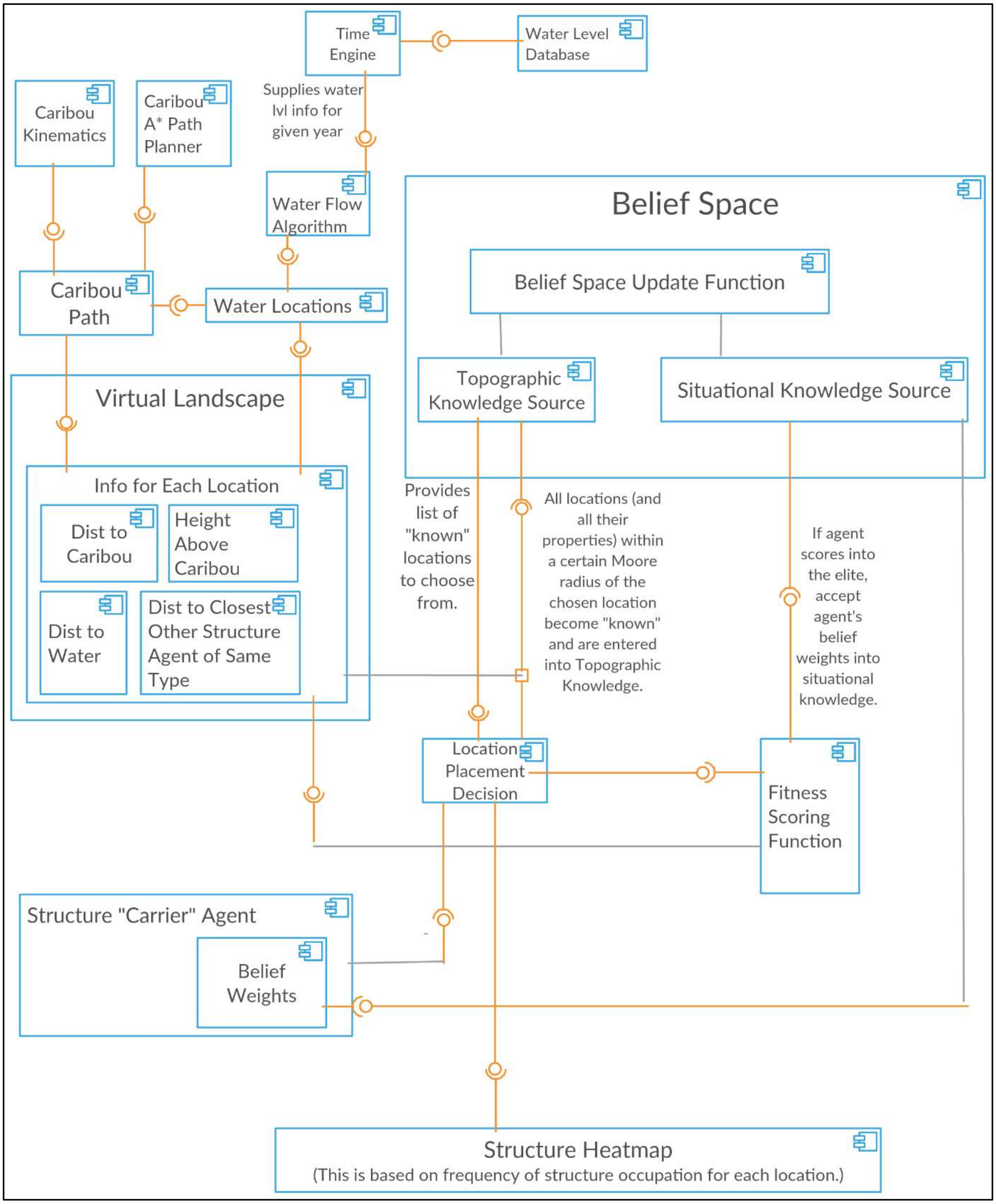


Figure 33: Component Diagram for Agent-Based System

5.4 Results

Here are several of these frames from representative years showing the learning process of the agent-based CA. We also provided the location of the Funnel Drive Structure (the most complex artifact found to date on the Alpena-Amberley Ridge [O'Shea, 2013]) within each of these year frames for comparative purposes.

5.4.1 Projected Hunting Blinds in 11800 BP

In 11800 BP, the Alpena-Amberley Ridge first became a crossable land bridge. Since this is the first generation, hunting blinds are simply placed in random non-water squares. Projection results are shown in Figure 34.

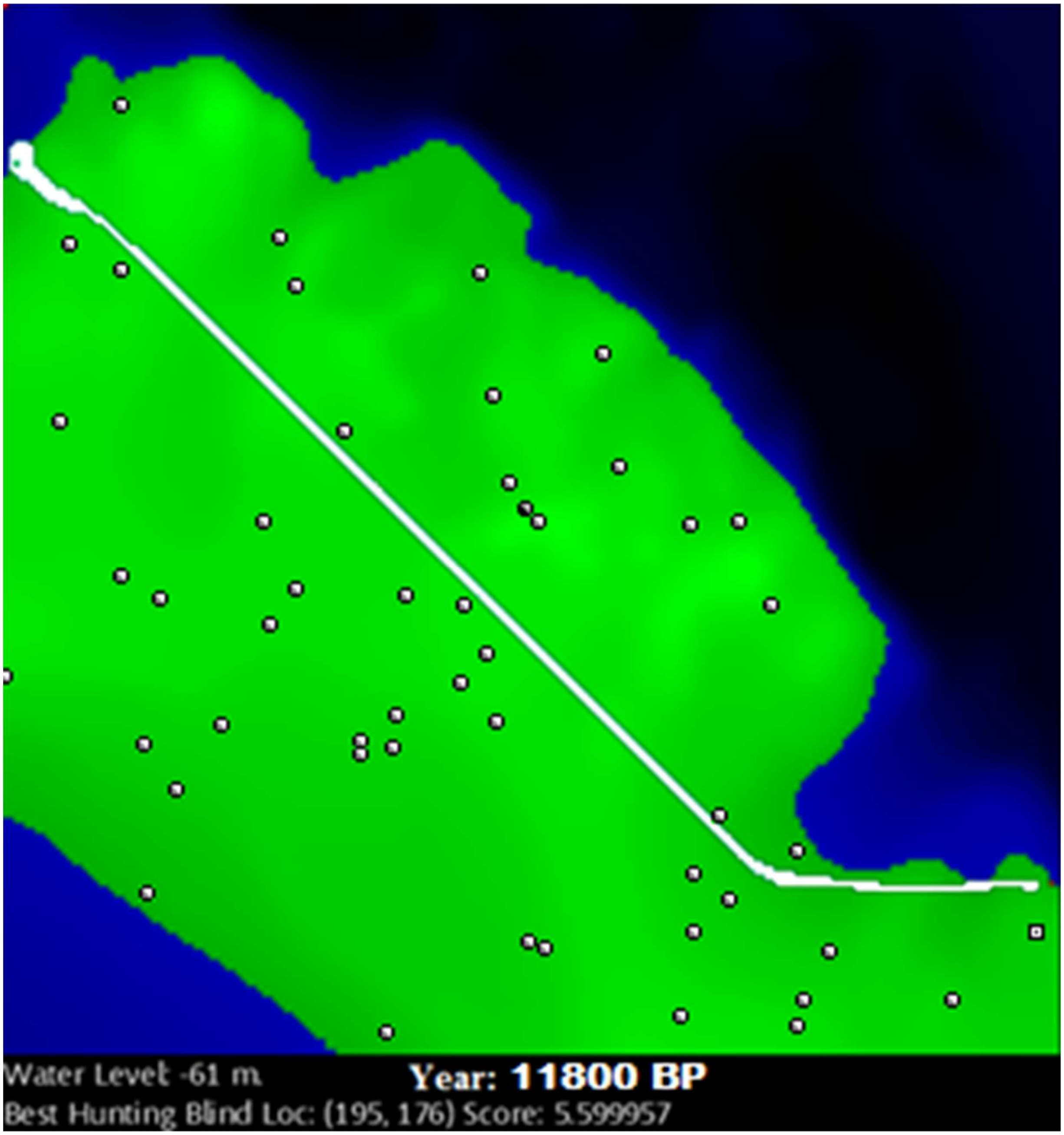


Figure 34: Projected Hunting Blinds in Land Bridge Area 1 in 11800 BP [Stanley, 2013]

Figure 35: 2013 Yearframes Map Key

5.4.2 Projected Hunting Blinds in 11750 BP

After just 10 generations (50 years), the algorithm 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 reasonable distance from one another, and hence many of the hunting blinds are still losing a lot of points as the result of tight clustering.

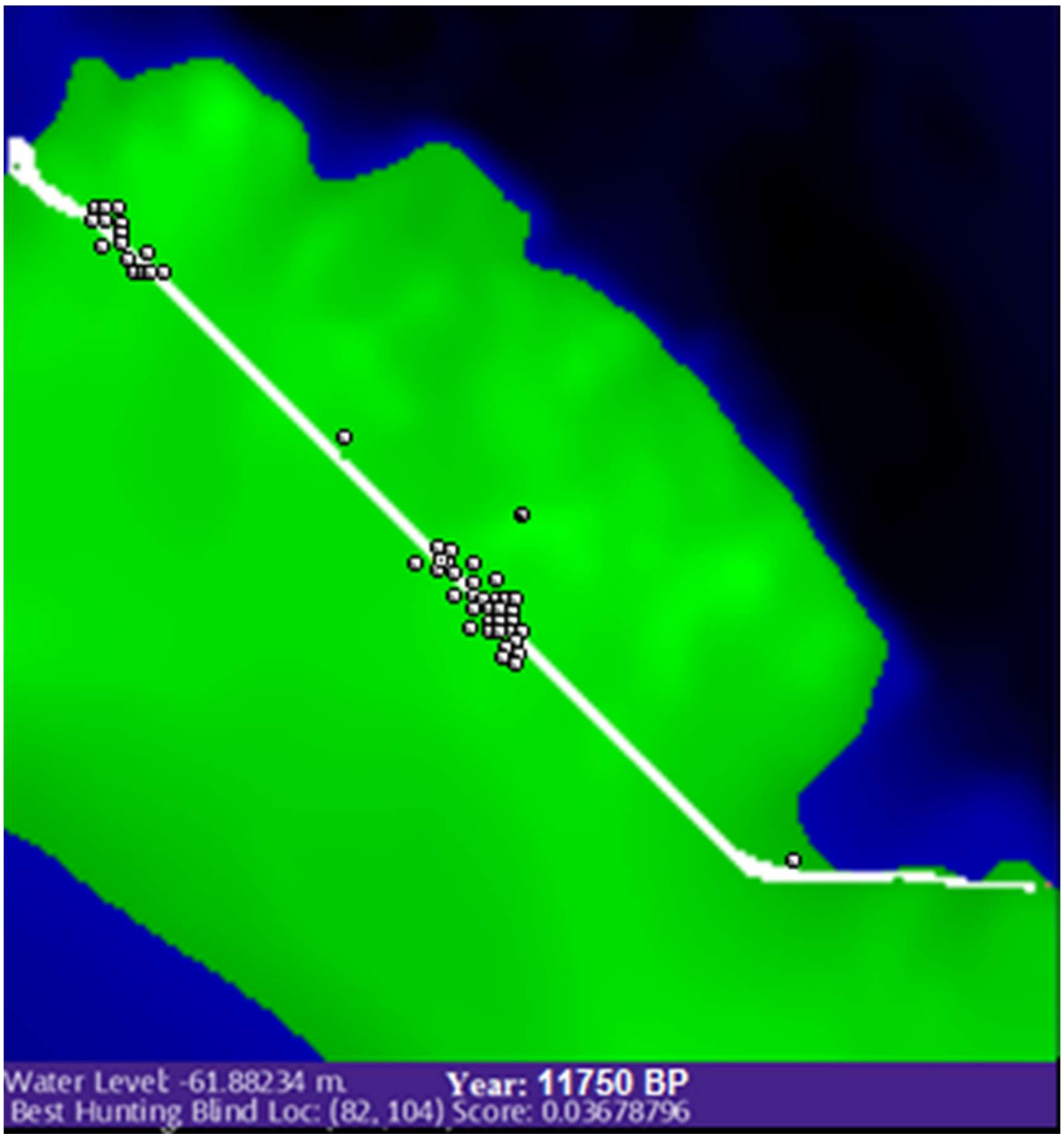


Figure 36: Projected Hunting Blinds in 11750 BP [Stanley, 2013]

5.4.3 Projected Hunting Blinds in 11700 BP

By this time the agents have learned to space out adequately, as well as to stay close to where the caribou path is most likely to be. A few are also seeking out high ground in order to gain extra points for having a vantage above the caribou. Most of the results from the individual generations from here on look more or less similar to this figure (Figure 37).

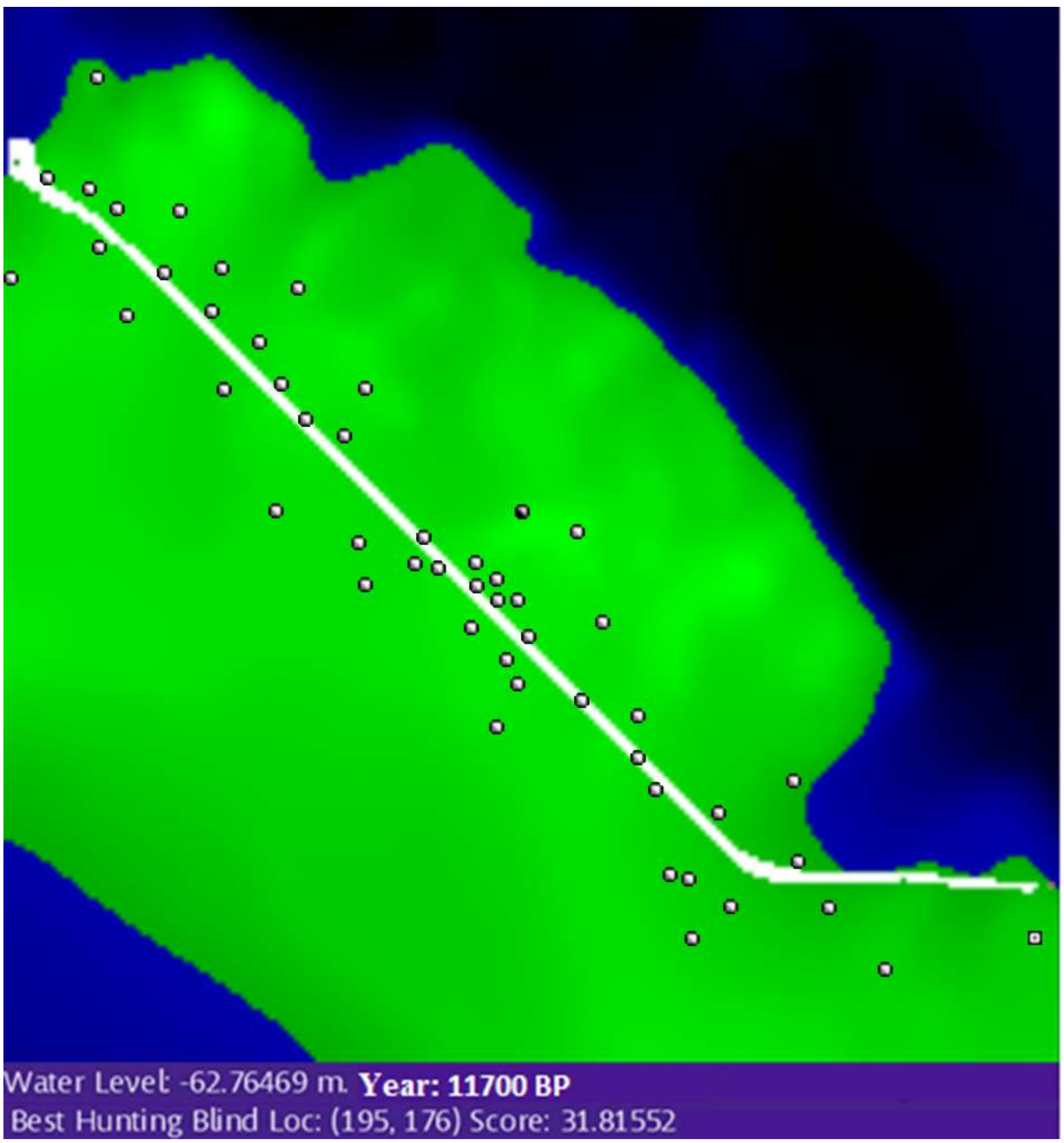


Figure 37: Projected Hunting Blinds in 11700 BP [Stanley, 2013]

5.4.4 Projected Hunting Blinds in 11370 BP

In Figure 38, when the simulated hunting blind agents chose 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 11430 BP to 10000 BP. That is when the water level is the lowest [Lewis, 2007], 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 blind agents 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, four of them have chosen the hill just a few generations into the Early and Mid Stanley periods.

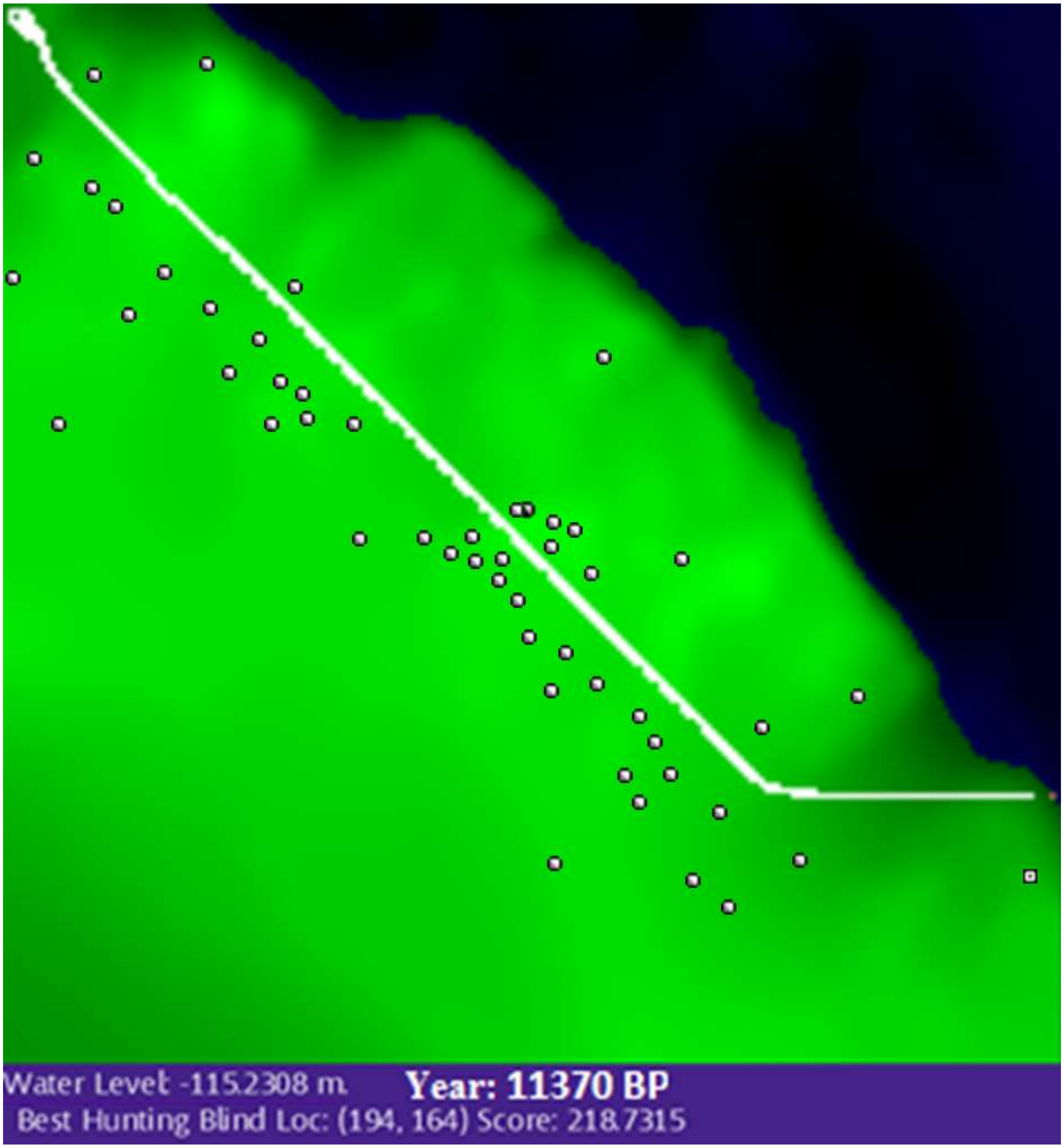


Figure 38: Projected Hunting Blinds in 11370 BP [Stanley, 2013]

5.4.5 Projected Hunting Blinds in 9400 BP

Figure 39 shows the typical behavior for the Mid-Late Stanley period, when lake levels are quite high. The caribou path is now significantly far to the southwest of the Funnel Drive Structure's location. The AI hunting blind agents now no longer have incentive to go near the drive's location again. A new desirable spot now emerges on a hill overlooking the southeastern part of the caribou path.

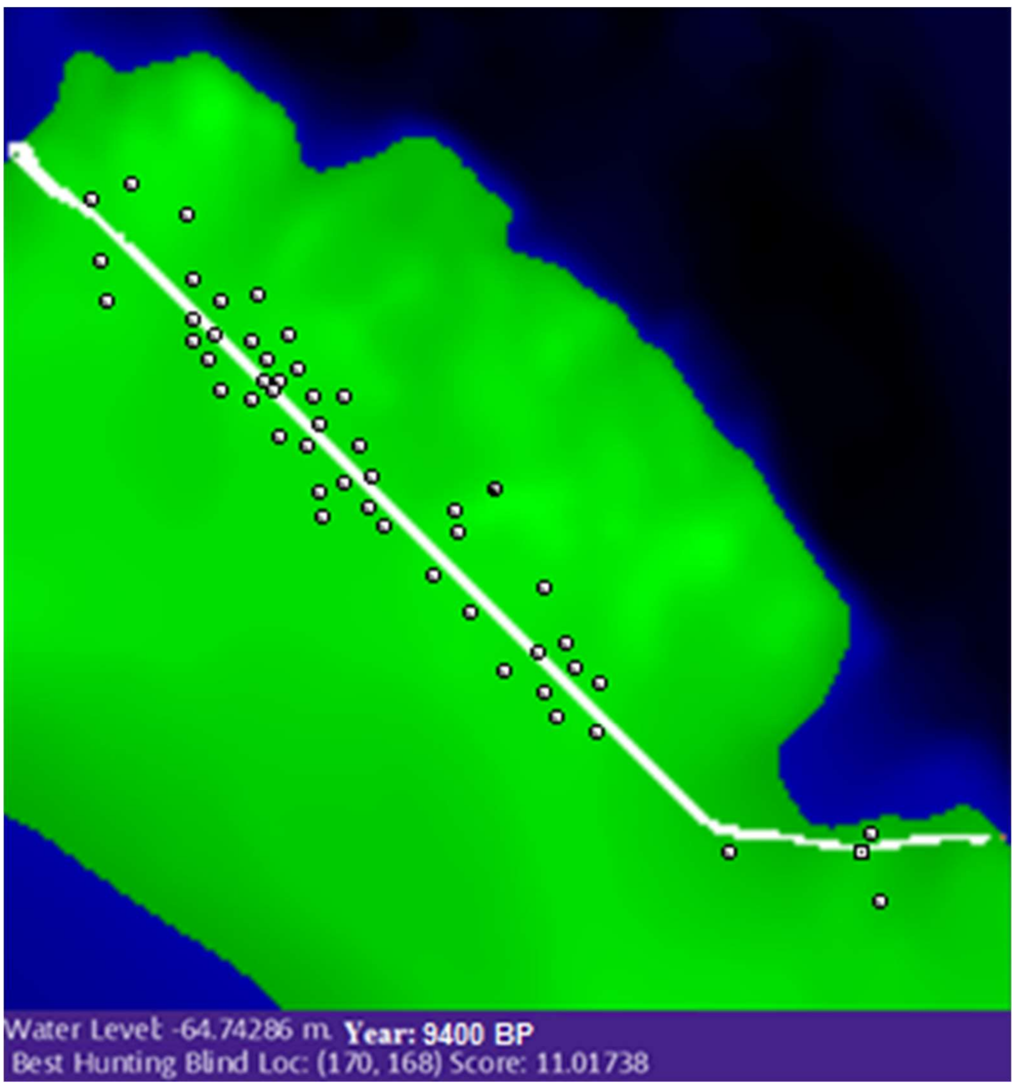


Figure 39 Projected Hunting Blinds in 9400 BP [Stanley, 2013]

5.4.6 Projected Hunting Blinds in 8360 BP

Figure 40 represents the "final hours" of the Late Stanley phase, and therefore the end of the Alpena-Amberley Land Bridge. 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 Lake Huron as lake levels continue to rise.

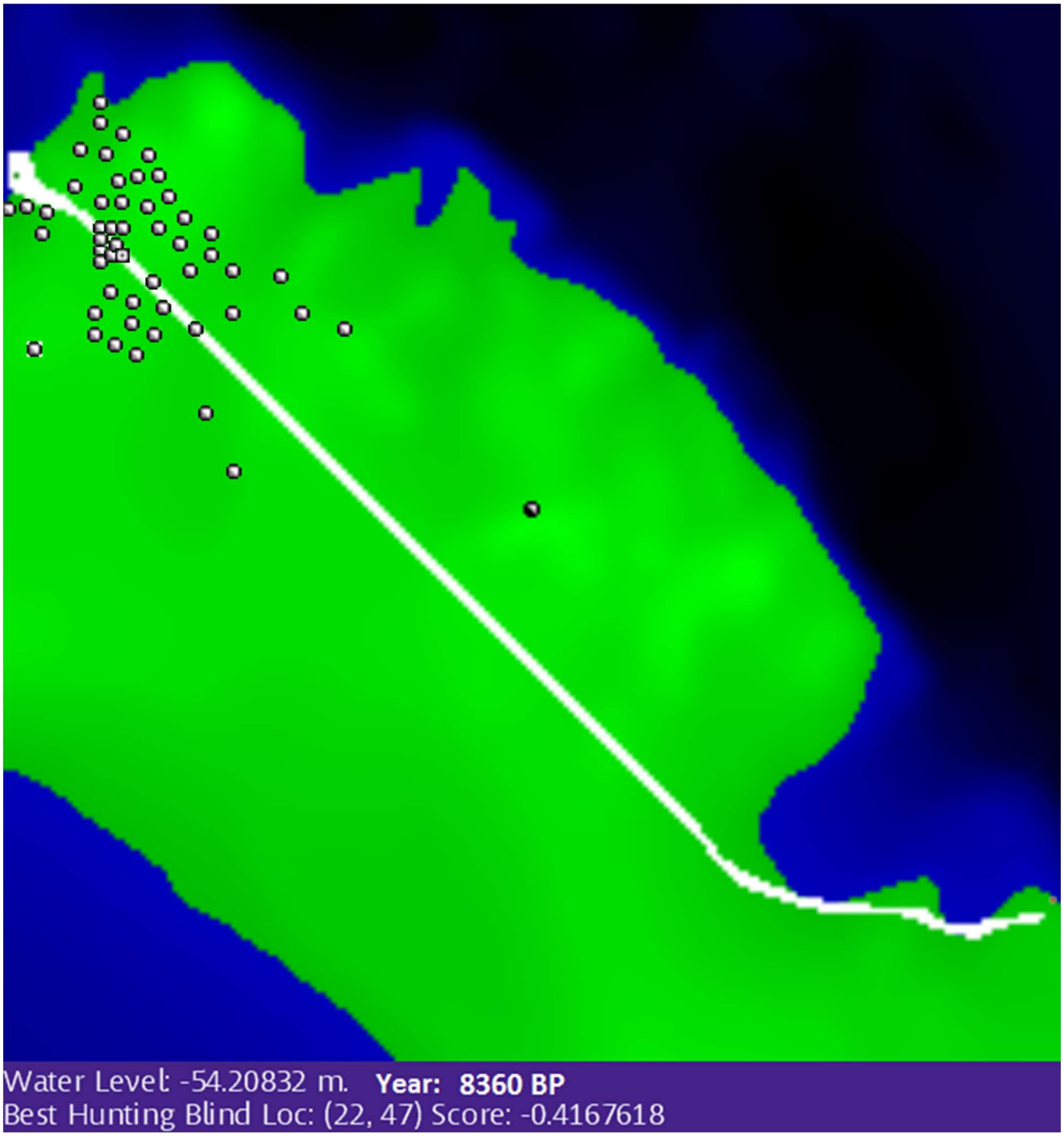


Figure 40: Projected Hunting Blinds in 8360 BP in [Stanley, 2013]

5.5 Learning Curve for Agent-Based CA

We now provide a "learning curve" graph in Figure 41 of the 10-generation moving average of the score of the highest-scoring hunting blind vs. the year.

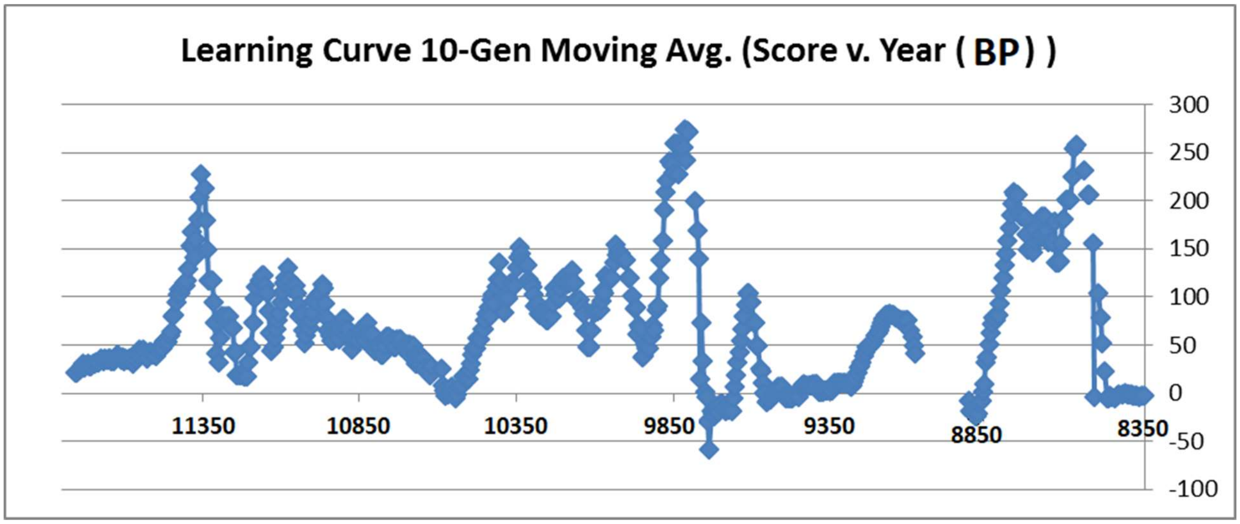
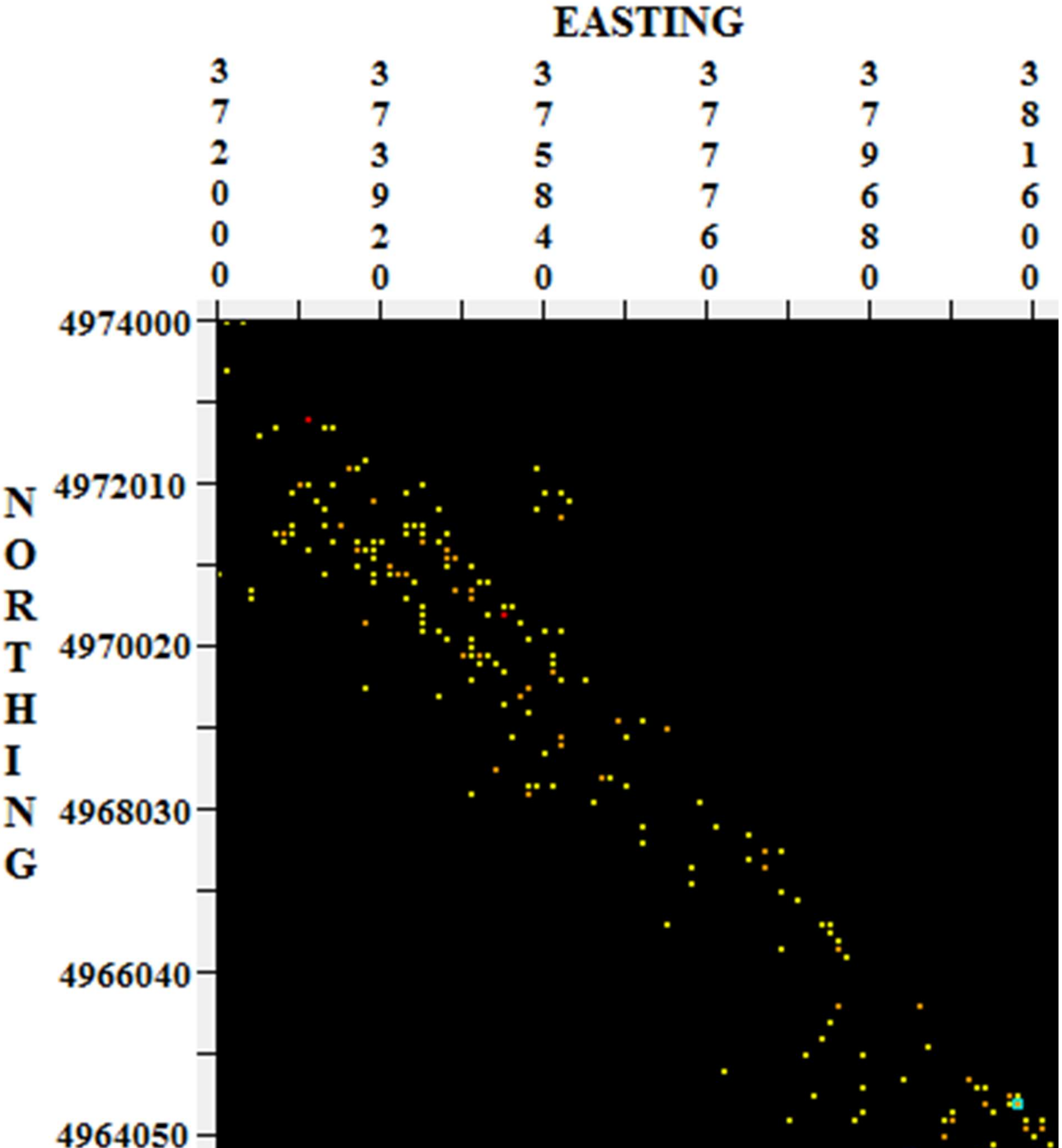


Figure 41: Learning Curve 10-Generation Moving Average in [Stanley, 2013]

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 befell the land bridge will force the agents to adjust their strategies, 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. Also, 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.

5.6 Heatmap

In order to fully demonstrate the results, we created another program which generates different kinds of heat maps, 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). The quadrant of this heatmap that is pertinent to the Funnel Drive Structure is shown in Figure 42. (For the full heatmap, please see [Stanley, 2014].)



Color	16-run avg # of generations that this square contained a hunting blind as a % of 690 gens = 3,450 years (i.e., the sim period).
	5%-10%
	3%-5%
	2%-3%
	Funnel Drive Structure Location

Figure 42: Artifact Heatmap in [Stanley, 2013]

5.7 David Warnke's Multipath Results

Around the time of the publication of the initial results in [Stanley, 2013], Dr. Robert Reynolds and David Warnke suggested the possibility of multiple entry points for the caribou and provided us with the following overlay in Figure 43.

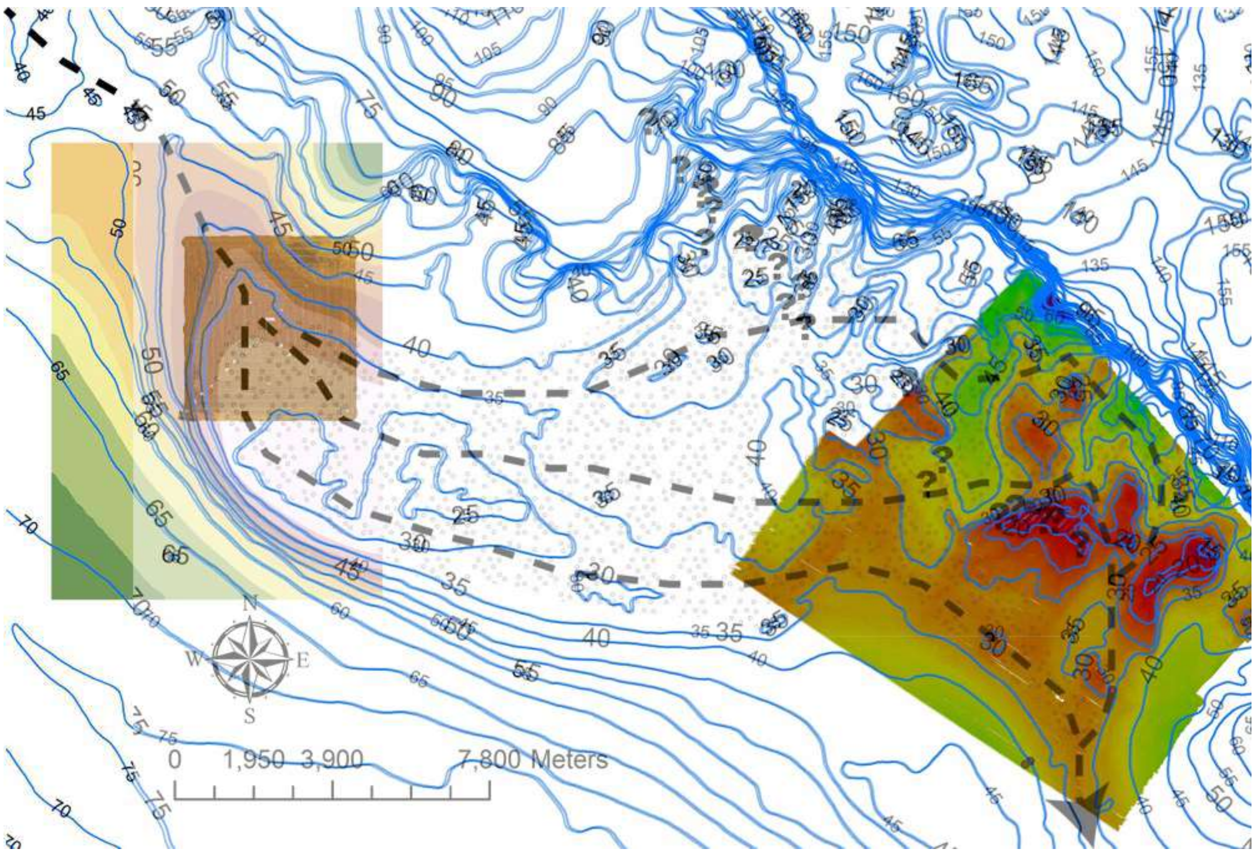


Figure 43: Team's 2013 Results vs. Multipath Scenario Caribou Projections

In Figure 43 above, the dotted lines indicate conjectured caribou paths and the ? symbols denote the 20 most highly predicted artifact locations as published in [Stanley, 2013]. Reynolds and Warnke devised a new experiment with the caribou entering Area 1 from multiple locations before converging into a single path and leaving to the south. This experiment covered the years from 11405 BP to 11244 BP with a timestep of 1 generation per year. Figure 44 contains a screenshot from this experiment which was included in [Stanley, 2014]. The black dots designate the hunting blind predictions while the white line is the caribou path.

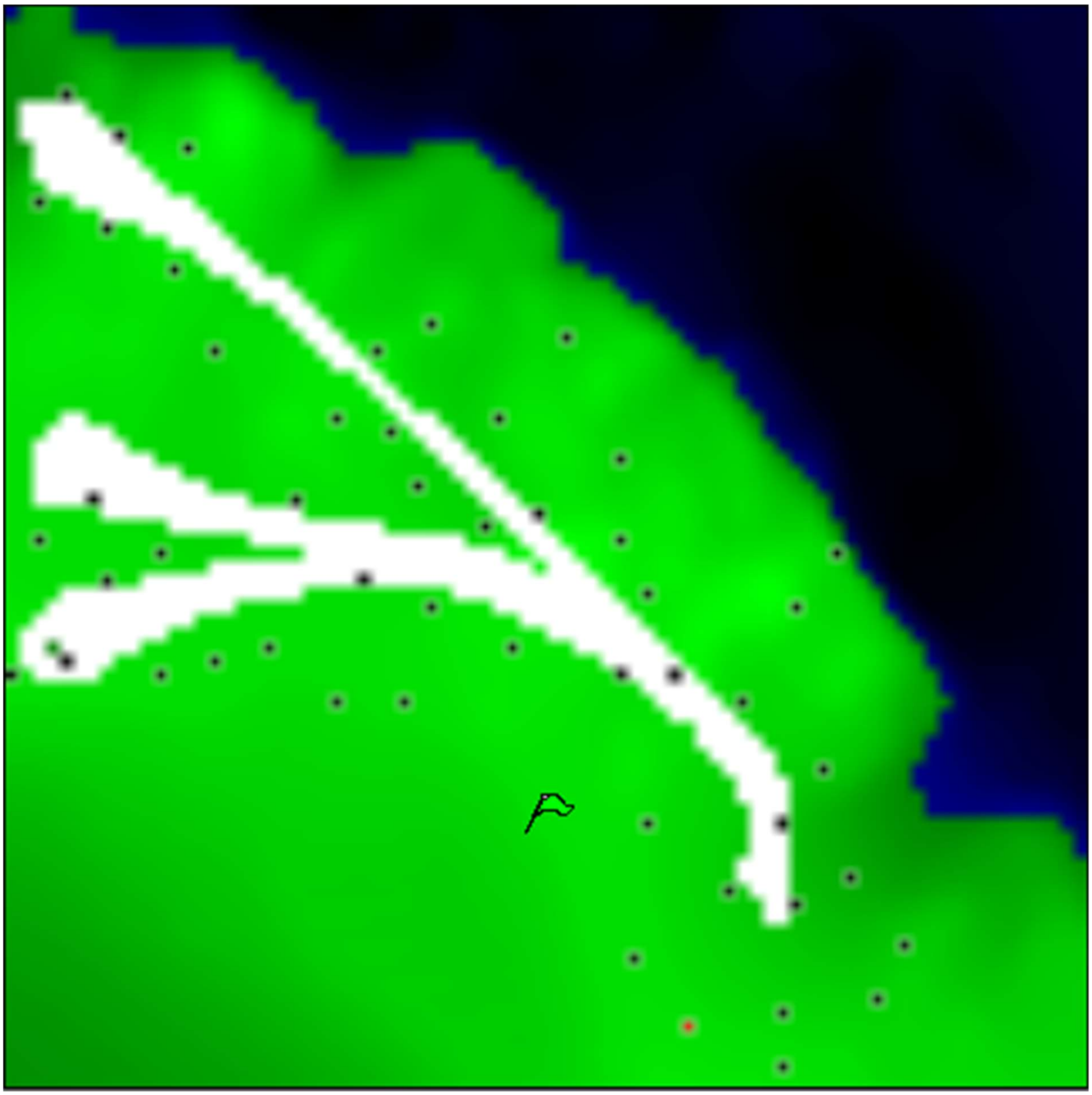
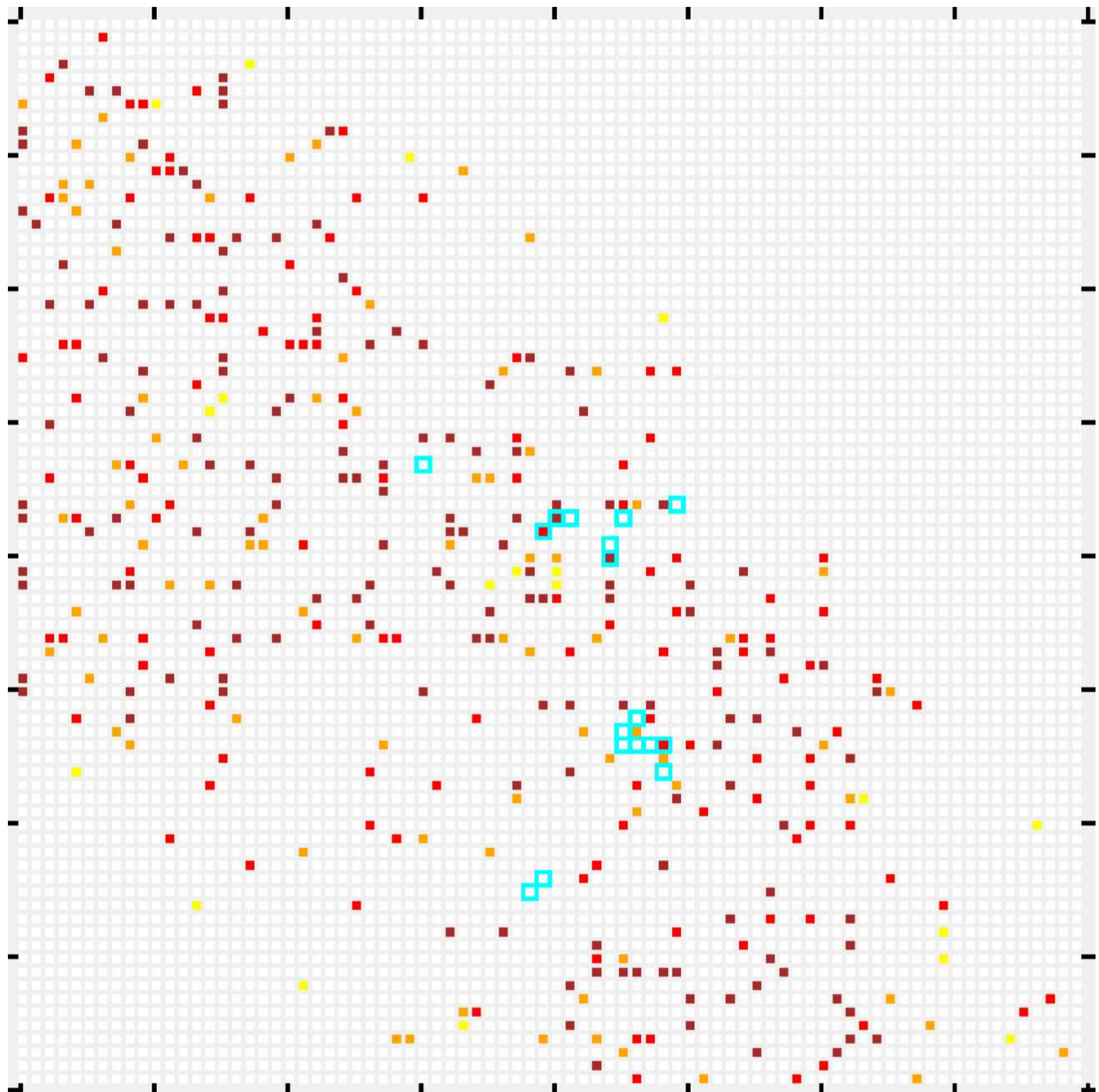


Figure 44: Screenshot from David Warnke's Multipath Experiment [Stanley, 2014]

5.7.1 80x80 map

Figure 45 below, is a plot of the results of Warnke's experiments against Dr. O'Shea's Area 1 finds as of April 2014 as reported in [Stanley, 2014].



Color	16-run avg # of generations that this square contained a hunting blind as a % of 690 gens = 3,450 years (i.e., the sim period).
	> 10%
	5%-10%
	3%-5%
	2%-3%
	Location of at least one artifact found by Dr. O'Shea

Figure 45: Predictions from Warnke's Multipath Experiment vs. Actual Finds 80 x 80 Map

5.7.2 40x40 map

We also investigated Figure 45's data on a 40x40 grid, shown in Figure 46 below [Stanley, 2014].

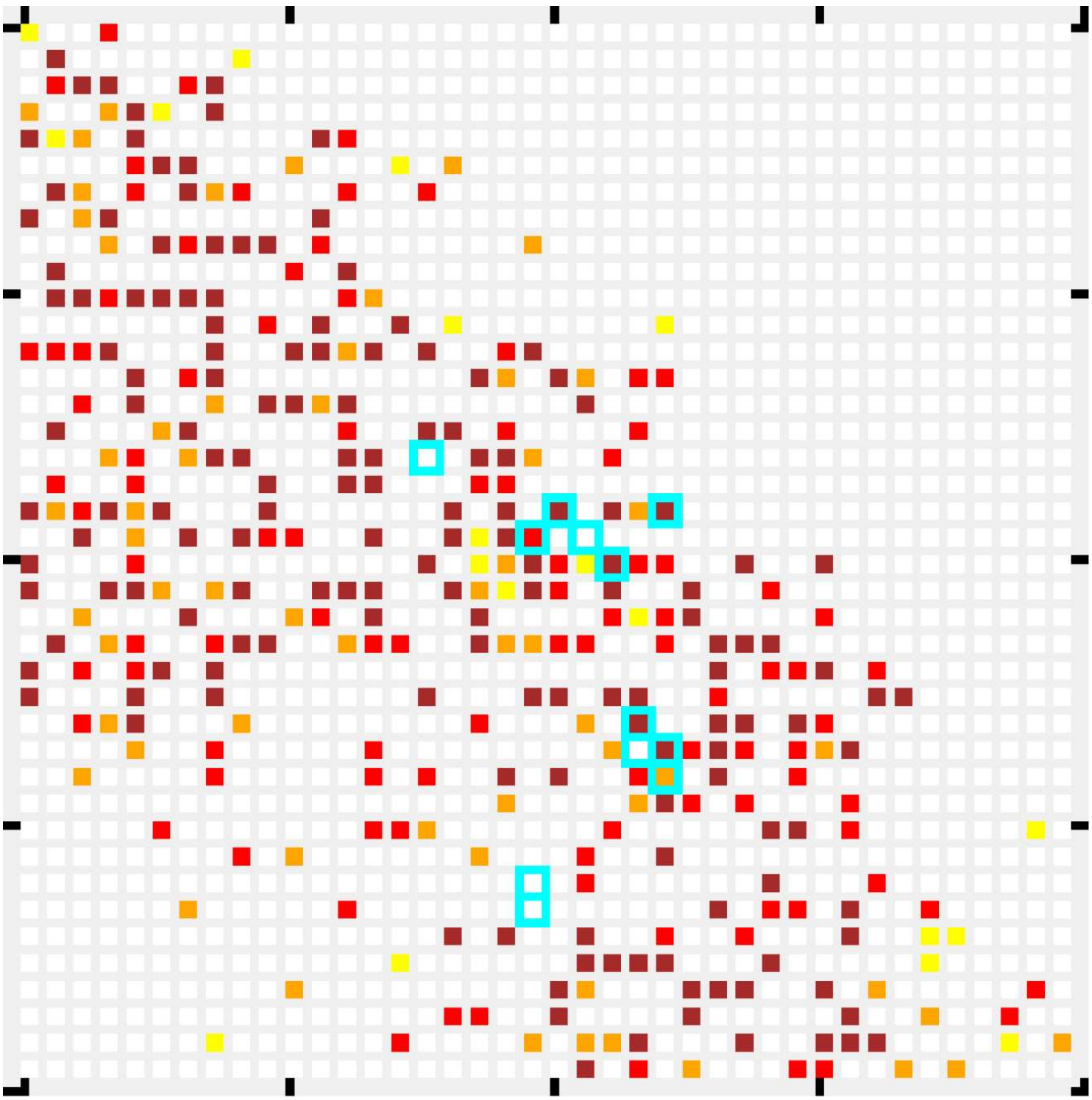
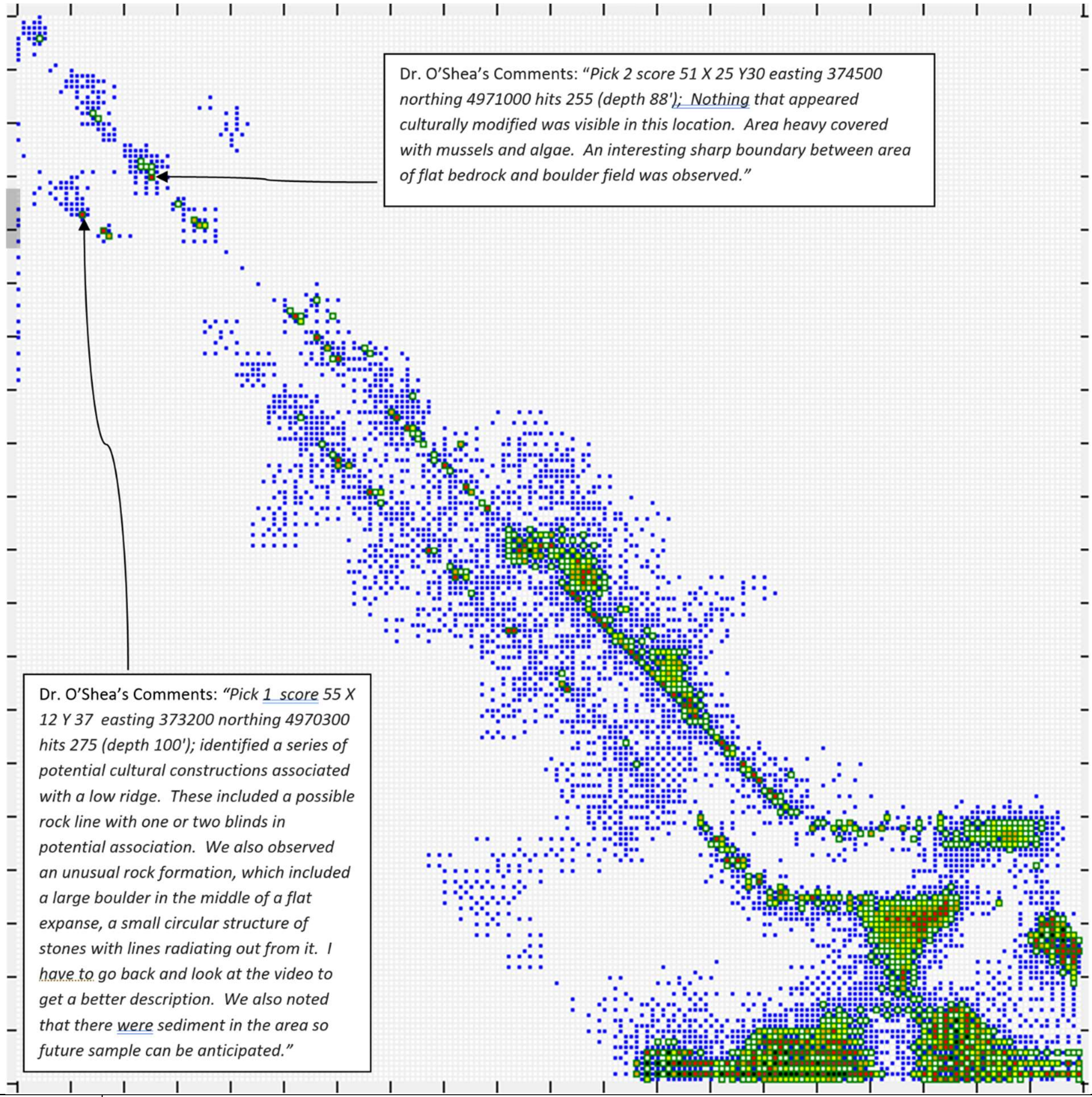


Figure 46: Predictions from Warnke's Multipath Experiment vs. Actual Finds 40 x 40 Map

5.8 Results from Fall 2016



Color	No. of gens that the square contained an Obs. Stand as a % of the time that Land Bridge was crossable (i.e, 690 gens; 3,450 yrs)
Black	>20%
Brown	10-20%
Red	5-10%
Orange	3-5%
Yellow	2-3%
Light Yellow	1-2%
Blue	<1%, but an AI Observation Stand occupied this square at least once during the simulation.

Figure 47: Observation Stand Heatmap

In Fall 2016, we produced the heatmap in Figure 47 for Dr. O'Shea for the Observation Stand structure type. As part of his 2016 expeditionary season, he went out to two highly-predicted locations designated on our heatmap. These were 375000E, 4971000N and 373200E, 4970300N (UTM-17). In the latter location, Dr. O'Shea found a potential Observation Stand in association with a potential Drive Line and Hunting Blind.

5.9 Lingering Issue

The main issue that continually haunted the agent-based approach was the inability to find "the perfect balance" between occupational structures predicted and locations flagged. It was possible to tweak the agent-based approach to flag less locations, by for instance, decreasing the number of agents or increasing the threshold needed to flag a location. However, getting the system to flag less locations invariably resulted in less structure predictions, since structures are inside of locations. Conversely, there were also ways to make the agent-based approach predict more structures, by for instance, increasing the number of agents or decreasing the threshold needed to flag a location. However, doing any of these things resulted of course in an increased number of flagged locations. For some time, the "way forward" seemed to be to change from an agent-based system to a rule-based expert system, and then to go through the Anthropological Archaeology literature to find "the perfect ruleset" in order to generate "the perfect balance" of locations flagged and structures predicted.

5.9.1 Change to Rule-Based Expert System and Search for the "Perfect Ruleset"

In 2017, the agent-based system was indeed "mothballed" and the change to a rule-based expert system (fully described in Chapter 6) was indeed made. However, the search for "the perfect ruleset" for describing hunter-gatherer settlement systems on the Alpena-Amberley Land Bridge proved just as fruitless. This is because there is a fundamental problem with reconstructing Early Holocene settlement systems by comparing them with modern-day Inuit settlement systems which cannot entirely be gotten around. The core problem is that by the time that anthropological and archaeological scholars reached the lands of the Inuit in the North American Arctic, their way of life had irrevocably changed from that of the Paleoindians. Even by the 1800s, factors such as guns, dogs, and modern life in general had irretrievably changed the nature of the hunt, and thus the settlement systems that revolve around it. This is probably why anthropological-archaeology experts have never made any attempt to come up with quantitative rules such as "Caches should be no more than 0.5km from the fall caribou trail" or "Hunting Blinds should be no more than 4km from a logistical camp" to try to describe life in the Early Holocene. This has to be the reason why said experts have always limited themselves to stating their rules qualitatively, such as "Campsites are typically located in a high-vegetation area to use plants for fuel for fires." or "Cache sites are likely to occur where any chance at hunting is located near a campsite if they are closer to hunting opportunity than they are to the village" [Binford, 1978b]. These qualitative rules are still very likely to be correct. However, it has become abundantly clear that there is no way to engineer "the perfect quantitative rules" out of them. And since in a rule-based expert system, what locations are flagged (and what structures are predicted) depends upon quantitative rule thresholds, the idea of "the perfect balance" between locations flagged and structures predicted must likewise be abandoned.

5.9.2 Change in Perspective

When we abandoned the ideas of "the perfect ruleset" and "the perfect balance", we reimagined the entire problem as an economic cost vs. benefit problem, specifically according to Pareto economic theory. Vilfredo Pareto originally became an economist in the 1880's; when he originally became an economist, nearly all other economic theories of value were intrinsic (i.e., "objective") theories. The vast majority of these were "labor theories" of value that stated in one form or another that the value of a good was proportionate to the labor applied into its production. According to intrinsic theories of economic value, it is possible to objectively calculate the value of each good and thus rank goods via a single objective according to supposed intrinsic values.

Pareto was among the earliest economists to reject conventional intrinsic (i.e., "objective") theories of economic value. He called for the replacement in economics of the notion of "objective optimality" with "Pareto-optimality". In Pareto Theory, a "Pareto-optimal" solution is a solution that is not dominated by any other. "Dominance" in Pareto Theory can be defined in the following way [Best, 2009]:

For an m-objective minimization problem, a solution x_1 dominates x_2 if $\forall i = 1, \ldots, m$, $f_i(x_1) \le f_i(x_2)$ and $\exists\, i \in \{1, \ldots, m\} \mid f_i(x_1) < f_i(x_2)$.

In Pareto Theory, dominated solutions are the only ones that are considered sub-optimal. For any given multi-objective problem, once all sub-optimal solutions have been removed, what is left over is a Pareto-optimal set of solutions, sometimes known as a "Pareto Front", which constitutes the "final result" for the problem. Pareto Theory does not outright reject judgments about what items within a Pareto Front are better than others, but it does say that such judgments are ultimately subjective.

5.10 Conclusion

After considering the problems with our agent-based approach, we decided to change our approach to a rule-based expert system approach, described in Chapter 6. However, we were still confronted with the same problem of being unable to find "the perfect balance" between locations flagged and structures predicted. We decided to abandon the idea of "the perfect balance" and to reformulate the problem as an economic cost vs. benefit problem according to Pareto Theory. In Chapter 7, we discuss our use of a Pareto-based multi-objective optimization system in order to winnow out sub-optimal solutions, producing for each occupational structure category a Pareto Front containing only Pareto-optimal locations vs. structures pairs, each with a corresponding evolved ruleset.

CHAPTER 6: RULE-BASED EXPERT SYSTEM SPECIFICATION

6.1 Objectives

In any rule-based expert system that is designed to output lists of suggested locations in the real world to prospect for where desired items might be found, the system must strive to minimize the effort spent prospecting out in the field while maximizing the payout gained through said prospecting. Thus, there are two countervailing objectives in producing location lists: The number of the desired items contained within locations in the list should be maximized, and the overall number of locations in the search list is minimized.

We are using Dr. O'Shea's latest set of discovered artifacts, provided to our team in April 2018, as a training set. Given the specification of the environment, the training set, and the general forms of the rules, what the system must do is discover the Pareto-optimal set of (number of locations flagged, number of training set artifacts in those locations) ordered pairs. In doing this task, the system is forced to evolve what is effectively a Pareto-optimal set of rulesets from the general forms of the rules provided by the archaeologists and the archaeological literature with each ruleset corresponding to a point within the aforementioned Pareto-optimal set of ordered pairs. These rulesets, and the lists of locations to be prospected that are produced by each of them, can only be improved in one of three ways: Obtaining better data about the prehistoric environment from the geologists and the geological literature, obtaining a better training set from the archaeologists, or obtaining more and/or better general forms of rules from the archaeologists and the archaeological literature.

6.2 Component Diagram

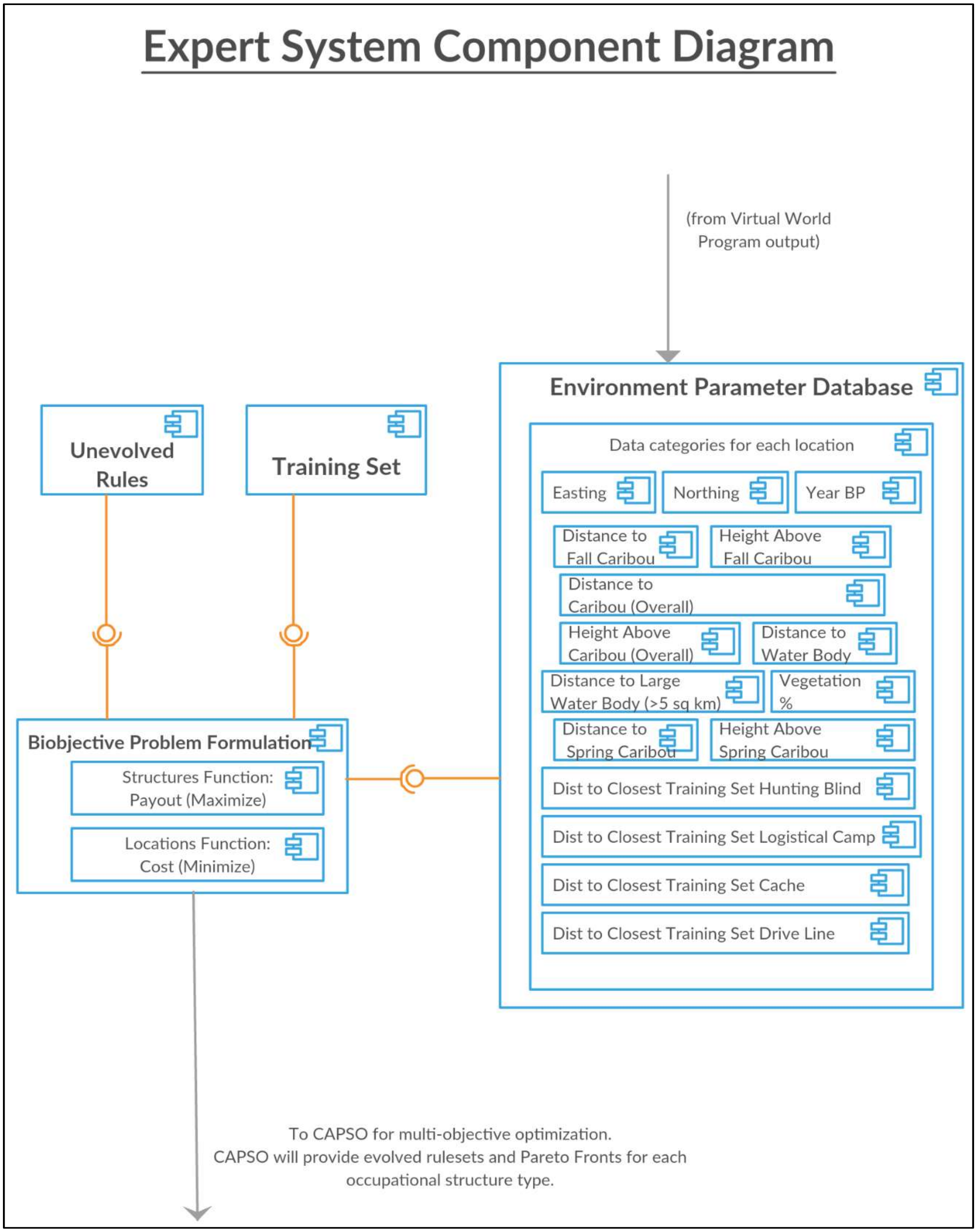


Figure 48: Expert System Component Diagram

6.3 Experimental Framework

For the rule-based expert system approach, Dr. O'Shea asked us to work with a 5km by 5km "jumbo region" comprising of Regions 6, 7, 8, and 9 (whose locations are shown in Figure 27). We divide the "jumbo region" into 40,000 individual locations which are 250m x 250m each in size. The system we are about to discuss works with our time engine which we discussed in Chapter 4. As explained in Chapter 4, we are investigating the time period of 11800 BP to 8400 BP (inclusive). We are dividing this time period up into 18 intervals of 200 years each.

6.4 Environment Specification

An environment model **E** is a set of elements h such that each h ∈ **E** is a latitude, longitude, pair denoting a location somewhere in the environment. The term h_i can be used to denote location h at time i. For all $h_i \in \boldsymbol{E}$, there exists P_{h_i}, which is a set of environment parameters at location h at time i. Also, for all $h \in \boldsymbol{E}$, there exists q_h , which is a variable equal to the number of actual artifacts of a given type if the archaeologists have found any at location *h*, or equal to -1 if the archaeologists have looked in location *h* and found nothing (false positive), or 0 in any other case.

6.5 Prediction Model Specification

A hypothetical prediction model, **M**, is an ordered pair (T, **R**) where T is the prediction threshold and **R** is a set of rules such that $\boldsymbol{R}$ = {$\boldsymbol{R_m}$, $\boldsymbol{R_s}$}, where $\boldsymbol{R_m}$ is the set of musthave rules within **R**, and $\boldsymbol{R_s}$ is the set of standard rules within R. These sets can further be divided into their individual constituent rules such that $\boldsymbol{R_m} = \{\boldsymbol{r_{m_1}}, \boldsymbol{r_{m_2}}, \boldsymbol{r_{m_3}}, \ldots\}$ and $\boldsymbol{R_s} = \{\boldsymbol{r_{s_1}}, \boldsymbol{r_{s_2}}, \boldsymbol{r_{s_3}}, \ldots\}$**.** Each rule $\boldsymbol{r}_j$ in **R**, regardless of whether it is standard or musthave, can be defined as a function that can be evaluated at node h_i containing parameters P_{h_i}.

$\boldsymbol{r}_j(P_{h_i})$ returns 1 if P_{h_i} satisfies $\boldsymbol{r}_i$, and 0 otherwise. **R** itself can be evaluated at each h_i such that

$$\boldsymbol{R}(h_i) = \prod_{k=1}^{u}[r_{m_k}(P_{h_i})] \cdot \sum_{n=1}^{v}[\mathrm{r}_{s_n}(P_{h_i})]$$

Equation 3: Evaluation of Ruleset R

where u is the number of musthave rules, v is the number of standard rules.

We define *t* as the *timestep threshold*, the threshold at which for location *h* at time *i*, if $\boldsymbol{R}(h_i) \geq t$, then one point is added to the prediction score for location *h*. We then define *T* as the *prediction threshold*. For all $i \in \Lambda$ where Λ is the set of time periods being investigated, for a given $h \in \boldsymbol{E}$, if $\sum_{i \in \Lambda}[\boldsymbol{R}(h_i) \geq t] \geq T$, then *h* is considered to have been predicted in **M**. The quantity q_h is denotes the total number of structures of the relevant type in the training set at location *h*. The quantity q_h can be thought of as location *h*'s individual "payout" granted for predicting it. Prediction model **M** can itself be evaluated over **E** in terms of the total number of predictions made such that

$$\boldsymbol{M}(\boldsymbol{E}) = \sum_{h \in \boldsymbol{E}} q_h \cdot \left\{ \left\{ \sum_{i \in \Lambda}[\boldsymbol{R}(h_i) \geq t] \right\} \geq T \right\}$$

Equation 4: No. of Relevant Structures Successfully Predicted by Model M (Total Payout: Maximize)

In words, **M**(**E**) gives the number of structures that model **M** has successfully predicted within environment **E**. Obviously, we want to maximize **M**(**E**), since all other things being equal it is better for the model to successfully find more structures than less structures. At the same time, however, we want to minimize the total number of locations that are predicted by model **M**, since all other things being equal it is better for the archaeologists using the model to have to visit fewer locations than more. This provides our second objective, which can be expressed as L(**M**(**E**)), where

$$L(\boldsymbol{M}(\boldsymbol{E})) = \sum_{h \in \boldsymbol{E}} \left\{ \left\{ \sum_{i \in \Lambda} [\boldsymbol{R}(h_i) \geq t] \right\} \geq T \right\}$$

Equation 5: No. of Locations Predicted by Model M (Total Cost: Minimize)

Again, L(**M**(**E**)) is the total number of locations predicted by a model M, and it should be minimized.

Optimizing **M**(**E**) and L(**M**(**E**)) as bi-objective functions will produce a Pareto Front which we can plot out as **M**(**E**) vs. L(**M**(**E**)). Each point on this Pareto Front will correspond to a certain prediction model **M** with a model score **M**(**E**) and a total number of predicted locations L(**M**(**E**)). (Ultimately, our system receives q_h for each h from the work that has already been done by the archaeologists and **P** from the environment engine. The process of bi-objective optimization implicitly generates a ruleset **R** and a prediction threshold t for each model **M** in the Pareto Front.)

6.6 Rule Parameter Design

Table 1 contains the categories for the parameters that we collected from the Virtual World system. These are the parameters that are going to go into the rules detailed later in this chapter. Each occupational structure category will be predicted based upon a subset of these rules.

Name	Variable Name	Description
Distance to Fall Caribou Trail	distToFallCaribou	The distance from the closest approach point of the fall caribou trail to this location.
Height Above Fall Caribou Trail	heightAboveFallCaribou	This location's height above (or below) the closest approach point of the fall caribou trail.
Distance to Spring Caribou Trail	distToSprCaribou	The distance from the closest approach point of the spring caribou trail to this location.
Height Above Spring Caribou Trail	heightAboveSprCaribou	This location's height above (or below) the closest approach point of the spring caribou trail.
Distance to Caribou Trail	distToCaribou	The distance from this location to the closest approach point of the caribou trail at during either season in the given year.
Height Above Caribou Trail	heightAboveCaribou	The height of this location above (or below) the closest approach point of the caribou trail at during either season in the given year.
Distance To Water	distanceToWater	The distance from this location to any water body.
Distance To Large Water Body	distToLargeWaterBody	The distance from this location to a large water body. Only the two large lakes that sandwich the Alpena-Amberley Land Bridge and any fjords and rivers that are connected to them are considered "large". All other water bodies are considered "small".
Vegetation Level	vegetationLevel	The amount, as a percentage, of vegetation in the square. 0.0 means that the square is completely bare of vegetation, while 1.0 means that the square is completely covered with vegetation.
Distance To Closest Training Set Logistical Camp	distToTSetLogCamp	Distance to closest actually-found logistical camp within the training set.
Distance To Closest Training Set Hunting Blind	distToTSetHuntingBlind	Distance to closest actually-found hunting blind within the training set.
Distance To Closest Training Set Drive Line	distToTSetDriveLine	Distance to closest actually-found drive line within the training set.
Distance To Closest Training Set Cache	distToTSetCache	Distance to closest actually-found cache within the training set.

Table 1: Rule Parameters

6.7 Rule Design

The specification forms of rules provided have all have the same overall form, which is:

$$g\big(Y, f(X)\big) = [f(X) \sim Y]$$

Equation 6: Overall Form of Rules

where X is the set of environmental variable arguments that are being tested, Y is a threshold, f is some function that acts upon the environmental variable arguments, and ~ is some relation between f(X) and Y. If the rule fires, it returns a 1, otherwise it returns a 0.

Most of the time, Y is treated as a mutable variable that the optimizer system is able to alter in its task of trying to optimize the bi-objectives listed in Equation 4 and Equation 5. However, in a few situations where the rule would not make sense otherwise, Y is a fixed variable which is always equal to 0. (An example is the "Don't Be Underwater Rule", which is formulated as "distToWater > 0".)

All of the specification forms of rules listed in the tables on the next pages come either from the anthropological-archaeological literature or discussions with Drs. John O'Shea and Ashley Lemke. References to specific pieces of literature within anthropological-archaeology are provided when available.

6.7.1 Logistical Camp Rules

Rule Name	Rule Type Rule Specificaton	Rule Description Commentary (if any)
dontBeUnderwaterRule (“Don’t Be Underwater Rule”)	Musthave distToWater > 0	Artifact cannot exist underwater (or on top of water).
distToFoodSourceRule	Musthave MIN(distToCaribou, distToLargeWaterBody) <= Y	This rule is fulfilled for a candidate site for a logistical camp if the site is within a certain distance of a caribou trail (for caribou meat) or major water source (for fish), whichever is lesser.
vegetationRule	Standard Vegetation% >= Y	This rule is fulfilled for a candidate site for a logistical camp if it contains a certain amount of vegetation, as this is desirable for firewood and protection from wind. “Camping within a willow stand is, during most periods of the year, desirable since there is ready firewood, protection from the wind, and, generally, water from springs.” [Binford, 1978, p. 256]
distToTSHuntBlindRule	Standard distToTSHuntingBlind <= Y	This rule is fulfilled for a candidate site for a logistical camp if it within a certain distance of a hunting blind within the training set, as the logistical camp could provide quarters for the people manning the hunting blind.
distToTSCacheRule	Standard distToTSCache <= Y	This rule is fulfilled for a candidate site for a logistical camp if it within a certain distance of a cache within the training set. Logistical camps built after hunting season is over can be used to house workers who do the work of preparing food from caches stored during the hunting season.

Table 2: Logistical Camp Rules

6.7.2 Hunting Blind Rules

Rule Name	Rule Type Rule Specificaton	Rule Description Commentary (if any)
dontBeUnderwaterRule (“Don’t Be Underwater Rule”)	Musthave distToWater >= Y	At the time that it was being used by prehistoric hunter-gatherers, this structure cannot have been located underwater.
lanceRule (musthave)	Musthave distToCaribou >= Y	A candidate site for a hunting blind must be within a certain distance of the caribou trail. *“As it seems clear that hunters employing lances must get relatively close to the animals in order to kill them, there must be some other mechanism or condition that enabled the lance-armed hunters to do so from the AAR hunting structures. The size of the hunting blinds would provide sufficient concealment to allow the hunters proximity to the animals without raising undue alarm.” [O’Shea, 2016]* *“The expectations for hunting architecture sites with atlatls is likely intermediate between the long range of arrows and the shorter range of lances, but the exact numbers cannot be certain as most known hunting architecture sites did not use this technology. Therefore, while atlatls cannot be ruled out, the current evidence is inconclusive.”* [Lemke, 2016]
levelWithCaribouRule	Standard |heightabovecaribou| <= Y	A candidate site for a hunting blind must be roughly level with the caribou trail.
vegetationRule	Standard vegetation% >= Y	This rule is fulfilled for a candidate site for a hunting blind if it contains a certain amount of vegetation, as this can be used to help build the blind.
campClosenessRule	Standard distToLogCamp <= Y	This rule is fulfilled for a candidate site for a hunting blind if it is within a certain distance of a residential or logistical camp.
obsStandClosenessRule	Standard distToObsStand <= Y	This rule is fulfilled for a hunting blind candidate site for if it is within a certain distance of an observation stand.

Table 3: Hunting Blind Rules

6.7.3 Drive Line Rules

Rule Name	Rule Type / Rule Specificaton	Rule Description / Commentary (if any)
dontBeUnderwaterRule (“Don’t Be Underwater Rule”)	Musthave distToWater > 0	At the time that it was being used by prehistoric hunter-gatherers, this structure cannot have been located underwater.
caribouClosenessRule “Caribou closeness rule”	Musthave distToCaribou <= Y	A candidate site for a drive line must be within a certain distance of the caribou trail.
distToTsetHBRule (“Training Set Hunting Blind Closeness Rule”)	Standard distToTsetHuntingBlind <= Y	A candidate site for a drive line receives a bonus if it is within a certain distance of a hunting blind within the training set.
distToTsetLCRule (“Training Set Logistical Camp Closeness Rule”)	Standard distToTsetLogCamp <= Y	A candidate site for a drive line receives a bonus if it is within a certain distance of a logistical camp within the training set.

Table 4: Drive Line Rules

6.7.4 Cache Rules

Rule Name	Rule Type / Rule Specificaton	Rule Description / Commentary (if any)
dontBeUnderwaterRule (“Don’t Be Underwater Rule”)	Musthave distToWater > 0	At the time that it was being used by prehistoric hunter-gatherers, this structure cannot have been located underwater.
huntingBlindClosenessRule	Standard	This rule is fulfilled for a candidate site for a cache if it is within a certain distance of a hunting blind that has been already found by the archaeologists.
	distToHuntingBlind <= Y	*“Caches are common components of a logistical strategy in that successful procurement of resources by relatively small groups for relatively large groups generally means large bulk. This bulk must be transported to consumers, although it may on occasion serve as the stimulus for repositioning the consumers. In either case there is commonly a temporary storage phase. Such "field" storage is frequently done in regular facilities, but special facilities may be constructed to deal specifically with the bulk obtained.”* *“On occasion kills (locations) may be made directly from a hunting stand, and the meat may be processed and temporarily cached there.”* [Binford, 1980]
fallcaribouPathClosenessRule	Standard distToFallCaribouPath<=Y	This rule is fulfilled for a candidate site for a cache if it is within a certain distance of the fall caribou path.
TSLogCampClosenessRule	Standard distToTSLogCamp <= Y	This rule is fulfilled for a candidate site for a cache if it is within a certain distance of a logistical camp in the training set.

Table 5: Cache Rules

6.8 Conclusion

We now need to optimize the various rulesets that we have laid out for our expert system, however our expert system does not have native optimization. It must rely upon an outside multi-objective optimizer to provide optimization services for it. For this purpose, we have created the CAPSO (Cultural Algorithm Particle Swarm Optimizer) system, which we explain in detail in the next chapter.

CHAPTER 7: MULTI-OBJECTIVE OPTIMIZATION SYSTEM AND CULTURAL ALGORITHMS

7.1 Overview

Our system that we developed to solve the problem specified in the previous chapter is a parallelized multi-objective optimizer that combines elements from Cultural Algorithms (CA), Particle Swarm Optimization (PSO) [Eberhart, 1995], and Vector-Evaluated Genetic Algorithms (VEGA) [Schaffer, 1985]. We have named our optimizer system “CAPSO”, which is short for Cultural Algorithm/Particle Swarm Optimizer.

7.2 Multi-Objective Optimization

Typically, a multi-objective problem is specified with three components: The set of functions to be optimized, the set of constraint functions, and the parameters along with parameter ranges. (In a multi-objective problem, “optimizing” the objective functions might mean minimizing all of them, maximizing all of them, or minimizing some and maximizing others.) A general formulation of a multi-objective problem can be written as such:

Let F: $\{f_1, f_2, f_3, \ldots, f_n\}$ be the set of objective functions.

Let G: $\{g_1, g_2, g_3, \ldots, g_m\}$ be the set of constraint functions.

Let $\vec{x} = < x_1, x_2, x_3, \ldots, x_k >$ be the vector containing the parameters.

Let $[r_{i_1}, r_{i_2}]$ be the range for each parameter x_i.

7.3 Cultural Algorithm Background

7.3.1 History

Cultural Algorithms (CA's) were originally devised by Dr. Robert Reynolds in the 1970s [Reynolds 1978, 1979]. In creating CA's, Dr. Reynolds drew an analogy between group learning and the tendency of group knowledge acquired in the past to influence current decisions by individual members of groups [Reynolds, 1994].

Reynolds was originally motivated to invent Cultural Algorithms when he was working on a research project in the 1970s concerning a Genetic Algorithm (GA). During this research work, Reynolds wasn't sure how much that the GA was actually learning. His solution was to create a "scorecard" for the GA in order to formally keep track of the knowledge that it was uncovering. Eventually, Reynolds realized that his "scorecard" functioned as a social "memory" for the GA population, and that it could not only receive knowledge from the GA, it could provide knowledge to the GA in order to guide its progress. Eventually, Reynolds called this shared social memory the *belief space* and invented the name *cultural algorithms*. Reynolds and his fellow CA researchers realized that this "scorecard", which he eventually termed the "belief space", could be attached just as well to other algorithms besides GAs (for instance, PSO algorithms), and could collect from and provide knowledge to them in just the same manner. Hence today the name *cultural algorithm* has been expanded to *any* algorithm or population based framework such as agents that uses a belief space and contains a communication protocol and dual inheritance between the population and belief spaces.

7.3.2 Structure of a Typical Cultural Algorithm

Formally, cultural algorithms contain a *population space* which is influenced by a *belief space* via a *communication protocol*. The 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 or domain 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 Figure 49:

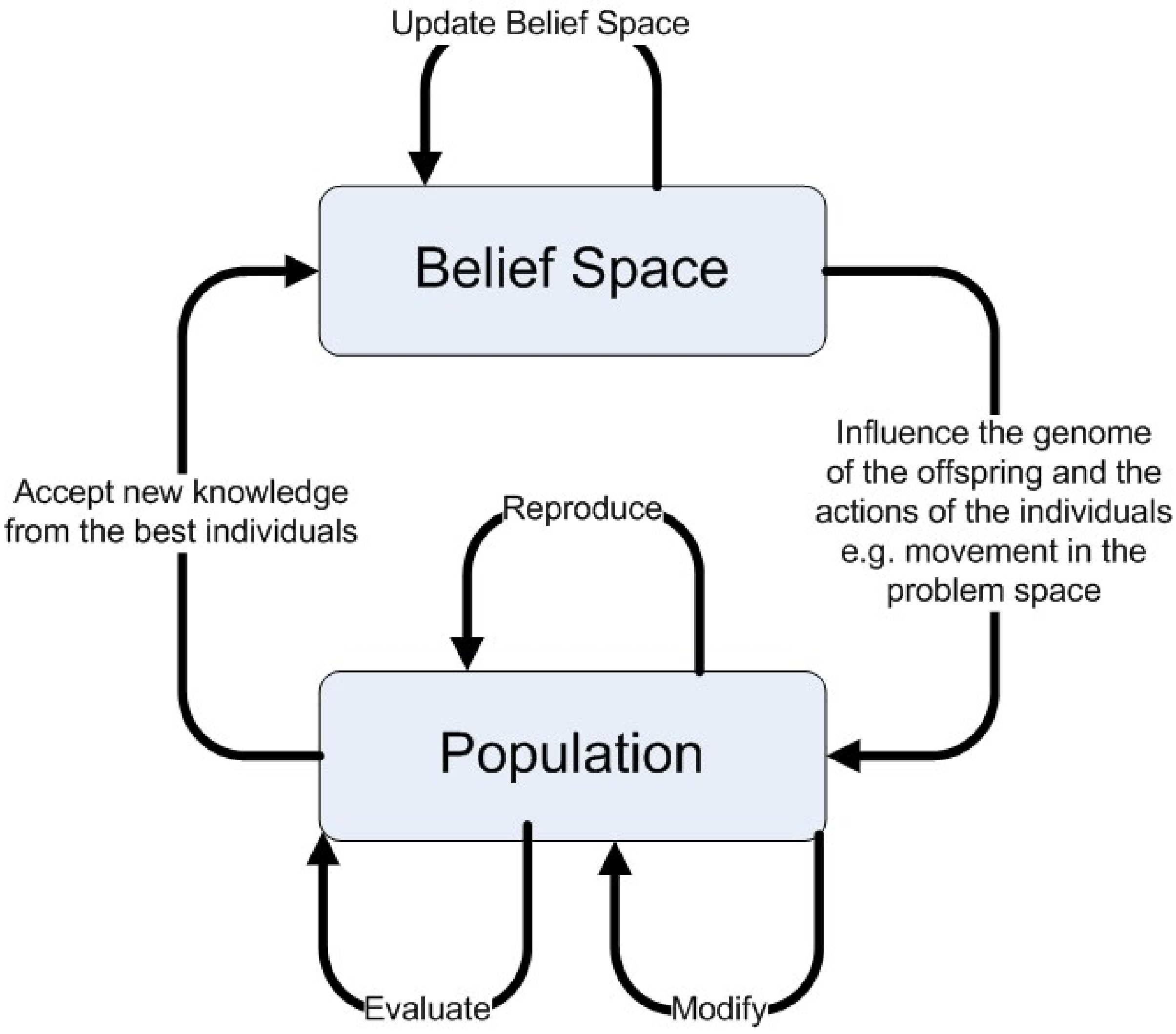


Figure 49: Schemata of Cultural Algorithms [Jin, 2011]

7.3.3 Acceptance Step

Depending upon the individual cultural algorithm being used, either all individuals in the population space will be allowed to influence the belief space, or only some. Oftentimes, the acceptance function is specified in terms of a percentage. An example might be "The best 10% of individuals according to the fitness function will be allowed to influence the belief space." [Reynolds, 2017]. In CAPSO, the top 1/7 of scorers for each objective are allowed to influence the belief space.

7.3.4 Belief Space Update Step

In the update function, the knowledge received through the acceptance function is encoded into the belief space. Also during this step, knowledge that is obsolete or otherwise no longer relevant can be discarded from the belief space. One way of doing this is through a function that uses certain criteria to identify obsolete knowledge and remove it from the belief space [Stanley, 2013, 2014]. Another way is by having a competition during the update step in which the new knowledge that was just received and the preexisting knowledge already in the belief space can be made to vie against each other. The different beliefs can be evaluated against each other through a "belief fitness function" or through some kind of game mechanism. A certain percentage of the beliefs that perform worst according to the evaluation mechanism used can then be culled from the belief space.

7.3.5 Influence Step

In the influence step, the different belief space knowledge sources cooperate and/or compete in order to influence each agent within the population space. Some of the different methods that have been used are simple random selection [Peng, 2005], a weighted roulette mechanism, an auction mechanism [Reynolds, 2013], or a complex game [Reynolds, 2018].

When a population agent calls the influence function, an influencer knowledge source is selected through a mechanism such as those described above, an individual in the belief space corresponding to that knowledge source is selected or randomly generated, and the population agent's values are "pulled towards" those of the individual within the belief space.

7.3.6 CA Belief Space Knowledge Source Types

Generally, researchers who use CAs divide knowledge into five different types: Normative knowledge, domain knowledge, topographical knowledge, historical (or temporal) knowledge, and situational (or exemplar) knowledge [Best, 2010]. In some CA implementations, different knowledge types compete against one another for the opportunity to influence individual agents [Reynolds, 2006]. In other implementations, the different knowledge types are cooperative and participate collectively in influencing the agents. We now describe each knowledge type in more detail:

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 [Best, 2009]. It can be thought of as general "behavioral boundaries" within which individual behavioral adjustments can be made [Reynolds, 1997].

Normative knowledge in CAPSO works as thus: CAPSO's Normative Knowledge container contains a range for each of the parameters in the problems. When Normative Knowledge is selected, a velocity is randomly generated from within the ranges contained in the Normative Knowledge source. Then, as for the individual who called the influence function, its velocity is changed to a randomly-weighted average between its old velocity and the generated one.

During the update step, for each population agent that was given permission during the Acceptance Step to influence the belief space, a simple average is taken between the population agent's velocity within each dimension and the nearest edge of the Normative Knowledge interval for that dimension, and that edge of the Normative Knowledge interval for that dimension is changed to the result of this simple average.

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.

Historical knowledge in CAPSO works as thus: CAPSO's Historical Knowledge container has a number of velocities that have adjoined elite particles in the past. Each of these historical velocities also contains the latest time in the past in which it was accepted or re-accepted into Historical Knowledge. During the influence step, when Historical Knowledge is selected as a knowledge source, CAPSO randomly selects one out of all the velocities in the Historical Knowledge container. Then, as for the individual who called the influence function, its velocity is changed to a randomly-weighted average between its old velocity and the chosen velocity.
During the Update Step, the entire Historical Knowledge container is checked and if any historical velocity has not been accepted or re-accepted in over 500 generations, it is removed from the Historical Knowledge container.

Situational Knowledge

Situational knowledge concerns positive and negative exemplars which agents can use to guide their behavior [Reynolds, 1997]. 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 some CAs, situational knowledge can also include negative exemplars. In these CA's, solutions that score low are considered negative exemplars, and the CA can take them into account and steer clear of similar solutions, so as to avoid wasting time with them.

Situational knowledge in CAPSO works as thus: At the beginning of the program, CAPSO generates a number of initial guesses (exemplars) and assigns a selection probability to each of them. Each of these initial guesses can be thought of as a vector-point in hyperdimensional space. When Situational Knowledge is chosen as a knowledge source, CAPSO chooses one of these exemplars and produces a randomly weighted average between the exemplar and the velocity of the individual that called the influence function. The individual that called the influence function then has its velocity changed to this weighted average.

During the update step, CAPSO checks if any accepted individual's velocity is sufficiently close (i.e., within 1%) to an exemplar velocity within the Situational Knowledge container. If so, the chance that this exemplar will be chosen out of the situational knowledge container in the future is incremented by 1%, and the exemplar itself is changed to a randomly-weighted average between its old value and the velocity of the aforementioned accepted individual.

For example, if a particle whose velocity is <1, 2, 5> calls Situational Knowlege, and the Situational Knowledge Source chooses an exemplar velocity of <8, 9, 4>, and the random weight chosen is 0.3, then the new velocity for the particle will be 0.3·<1, 2, 5> + 0.7·<8, 9, 4> = <5.9, 6.9, 4.3>.

Domain Knowledge

Domain Knowledge concerns the overall shape of the search space itself [Best, 2009]. The purpose of Domain Knowledge is to deduce the shape of the search space and to explore the search space's margins. Because optimal values are often found out on the margins of the search space, Domain Knowledge is great as a "finalization mechanism" in an optimizer system.

In CAPSO, the Domain Knowledge container contains points believed to be on the boundary of the search space. If a particle selects Domain Knowledge as its knowledge source, a point is selected from the Domain Knowledge container, and a target velocity is generated equal to the vector difference between the location of the point on the boundary and the current location of the particle. Then, the particle's velocity is changed to a randomly-weighted average between its old velocity and the target velocity.

During the Acceptance Step, for each solution set newly accepted into the Pareto Front, a location is created from a randomly-weighted average taken between the point in the search space corresponding with said solution set and the closest other point in the search space that corresponds to another solution set within the Pareto Front. Each of these locations is then placed within the Domain Knowledge container. During the Update Step, if any point on the Pareto Front dominates any point in the Domain Knowledge container, the dominated point is removed from said container.

Topographic Knowledge

Topographic Knowledge was first devised as a knowledge source in [Jin, 1999]. Topographic Knowledge is knowledge concerning the layout and different regions of the search space itself and the performance landscape. In other words, Topographic Knowledge concerns which portions of the search space have yielded good solutions and which have not.

In Cultural Algorithms, the Topographic Knowledge Space is effectively a map of the search space consisting of "Belief Cells". Belief Cells that fail to produce enough optimal results are pruned, while those that do produce enough optimal results are divided into "sub-cells". Topographic Knowledge can be implemented as a recursive "drill-down" mechanism [Reynolds, 2018], and indeed this is the way that it is implemented in CAPSO.

In CAPSO, Topographic Knowledge is the knowledge source that governs how the algorithm searches through the search space as a whole rather than governing individual agent behavior. As mentioned before, Topographic Knowledge works on a recursive "Drill-Down" basis. If the algorithm is searching within a certain portion of the search space and it discovers a parameter set that corresponds to either an entirely new point for the Pareto Front or a point that dominates another point within the Pareto Front, the Topographic Knowledge component will divide the aforementioned portion of the search space into four equal subportions, and the algorithm will recursively search within those subportions.

7.4 CAPSO Population Component

In CAPSO, the Cultural Component (Belief Space) described in the previous section (6.3) acts upon the overall algorithm by influencing a population component. CAPSO's population space uses a particle swarm optimization (PSO) algorithm that borrows its elite selection process from VEGA (Vector Evaluted Genetic Algorithms). VEGA was originally devised in the 1980s by David Schaffer as a type of genetic algorithm for doing multi-objective problems in which the elite is comprised by admitting a certain percentage of the top scorers for each individual objective function taken singly in turn. This is the way that the population elite are chosen in CAPSO's population space algorithm. (In CAPSO, the top 1/7 of scorers for each of the individual objective functions in the problem are admitted into the population elite.) In standard implementations of VEGA, various genetic operators such as mutation and crossover are used to generate a decent spread of individuals so as to partially compensate for the fact that the elite are chosen from the objective functions taken singly. CAPSO, too, uses such genetic operators, but unlike in standard VEGA, individuals in CAPSO are additionally able to take advantage of CA knowledge from the various knowledge sources in the belief space. All-in-all, the CA dovetails well with VEGA because VEGA's simplicity works well in a compound algorithm, likewise cultural knowledge from the CA is able to drastically ameliorate, and oftentimes entirely resolve, the specific shortcomings that come out of VEGA's simplicity.

7.5 CAPSO Component Diagram

Having explained the main parts of CAPSO, we now provide a full component diagram in Figure 50.

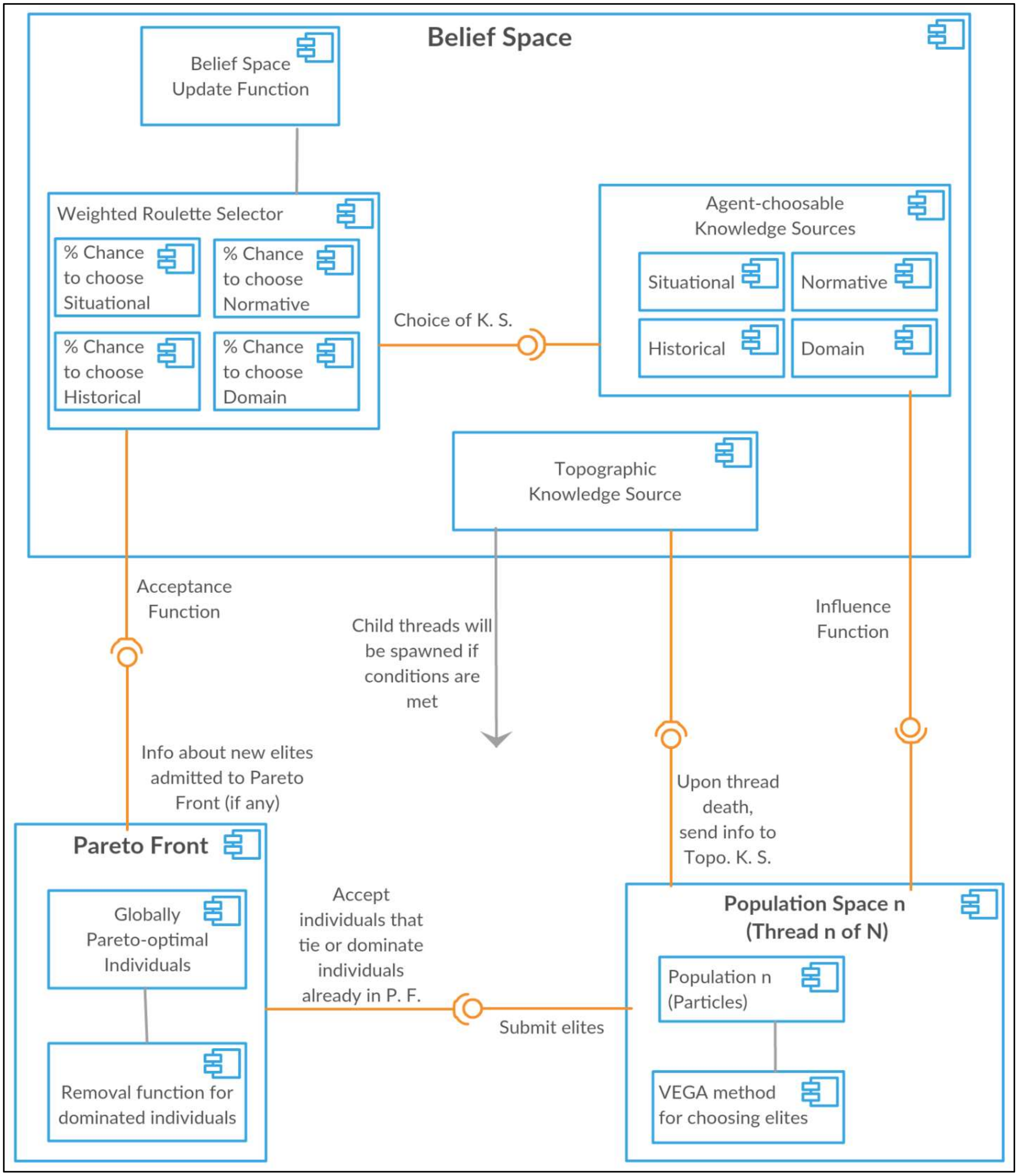


Figure 50: CAPSO Component Diagram

7.6 CAPSO Pseudocode

The CAPSO system is a hybrid system composed of a Particle Swarm and Vector Evaluated Genetic Algorithm population component operating under the control of a Cultural Algorithm framework. The guiding principle in its design is to keep each as vanilla as possible in order to facilitate their interaction and support explicit parallelism in the search process.

The **Main** function recursively calls **SearchInSpace** to generate a new swarm thread. A swarm population is associated with that thread via a call to **PopSpaceAlg**. PopSpaceAlg is in charge of updating the swarm associated with the thread. If any swarm ever goes *maxGensWoImprov* generations without improving the Pareto front, it is removed and the thread associated with it is joined with the main thread. If on the other hand it survives for a number of generations equal to the subdivision threshold (“subdivThresh"), four child threads are spawned each containing an offspring particle swarm, each of whose territory consists of one fourth of the parent swarm’s old territory. After this act of reproduction, the parent swarm dies (is removed) and the thread associated with it is joined with the main thread.

In PopSpaceAlg, selection of an elite takes place via the VEGA method: The population’s agents are ranked according to their performance vis-a-vis each individual objective function taken in turn. If an agent is in the top 1/7 of performers for any of the objective functions, it is added to the elite. Genetic operators (i.e. Crossover, Mutation, and Vector Weighted Average) are then used to create a new generation with an adequate amount of agent “spread”.

CASteps is then called and accepts a certain number of points, elite, into the Belief Space in order to update it**.** It then applies the knowledge sources to selectively modify the remaining ones based upon their relative performance using a weighted Roulette Wheel mechanism.

The process continues recursively until all swarm threads have finished and have joined with the program's main thread. In that case the system can be restarted with a new random swarm but still using the acquired knowledge from the currently completed run and any previous runs that resides in the Belief Space. In the problems described below most were solved in one pass with a second and third try producing no new points. Only SRN benefited from a second and third iteration as shown in Figure 58 in the next section. There the existing front was successfully refined in each the subsequent two steps.

CAPSO Pseudocode Listing:

```
Function Main()
    pFront = ParetoFront.Initialize()
    CA.Initialize()
    SearchInSpace(initSearchSpace)
    #The last line here is recursive, and will continually subdivide #the search domain and #"drill
    down" into each subdivision until specificed stop conditions are reached.


Function SearchInSpace(topographicCell):
    particleSwarm = new ParticleSwarm(topographicCell)
    t = new Thread(func = PopSpaceAlg, arg = particleSwarm)
    if t adds at least 1 new point to ParetoFront && maxRepeats is reached by PopSpaceAlg:
        newSubspaces = DivideIntoEqualPortions(subspace)
        foreach sSpace in newSubspaces:
            searchInSpace(sSpace)

Function PopSpaceAlg(partSwarm):

    DO:
```

```
        #Particle Swarm Movement Step
        Foreach indiv in partSwarm:
            indiv.position += indiv.velocity

        #Pareto Front Update Step
        Foreach indiv in partSwarm:
            If no pFront members dominate or equal F(indiv):
                pFront.Add(F(indiv))
            If F(indiv) dominates an item(s) in pFront:
                remove dominated item(s) from pFront

        #Particle Swarm Elite Selection Step
        elite = SelectElite(VEGA Method [Schaffer, 1985], select top 1/7 of performers according
        to each individual obj function.)

        #Particle Swarm Velocity Update Step
        Foreach indiv in partSwarm and not in Elite:
            rndNum = randomBetween(0, 1)

            If rndNum<0.2: #both crossover and mutation
                Indiv.velocity = Crossover(elite.pickrandom().velocity, indiv.velocity)
                Indiv.velocity = Mutation(Indiv.velocity)

            Else if rndNum<0.4: #(crossover but no mutation)
                Indiv.velocity= Crossover(elite.pickrandom().velocity, indiv.velocity)

            Else if rndNum<0.6: #(mutation but no crossover)
                Indiv.velocity = Mutation(Indiv.velocity)

            Else if rndNum<0.8: #(weighted average)
                Indiv.velocity = vectorWgtAvg(elite.pickrandom().velocity, indiv.velocity)

            Else: #Neither crossover nor mutation

        CASteps(partSwarm, elite)

    UNTIL (++numRepeats == maxRepeats) OR no pFront Improvement for maxGensWoImprov
                                                                        generations

Function CASteps(pop, elite):
    CA.Acceptance(elite)
```

```
CA.Update()
Foreach indiv in pop but not in elite: #CA Influence Step
    indiv.knowSource = CA.ChooseKnowSource(situational, normative, historic, or domain)
    targVelocity = CA.Influence(indiv, indiv.knowSource)
    indiv.velocity = vectWgtAvg(indiv.velocity, targVelocity)
```

7.7 Creating Learning Curves

In situations where an evolutionary algorithm is used in a single-objective problem, a "learning curve" is typically used to track the progress of the algorithm. It is typically a plot of the best-achieved fitness function value vs. the number of generations elapsed. For this problem, we cannot use that methodology because our final deliverable is a Pareto Front rather than a single best-achieved value, so we have come up with an alternate methodology to track the progress of the algorithm: If a solution set (represented by a point in vector-space) is added to the Pareto Front and it does not dominate any existing points in the Pareto Front, a raw score of 5 is added to the total score for the knowledge source currently influencing the particle that achieved that point (10 if it is the first point ever added to the Pareto Front). However, if a point is added to the Pareto Front and it *does* dominate one or more existing points within the Pareto Front, the total score for the knowledge source currently influencing the particle that achieved the new point is incremented by the absolute value of the vector distance between the new point and the closest dominated point.

7.8 Benchmark Tests

For initial evaluation purposes, we are testing our system on four very well-known benchmark problems found in the multi-objective optimization literature: CONSTR, SRN, TNK, and KITA. We have taken the formulations for each of these benchmark problems from [Zhao, 2007] with the exception of KITA which we have taken from [Raquel, 2005]. For each of our benchmark tests, we produce a Pareto Front, learning curves, and a graph of knowledge source dominance over time.

For all four of these benchmark problems, we use the same program input parameters, found in Table 6 below:

Particles in Swarm	1000
Initial Guesses for Situational Knowledge	40
Nonimprovement Thread Cutoff Threshold	3 generations
Max Generations Thread Cutoff Threshold *(If this threshold is hit, the subspace currently being searched will be subdivided and new threads will spawn subswarms in each of the subdivisions as described in the pseudocode.)*	30 generations
Number of Runs	3

Table 6: CAPSO's Inputs

We now present the specifications for the four benchmark problems and CAPSO's results for each:

7.8.1 CONSTR

Problem Formulation

Functions (minimize):	Constraints	Parameter Ranges
$f_1 = x_1$ $f_2 = \frac{(1 + x_2)}{x_1}$	$g_1 = x_2 + 9x_1 - 6 \geq 0$ $g_2 = -x_2 + 9x_1 - 1 \geq 0$	$x_1 \in [0.1, 1.0]$ $x_2 \in [0, 5]$

Table 7: CONSTR Multi-Objective Optimization Benchmark Problem

Problem Overview

CONSTR was first proposed by Kalyanmoy Deb in [Deb, 2001]. CONSTR's Pareto Front is constrained on the right side by x_1's parameter range, it is constrained on the left side by constraint g_2, and it is constrained on the bottom by a combination of x_2's parameter range and constraint g_1. What makes this problem interesting is that a portion of the unconstrained Pareto Optimal region is infeasible. Therefore, constrained optimal Pareto front is a concatenation of the first constraint boundary and a portion of the unconstrained optimal Pareto front.

Our Results for CONSTR

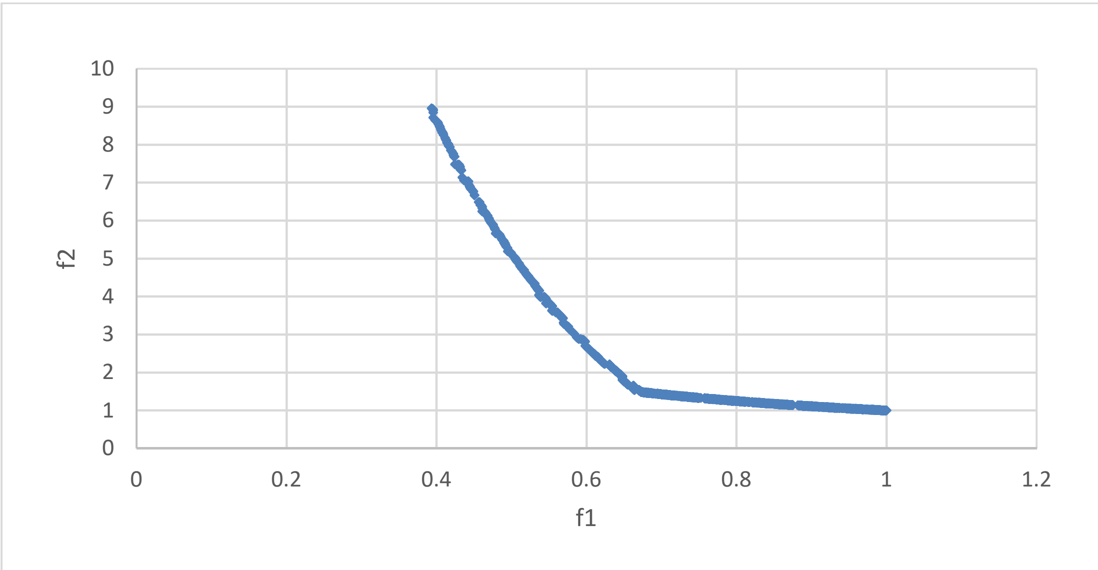


Figure 51: Our Results for CONSTR Multi-Objective Benchmark Problem

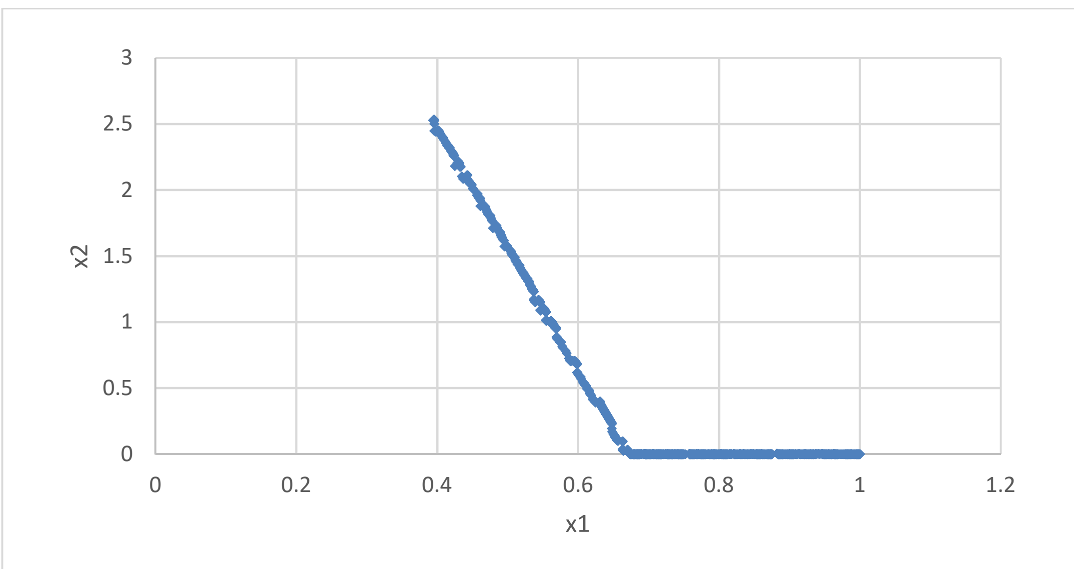


Figure 52: CONSTR Search Space

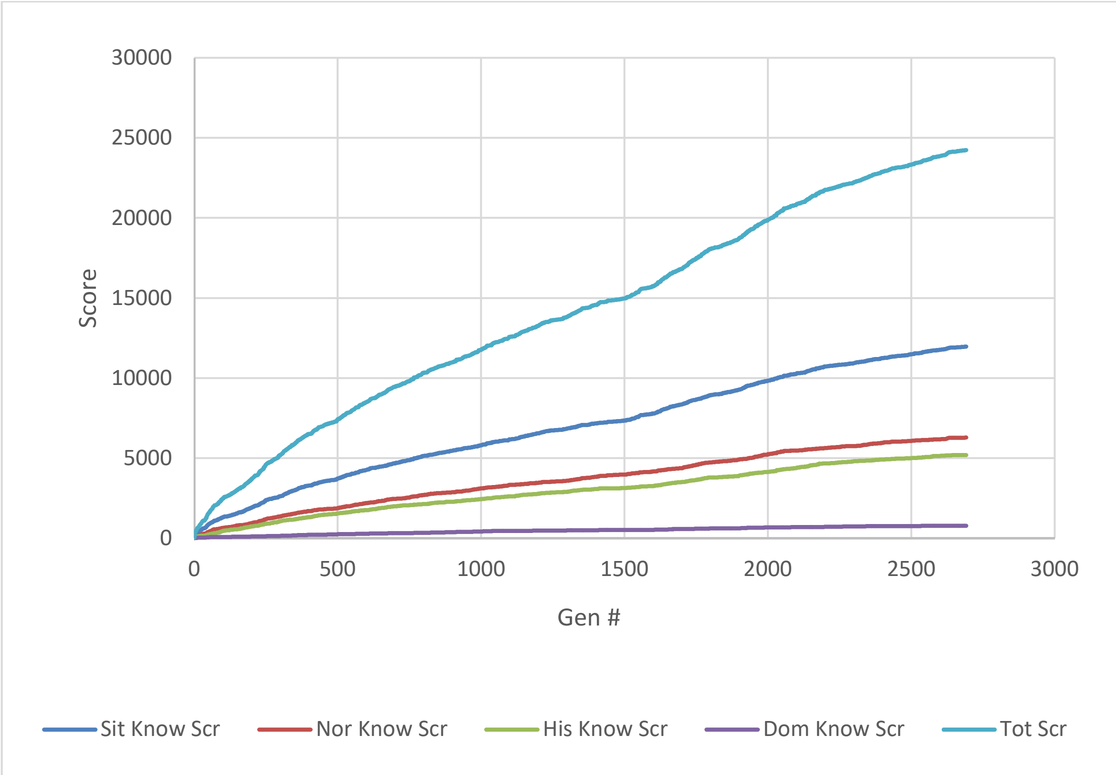


Figure 53: CONSTR Learning Curves

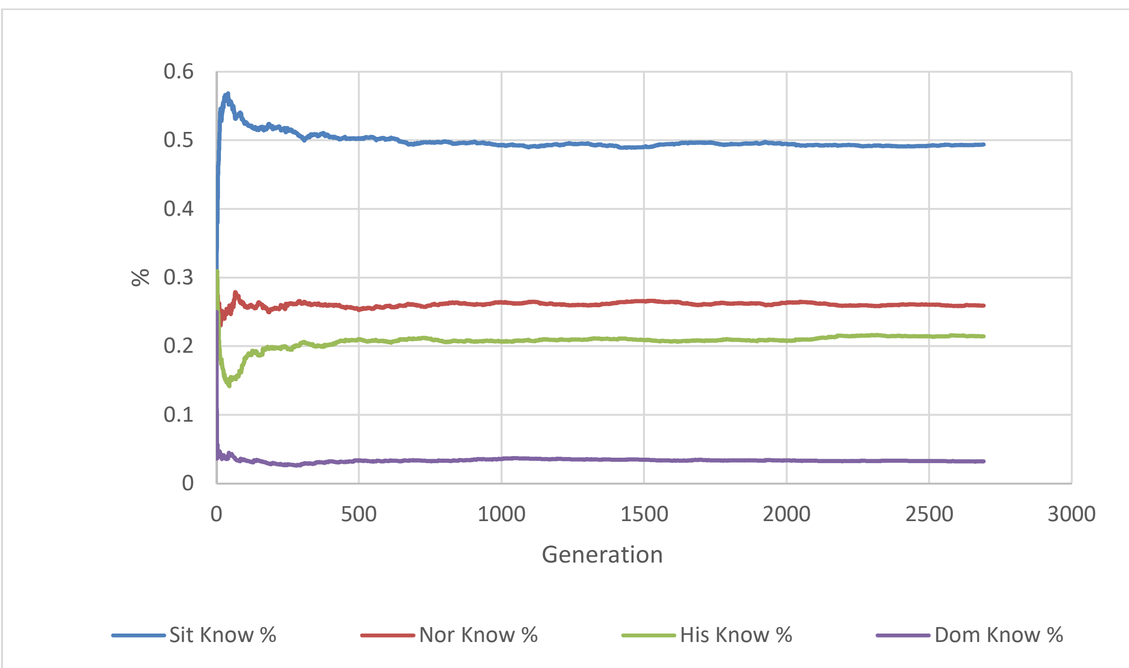


Figure 54: CONSTR Knowledge Source Domination

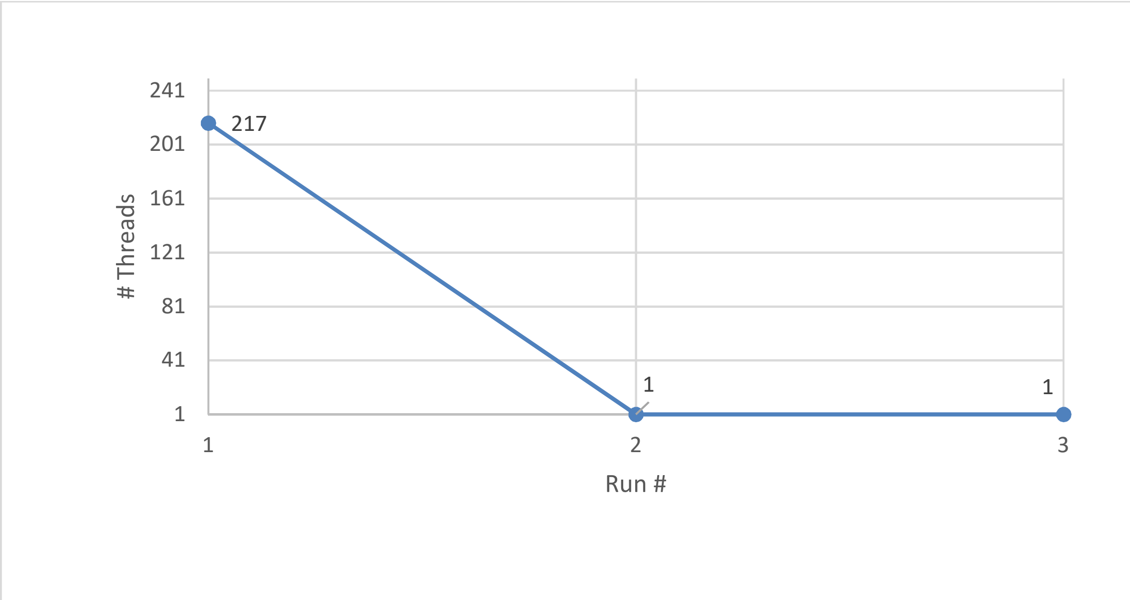


Figure 55: CONSTR: Number of Threads Per Run (Topographic Knowledge Source Progress)

CONSTR Results Discussion

CONSTR was Historical Knowledge's worst performance out of the four problems. This is most likely because the parameters corresponding to the Pareto Front (Figure 51) form two distinct intersecting lines with a very abrupt transition between the two. Any Historical Knowledge gained through the discovery of one of these lines is useless in intuiting the other.

On the other hand, CONSTR was Situational Knowledge's best performance among the four problems, reaching nearly 50% dominance among the four knowledge sources (Figure 53). This is probably because there happened to be two (or more) initial guesses corresponding to the correct velocity "moves" needed to discover the two lines.

7.8.2 SRN

Problem Formulation

Functions (minimize)	Constraints	Parameter Ranges
$f_1 = (x_1 - 2)^2 + (x_2 - 1)^2 + 2$	$g_1 = x_1^2 + x_2^2 - 225 \leq 0$	$x_1 \in [-20, 20]$
$f_2 = 9x_1 - (x_2 - 1)^2$	$g_2 = x_1 - 3x_2 + 10 \leq 0$	$x_2 \in [-20, 20]$

Table 8: SRN Multi-Objective Benchmark Problem Specification

Problem Overview

SRN was first proposed by N. Srinivas in [Srinivas, 1994]. SRN is a difficult problem is due to the large search space and the large number of particle moves needed to flesh out the entire Pareto Front (see Figure 57).

Our Results for SRN

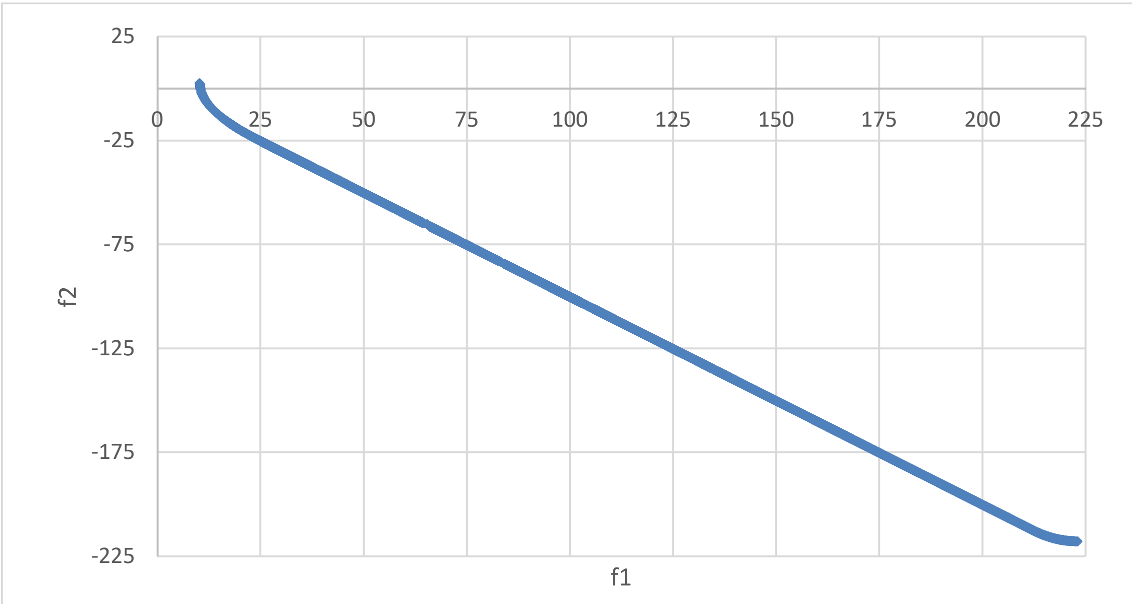


Figure 56: CAPSO's Results for SRN Multi-Objective Benchmark Problem

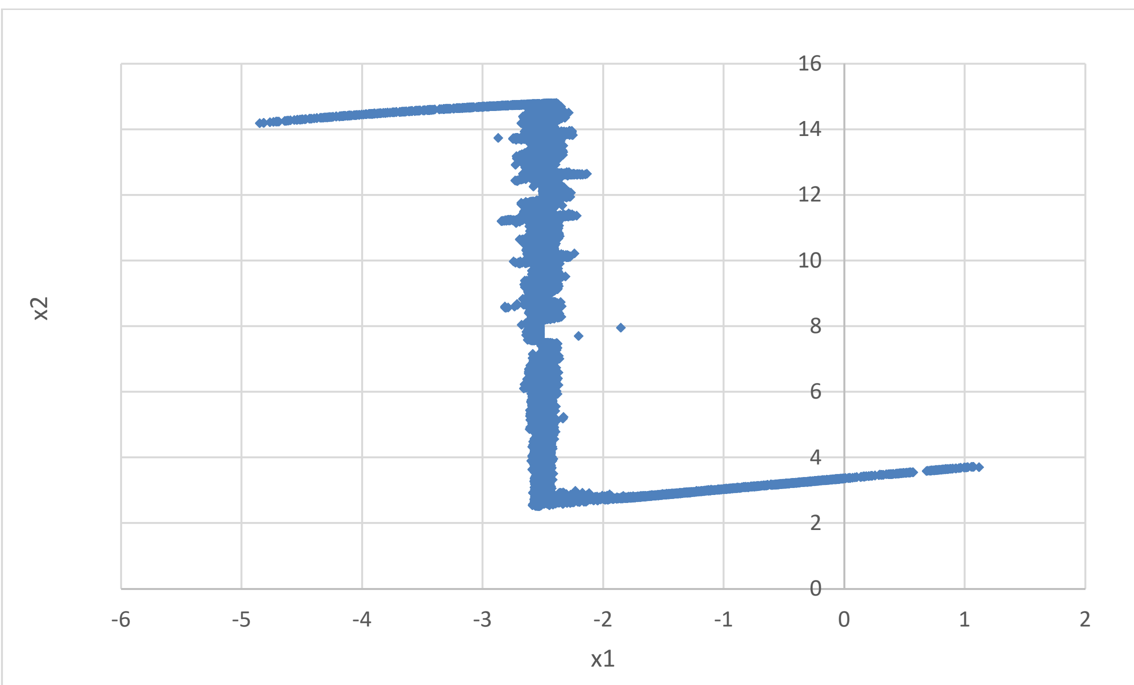


Figure 57: SRN Search Space

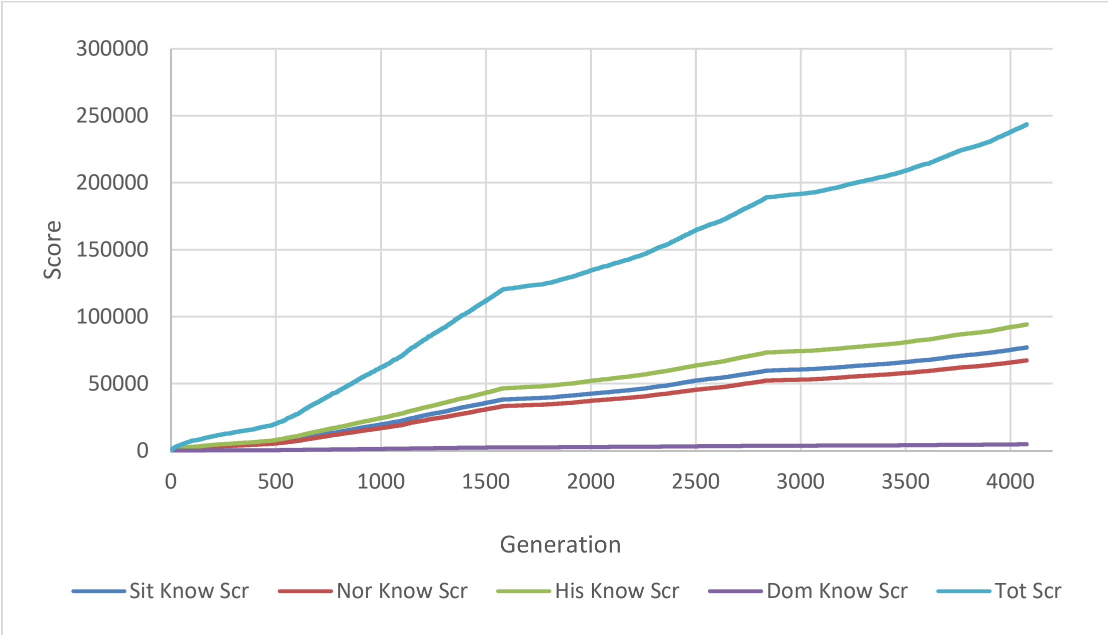


Figure 58: SRN Learning Curves

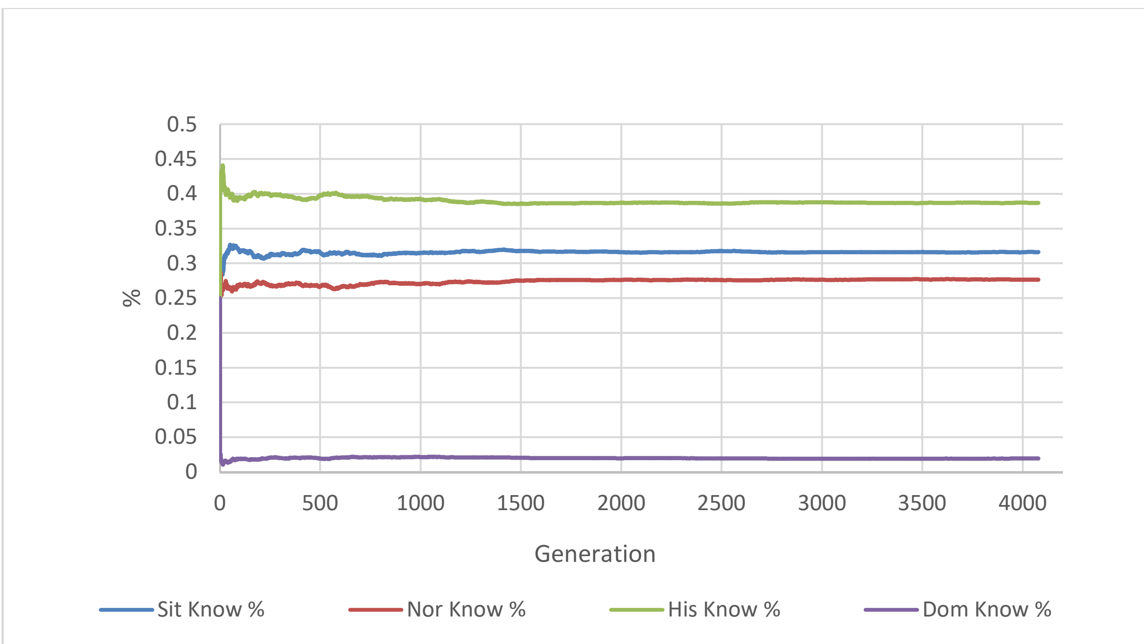


Figure 59: SRN Knowledge Source Dominance Graph

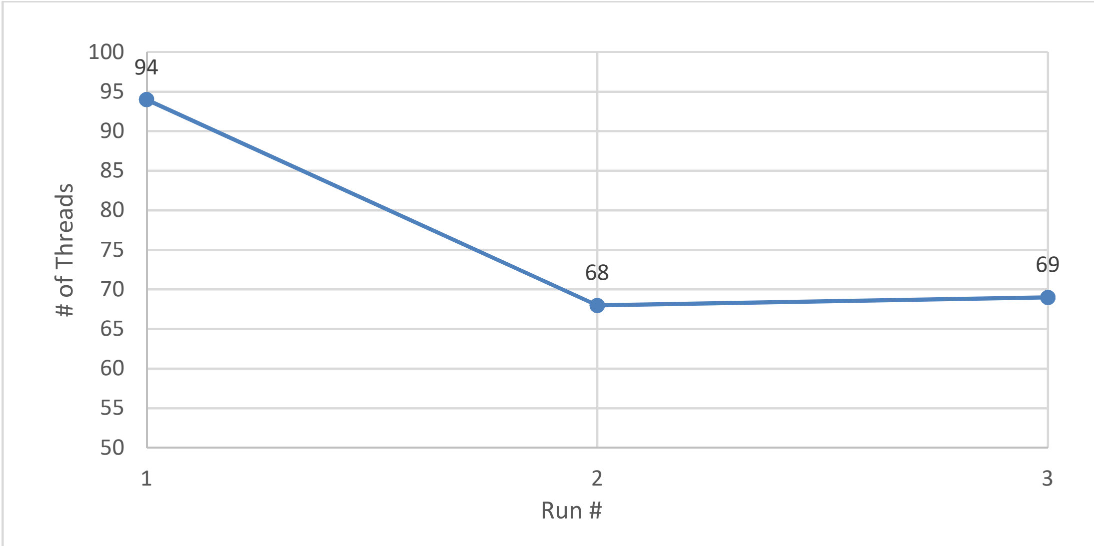


Figure 60: SRN Threads Per Run (Topographic Knowledge Progress)

SRN Results Discussion

In our evaluation of SRN, Historical Knowledge was the best-performing knowledge source. This is probably because the set of parameter pairs corresponding to the Pareto Front (Figure 57) is mostly composed of a thick central “shaft”. This “shaft” can be discovered through making similar back-and-forth velocity motions that can be stored within the Historical Knowledge space.

7.8.3 TNK

Problem Formulation

Functions (minimize)	Constraints	Parameter Ranges
$f_1 = x_1$ $f_2 = x_2$	$g_1 = -x_1^2 - x_2^2 + 1 + 0.1\cos\left(16\,arctan\left(\frac{x_1}{x_2}\right)\right) \leq 0$ $g_2 = (x_1 - 0.5)^2 + (x_2 - 0.5)^2 - 0.5 \leq 0$	$x_1 \in [0, 0.5 + \sqrt{0.5}]$ $x_2 \in [0, 0.5 + \sqrt{0.5}]$

Table 9: TNK Multi-Objective Benchmark Problem Specification

Problem Overview

TNK was first proposed by M. Tanaka in [Tanaka, 1995]. TNK's second constraint, g_2, designates as infeasible any solution set that is outside a circle whose center is at (0.5, 0.5) and whose radius is $\sqrt{2}$. The effect of this constraint is to "clip" the Pareto Front so that the leftmost and rightmost ends are slightly shorter than they otherwise would be. The first constraint designates as infeasible any solution set lying inside a hypotrochoid whose formula is given by g_1. TNK's Pareto Front has two discontinuities. The first is caused by the fact that the portion of the hypotrochoid going from $x_1 \in (0.195, 0.459)$ lies up and to the right of the portion going from $x_1 \in (0.056, 0.186)$, the latter thus dominating the former. The second discontinuity is caused by the fact that the portion of the hypotrochoid going from $x_2 \in (0.173, 0.460)$ lies directly above the portion going from $x_2 \in (0.057, 0.173)$, the latter once again dominating the former.

Our Results for TNK

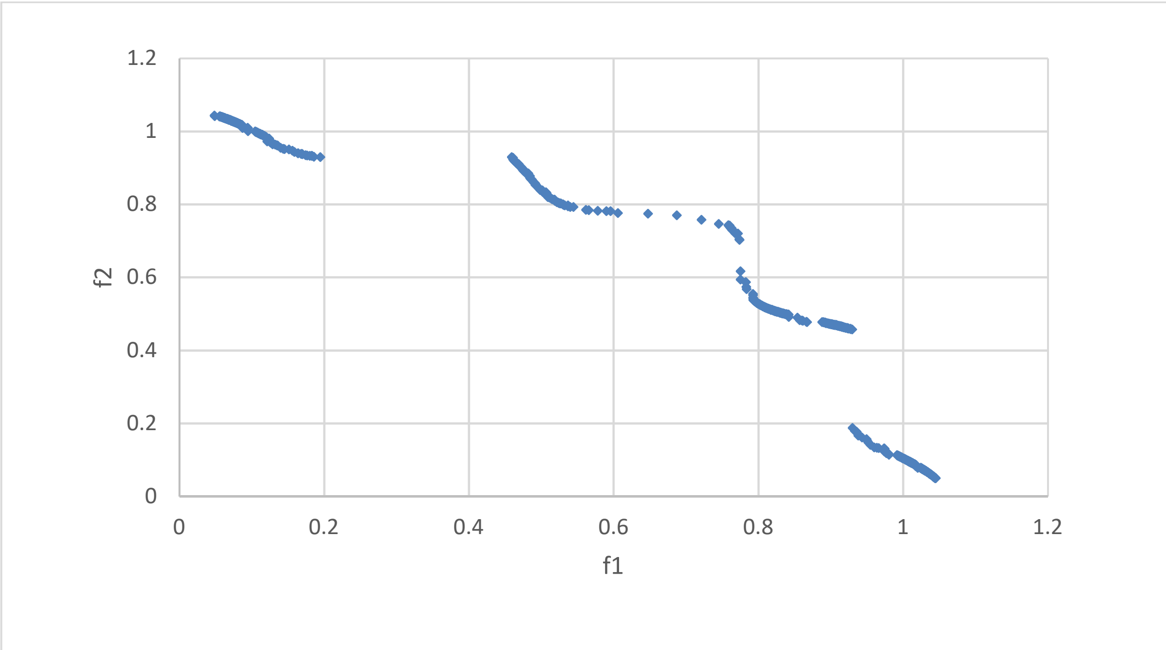


Figure 61: TNK Pareto Front

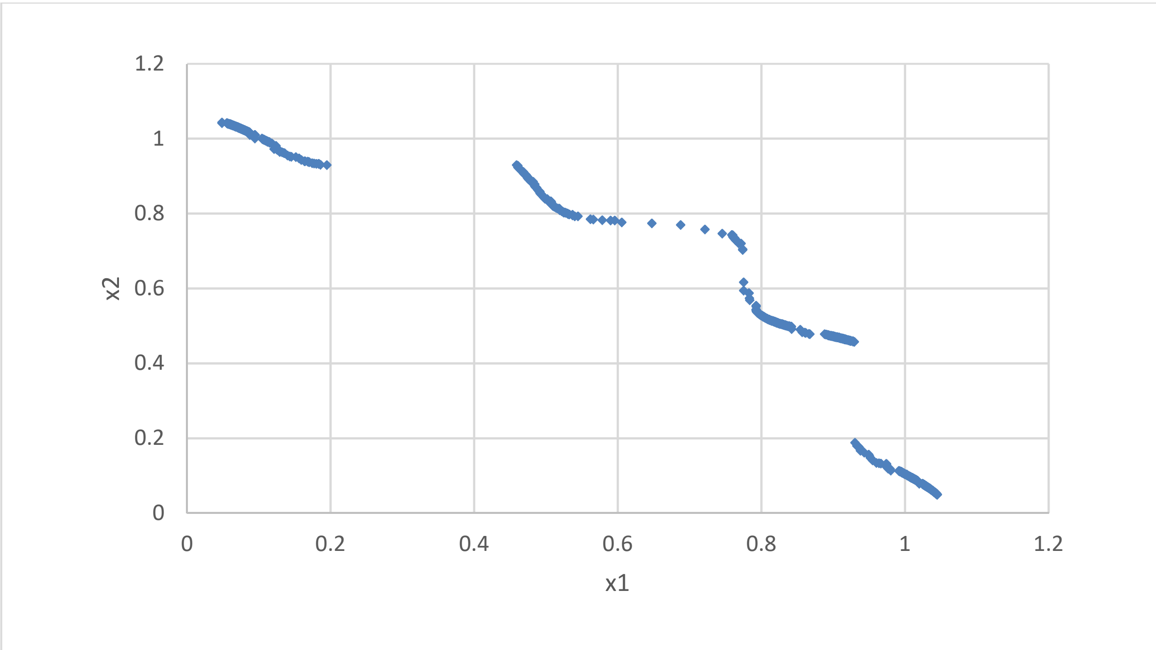


Figure 62: TNK Search Space

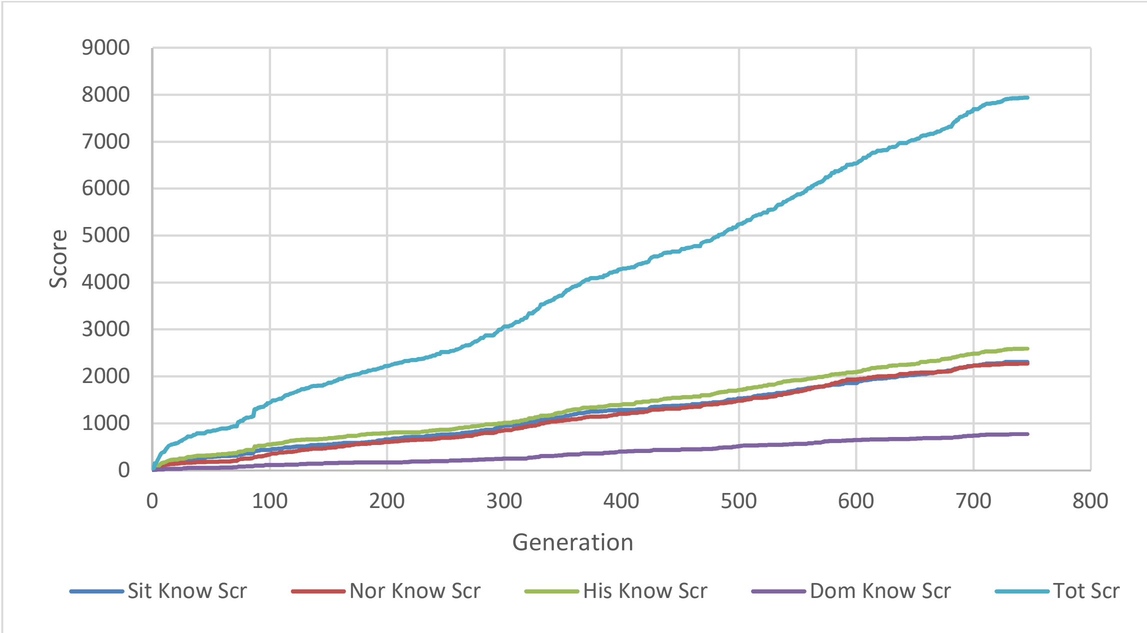


Figure 63: TNK Learning Curves

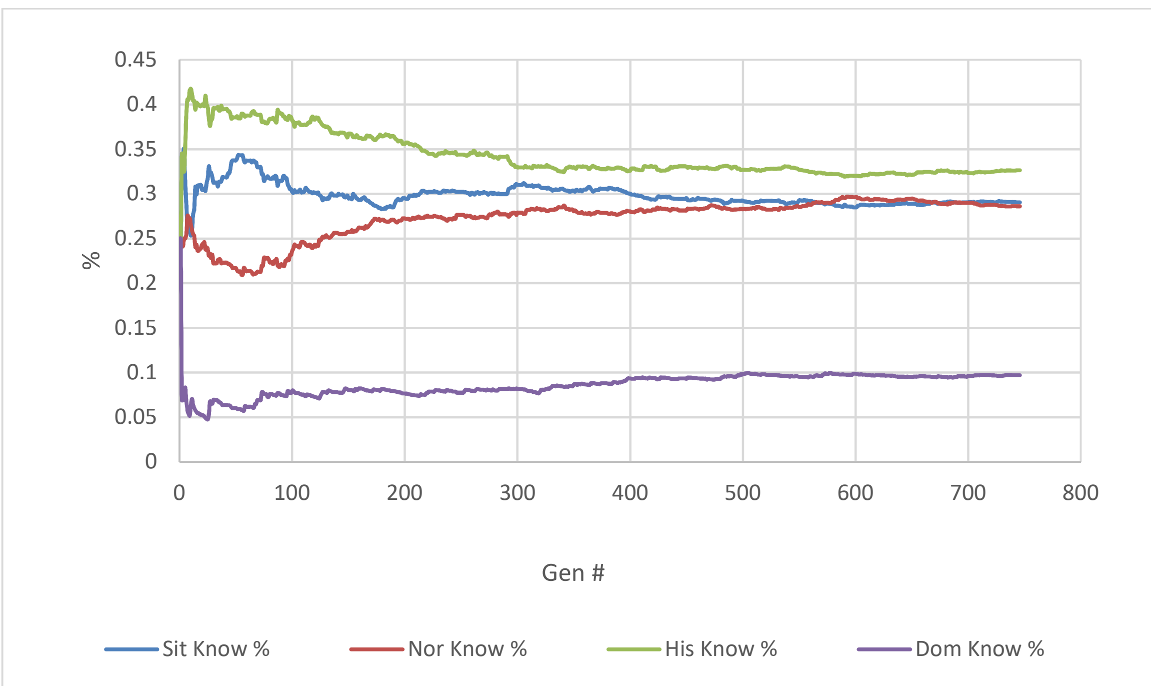


Figure 64: TNK Knowledge Source Dominance

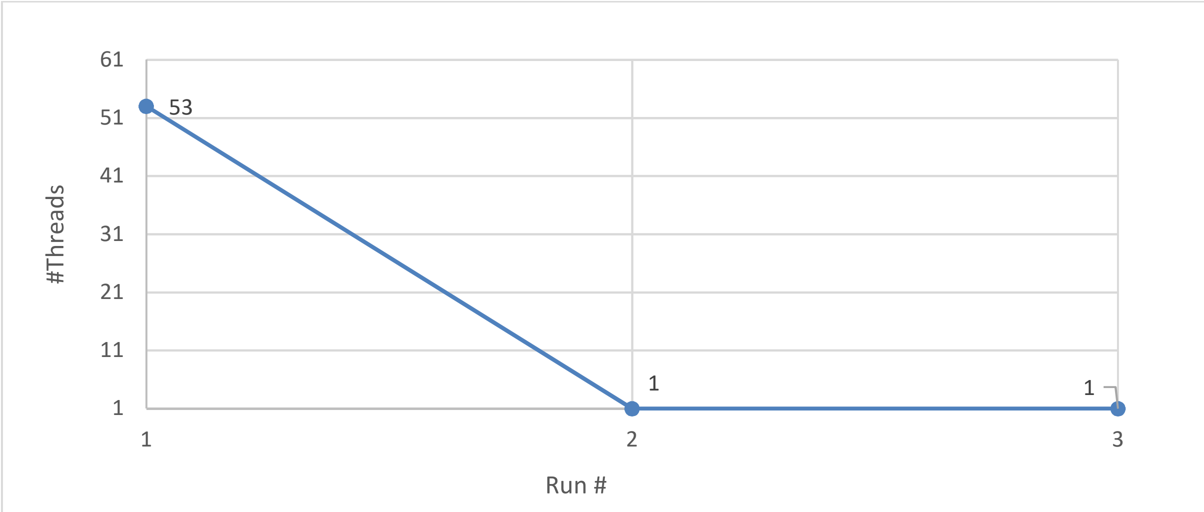


Figure 65: TNK Threads Per Run (Topographic Knowledge Progress)

TNK Results Discussion

TNK is an interesting problem not only because the Pareto Front is disjoint, but because each of the functions are simply set equal to each of the parameters (i.e., $f_1 = x_1$ and $f_2 = x_2$), thus the graph of TNK's Pareto Front (Figure 61) is exactly the same as the graph of the parameter values used to achieve it (Figure 62). Historical Knowledge was the best knowledge source in our evaluation of TNK, finishing with around 32.6% dominance among the four knowledge sources. This is probably because even though the Pareto Front is disjoint, there are some parts which are extremely similar to other parts. For instance, the portion stretching from $f_1 \in (0.05, 0.2)$ is extremely similar in shape and slope to the portion stretching from $f_1 \in (0.8, 0.92)$. Thus, Historical Knowledge used to fully discover one of these could be used to fully discover the other.

7.8.4 KITA

Problem Formulation

Functions (maximize)	Constraints	Parameter Ranges
$f_1 = -x_1^2 + x_2$ $f_2 = \frac{x_1}{2} + x_2 + 1$	$g_1 = \frac{x_1}{6} + x_2 - \frac{13}{2} \leq 0$ $g_2 = \frac{x_1}{2} + x_2 - \frac{15}{2} \leq 0$ $g_3 = \frac{5}{x_1} + x_2 - 30 \leq 0$	$x_1 \in [0, 7]$ $x_2 \in [0, 7]$

Table 10: KITA Multi-Objective Benchmark Optimization Problem Specification

Problem Overview

KITA was first proposed by H. Kita in [KITA, 1996]. Out of the four benchmark problems that we evaluated, Domain Knowledge most came into play in KITA.

Our Results for KITA

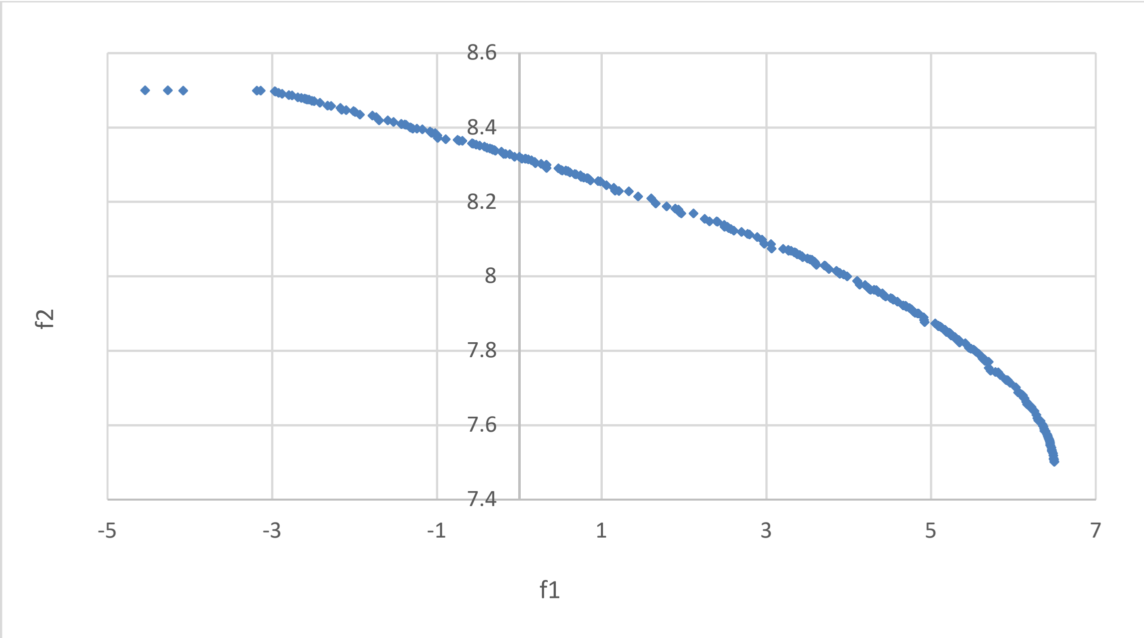


Figure 66: KITA Pareto Front

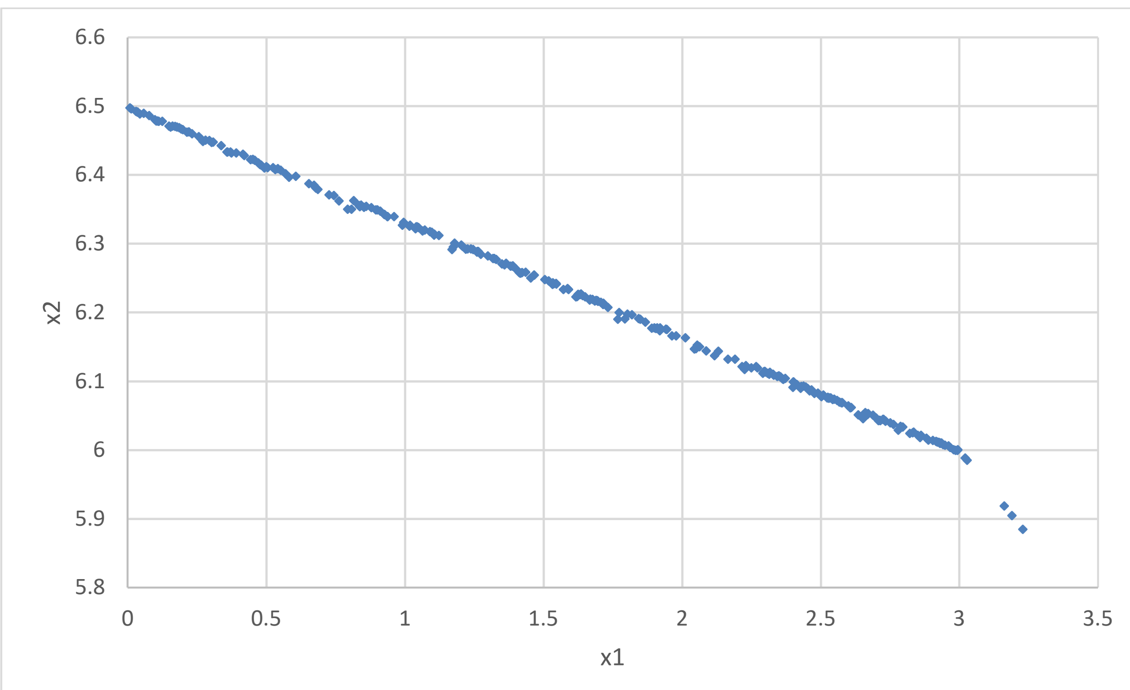


Figure 67: KITA Search Space

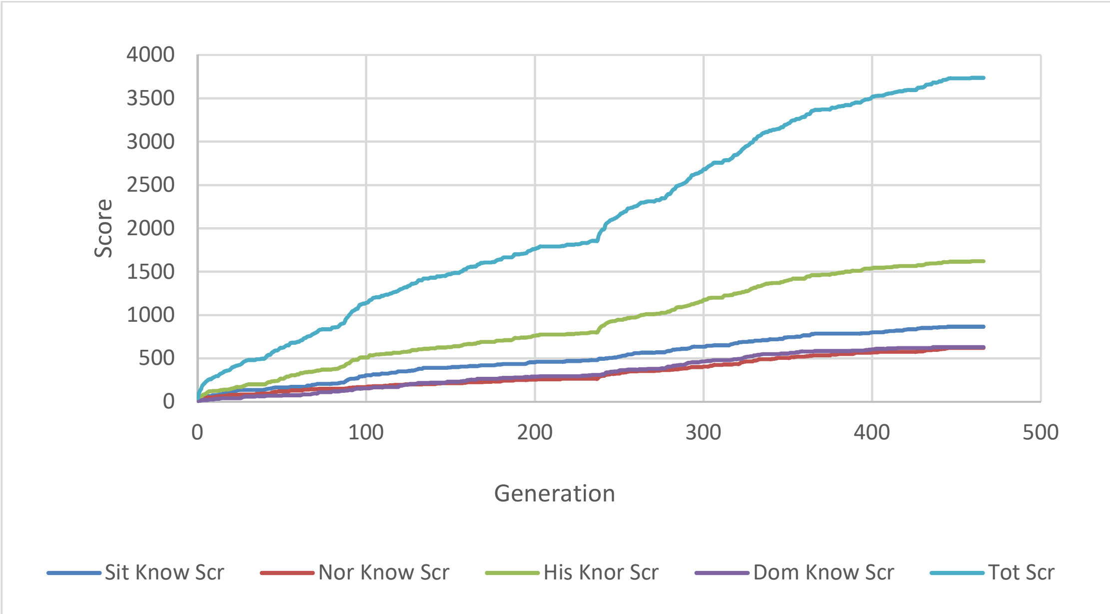


Figure 68: KITA Learning Curves

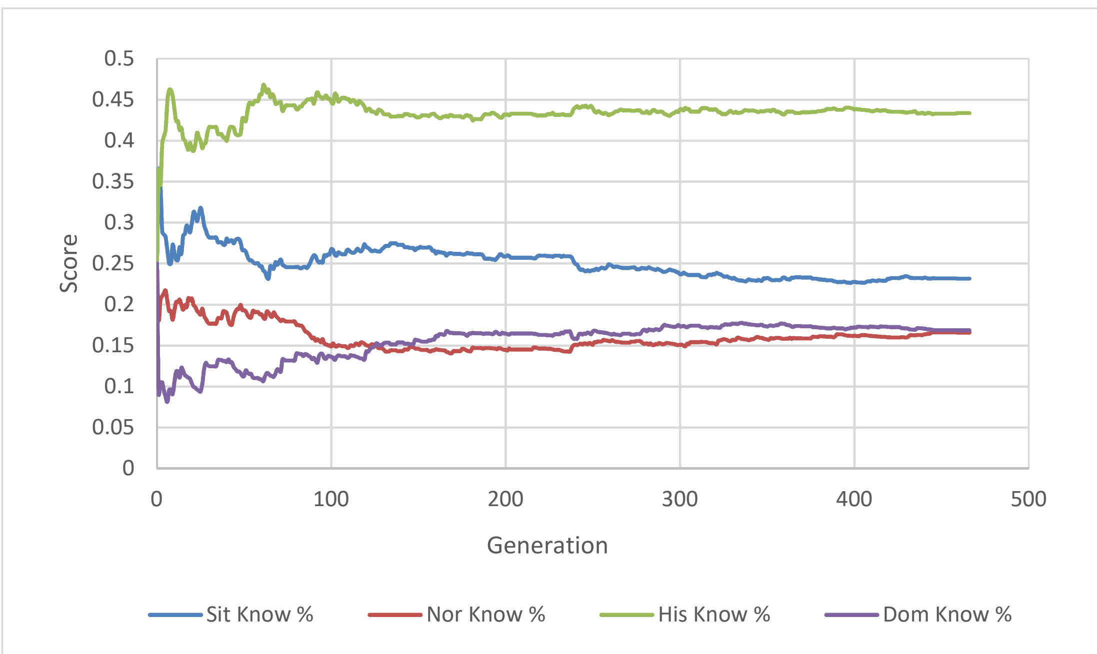


Figure 69: KITA Knowledge Source Dominance

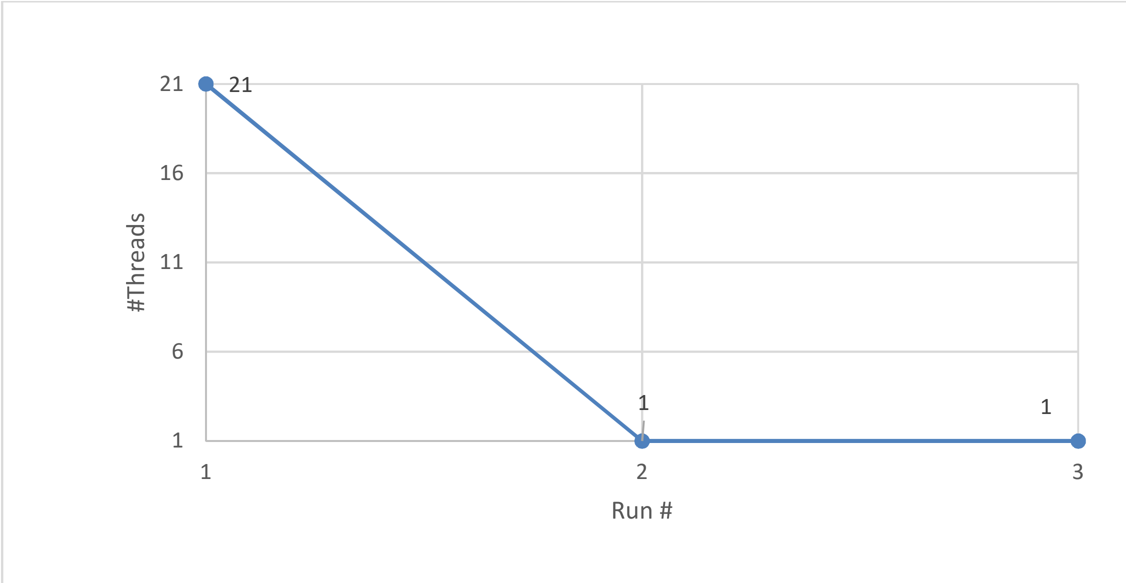


Figure 70: KITA Threads Per Run (Topographic Knowledge Progress)

KITA Problem Discussion

KITA was Situational Knowledge's best-performing problem because the parameter values corresponding to the achieved Pareto Front almost entirely corresponded to a single line with domain $x_1 \in (0, 3)$ and with a slope of -2.167. The velocity "moves" needed to "flesh out" this line after its initial discovery would thus logically correspond with this slope, which is a fact very easily remembered by Historical Knowledge. In our evaluation of KITA, Historical Knowledge finished with around 43.4% dominance, way ahead of the other knowledge sources.

Domain Knowledge finished third out of the four knowledge sources for KITA. KITA was the only problem where Domain Knowledge did not finish last out of the knowledge sources. In general, Domain Knowledge is usually the least-dominant knowledge source because it is effectively the "clean-up crew" which polishes up Pareto Fronts which have been achieved by the other knowledge sources.

In KITA, Domain Knowledge played an explorative role by helping establish the outer boundaries of the Pareto Front along with an exploitative role by removing certain subtly non-Pareto optimal solution sets. This is because the leftmost side of KITA's Pareto Front contains a "tail" that begins when $f_1 < -3$ and after a certain point begins to very subtly start bending backward, putting forth solutions that are very subtly non-Pareto optimal. In preliminary trials of CAPSO, before Domain Knowledge was implemented into the program, this "tail" would sometimes reach as far back as $f_1 = -42$. Once Domain Knowledge was implemented, the sub-optimal portions of the "tail" stopped appearing in the results. The remaining portion appears to be weakly Pareto-optimal.

7.9 Benchmark Test Conclusions

In all four of these benchmark problems, the exploitative knowledge sources (i.e., Situational and/or Historical) dominate from the very start. These problems have some very interesting and even potentially deceptive features (e.g., the "KITA tail"). It is probably worth testing one's optimizer system to see how it deals with these sorts of features, on the other hand, the start-to-finish dominance of exploitative knowledge sources betray a certain lack of complexity to these problems when compared to, for instance, the real-world problems detailed in the next chapter. The reason why there is such a dominance of exploitative knowledge sources for these problems is that most of the work here consists of continually filling in smaller and smaller gaps in their long and continuous Pareto fronts. Frankly, these Pareto fronts would probably look the exact same to the human eye if the optimizer's cutoff point was far sooner for these problems.

CHAPTER 8: EXPERIMENTAL SETUP AND RESULTS

The problem that we need to do for each of the four structure types (Hunting Blinds, Drive Lines, Caches, and Logistical Camps) is a bi-objective optimization problem in which Equation 5 from Chapter 5 determines the number of flagged locations, and is minimized while Equation 4 from Chapter 5 determines the number of structures of a given type within those locations, and is maximized. Before setting up this problem in CAPSO, we must first generate the environmental parameter data described in Table 1 in Chapter 5 which is what goes into the rules that determine the values of Equations 5 and 6 during each evaluation thereof. The Land Bridge Environmental Parameter Program (interface shown in Figure 1 in Chapter 1) is what generates this data.

Table 11 below contains the initial inputs entered into the Land Bridge Environmental Parameter Program:

Start year	11800 BP
End year	8400 BP
Timestep	200 years
Effort (initial)	10
Risk (initial)	20
Nutrition (initial)	90
Consume	100
Grow	50
Herd Size	40
Calories	400
Cal Cost	10
Cal Benefit	100
Fall Entry	Enter (2, 2) Exit East Deny North & West
Spring Entry	Enter (193, 199) Exit North Deny South & East

Table 11: Land Bridge Environmental Parameter Program Inputs

Figure 71 below is a composite image of all time slices that were generated by the Land Bridge Environmental Parameter Program given our initial inputs in Table 11. Green designates land, blue designates water, and red designates the caribou path. (The images in Figure 71 are reprinted in full-page scale in the Appendix - Figure 158 through Figure 193.)

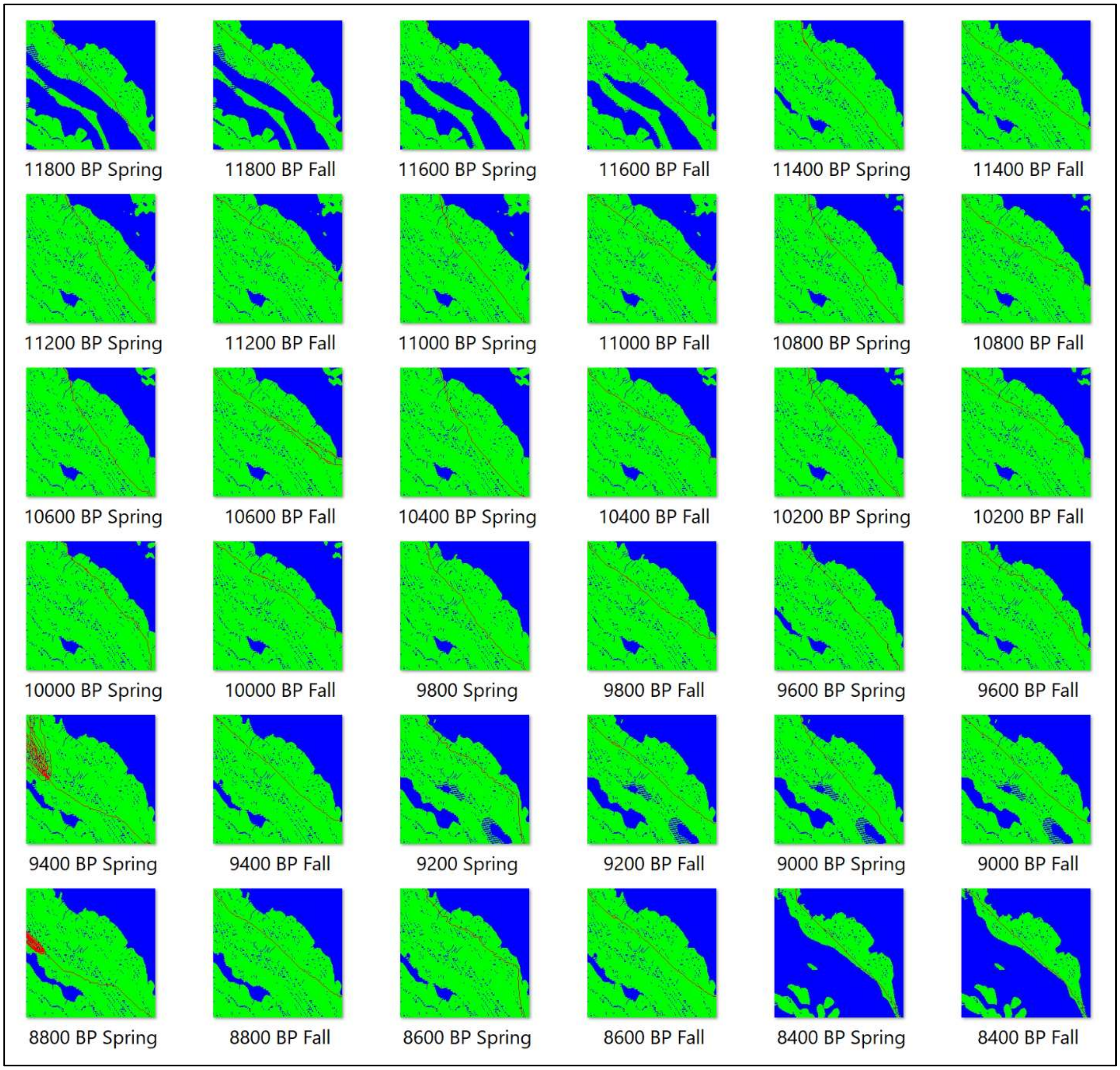


Figure 71: 11800BP-8400BP Composite from Land Bridge Environmental Parameter Program

We now have the data necessary to do the Land Bridge Problem for each of the four structure types. We now place Equation 4 and Equation 5 into CAPSO as two objectives in a bi-objective problem and tell CAPSO to maximize the former and minimize the latter while plugging the data that we just produced from the Land Bridge Environmental Parameter Program into these objectives. (If necessary, please review Chapter 5 for the full explanation.)

As for the CAPSO program inputs, we use the same values throughout our experiments in this chapter. These can be found below in Table 12 below.

Particles in Swarm	100
Initial Guesses for Situational Knowledge	15
Nonimprovement Thread Cutoff Threshold	3 generations
Max Generations Thread Cutoff Threshold	9 generations
Number of Runs	4

Table 12: CAPSO Program Inputs for the Land Bridge Problem

8.1 Hunting Blinds

8.1.1 Hunting Blinds CAPSO Output

Table 13 contains the Pareto Front along with corresponding parameter values for the Hunting Blind structure type. It took 172.217 hours for CAPSO to produce these results.

# Predicted Locations	# Predicted Structures	Rules:	Dist to Caribou <	Height Above or Below Caribou <	Veg. % >	Dist to T. Set Drive Line <	Dist to T. Set Logistical Camp <	Timestep Threshold	Prediction Threshold
1	5	Thresh:	3774.46914	15.1240928	0.75885	159.794246	1669.80513	3	10
1	5	Thresh:	3767.61556	11.732957	0.766371	8.34140211	1080.33418	3	10
2	7	Thresh:	3995.00896	7.59481961	0.415068	83.1500666	1827.3879	3	9
4	10	Thresh:	2746.38521	11.5644474	0.425523	174.77019	1171.70172	3	6
6	14	Thresh:	2818.98021	9.5374702	0.53204	467.536699	2920.3759	3	11
10	15	Thresh:	2546.10723	2.17583673	0.640698	474.919748	2057.45224	3	6
12	16	Thresh:	1284.64311	7.68797613	0.569737	460.102827	2199.27112	3	4
11	16	Thresh:	1749.15627	19.093634	0.227354	364.499283	2107.42676	3	6
14	17	Thresh:	3558.70909	2.53402449	0.302898	479.740292	2238.10853	3	6
17	18	Thresh:	2311.57512	7.19710671	0.093239	399.663707	2027.05686	3	4
20	25	Thresh:	2919.0686	6.06427488	0.592259	452.615644	2026.8432	3	6
26	27	Thresh:	3059.78855	6.90097375	0.598525	367.791875	2242.21892	3	5
31	28	Thresh:	3884.13132	7.36275545	0.647653	453.640866	2350.23546	3	4
34	29	Thresh:	3551.62591	7.85175025	0.59673	401.002546	4158.5136	3	8
54	32	Thresh:	3819.47839	6.425704	0.36636	422.931975	2049.36695	3	3
63	33	Thresh:	3782.30335	8.43152735	0.767992	354.558186	2722.52705	3	3
91	34	Thresh:	3677.31587	14.4705152	0.203699	524.928862	2539.85266	3	3
124	35	Thresh:	3733.2714	9.09681636	0.508113	688.283358	2779.21805	3	3
240	36	Thresh:	3991.75189	15.1047659	0.44677	629.874339	4187.95066	3	3
306	37	Thresh:	3773.66048	15.0350041	0.440155	521.421269	3802.52468	3	2
614	38	Thresh:	3826.32138	5.68015002	0.566012	531.261162	4479.54682	3	1
763	39	Thresh:	3914.94629	1.41021293	0.45298	963.122757	2310.22179	2	3
2065	44	Thresh:	3972.5843	4.01738891	0.289475	538.89339	3557.65329	2	3
4539	46	Thresh:	3847.76593	5.15644031	0.28891	332.484986	4277.45444	2	2
10184	47	Thresh:	3665.22138	15.068512	0.415672	700.907697	4376.57135	2	1
8189	47	Thresh:	3609.20026	5.94267616	0.343558	63.5010971	4261.34251	2	1

Table 13: CAPSO Output – Hunting Blinds Structure Type

8.1.2 Hunting Blinds Pareto Front and Knowledge Source Progress and Dominance Graphs

Figure 72 through Figure 76 contain graphs of the Pareto Front, knowledge source progress, and knowledge source dominance for the Hunting Blind structure type.

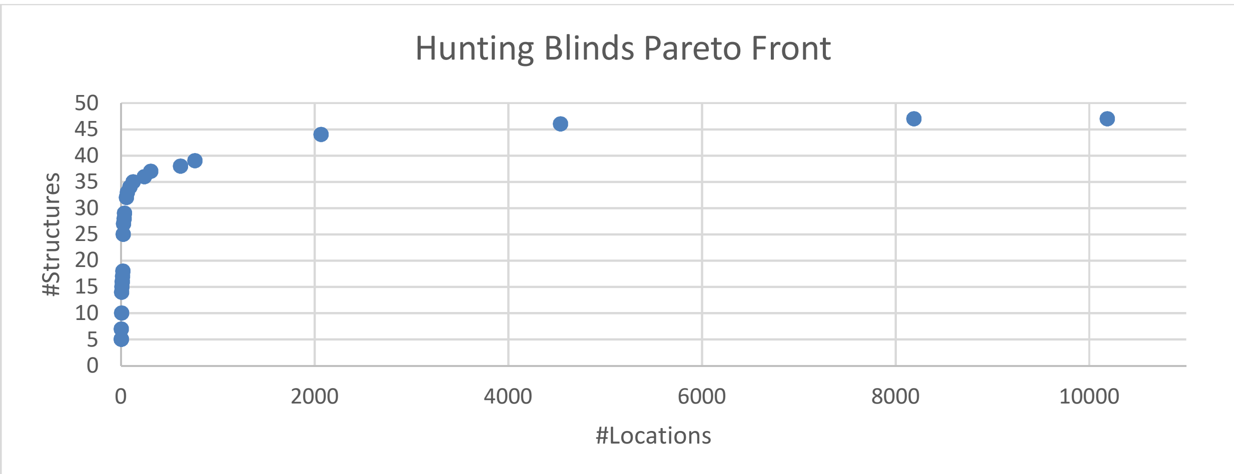


Figure 72: Hunting Blinds Pareto Front

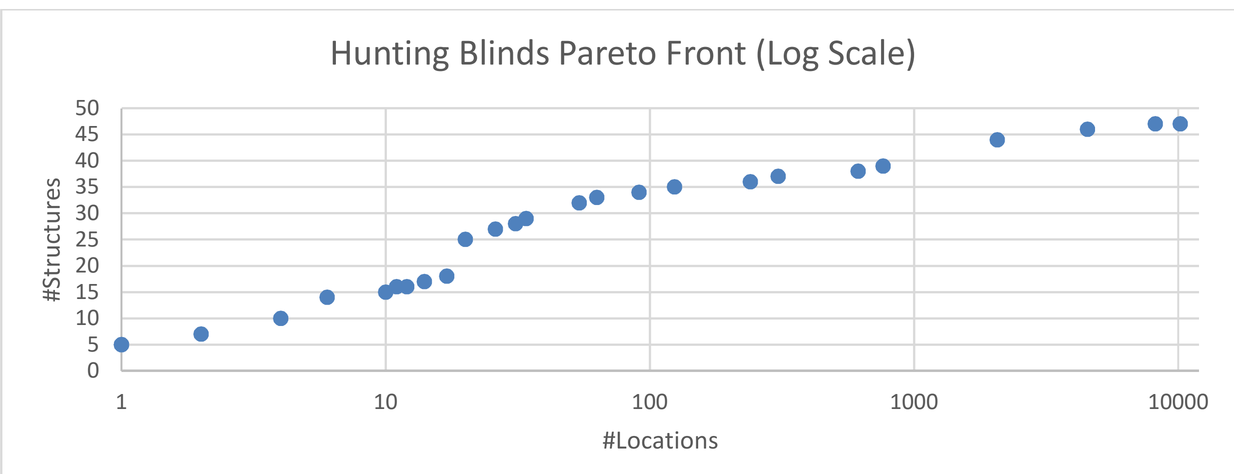


Figure 73: Hunting Blinds Pareto Front (Logrithmic Scale)

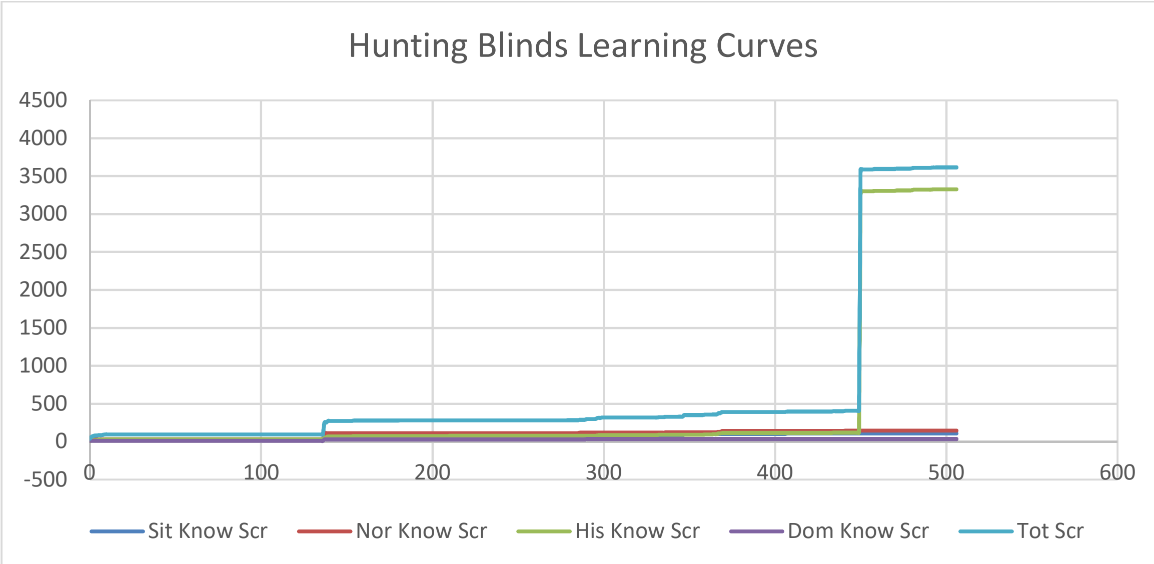


Figure 74: Hunting Blinds Learning Curves

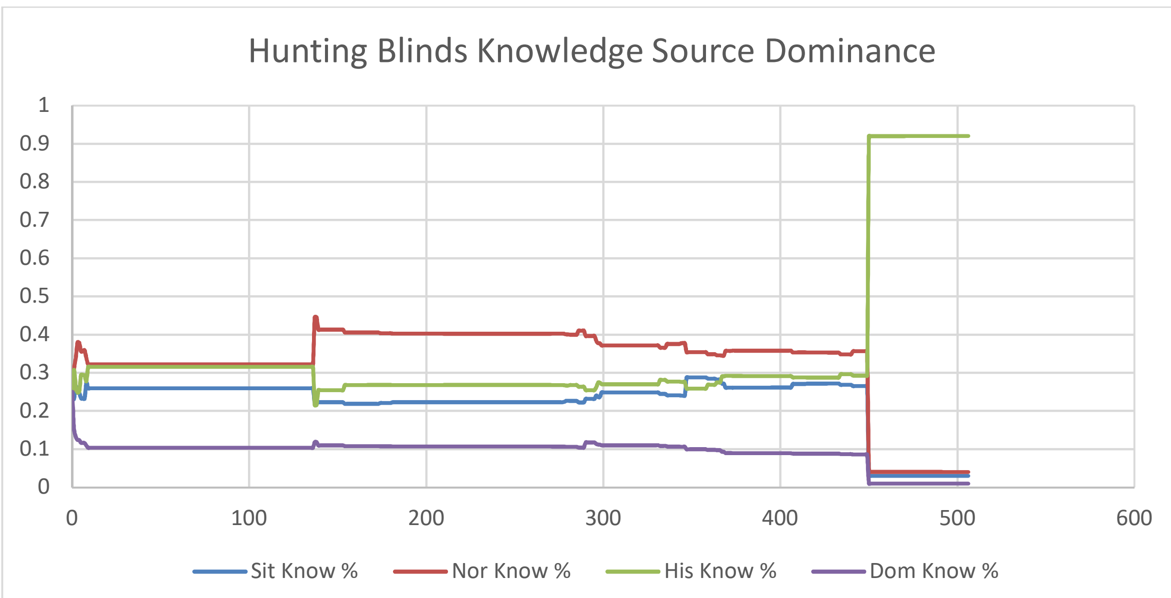


Figure 75: Hunting Bilnds Knowledge Source Dominance

The learning curves for the Hunting Blind structure type were the most interesting of the four. Explorative Knowledge, in the form of Normative Knowledge, was the clear dominator until Generation 450. At that point, Exploitative Knowledge, in the form of Historical Knowledge, jumps way ahead.

In retrospect, these learning curves make sense for the Hunting Blinds problem, as due to the large number of Hunting Blinds in the training set, this is a superbly large and difficult problem, far more so than any of our benchmark problems in Chapter 6 or even the three other real-world problems relating to the three other occupational structure types. So in this extremely large and difficult problem, both Explorative and Exploitative Knowledge have to play their various roles at the proper times rather than one of them simply "winning" throughout the entire process. Namely, Explorative Knowledge "explores" during the first portion of the optimization process until Exploitative Knowledge finds a critical opportunity that its knowledge can "exploit", sending it ahead of Explorative Knowledge at that point.

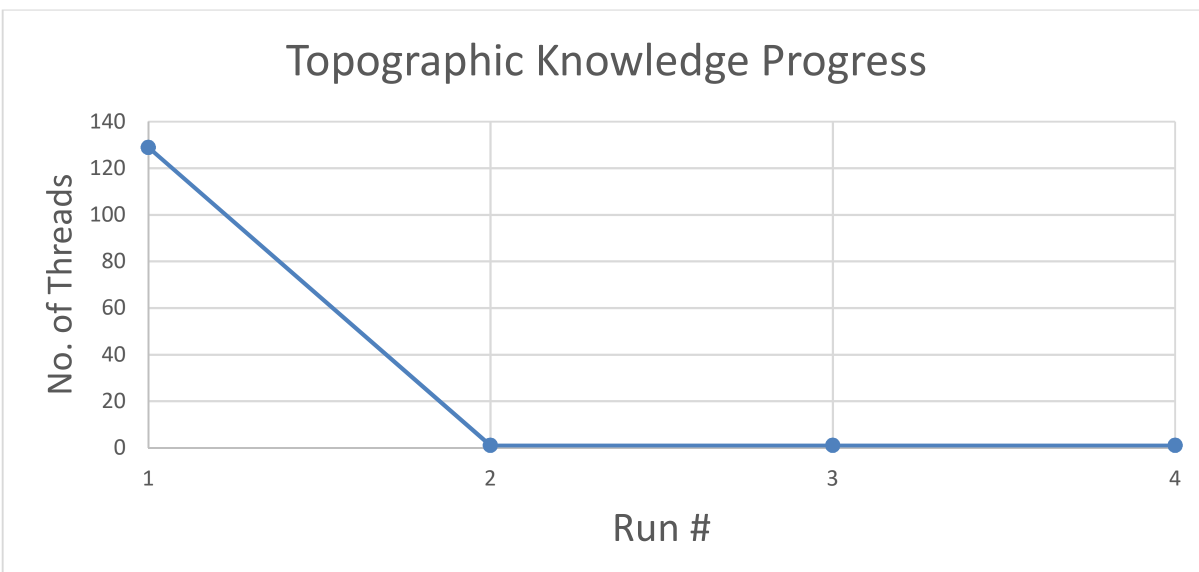


Figure 76: Topographic Knowledge Progress

8.1.3 Hunting Blinds Frames

We can use a data visualizer system to convert each of the entries in *Table 13: CAPSO Output – Hunting Blinds Structure Type* into a geographical heatmap corresponding to each entry. Said heatmaps can be found in Figure 78 through Figure 103. Below in Figure 77 is the key for said heatmaps.

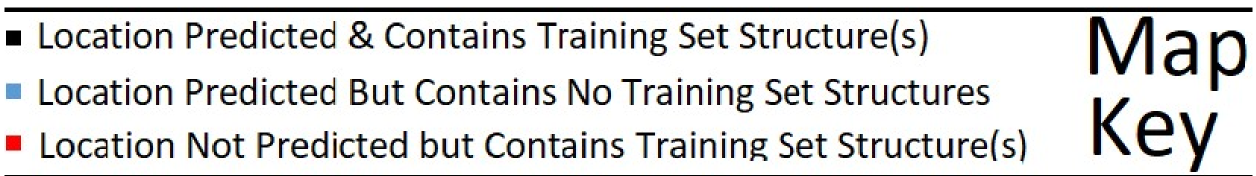
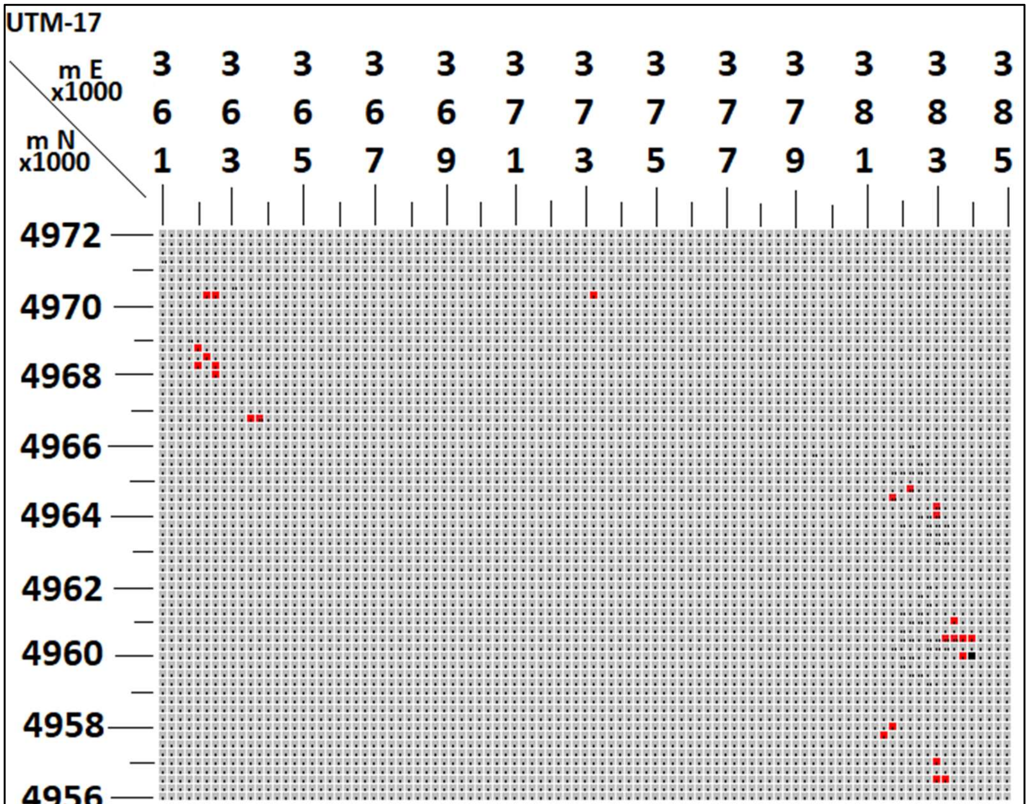
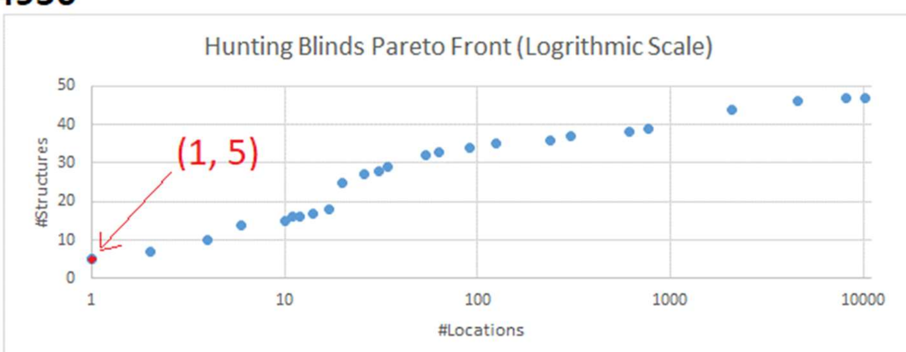


Figure 77: Map Key





Musthave Rules:

Dist to Water > 0m   Dist to Caribou < 3774.47m

Standard Rules:

|Height Above/Below Caribou| < 15.12m

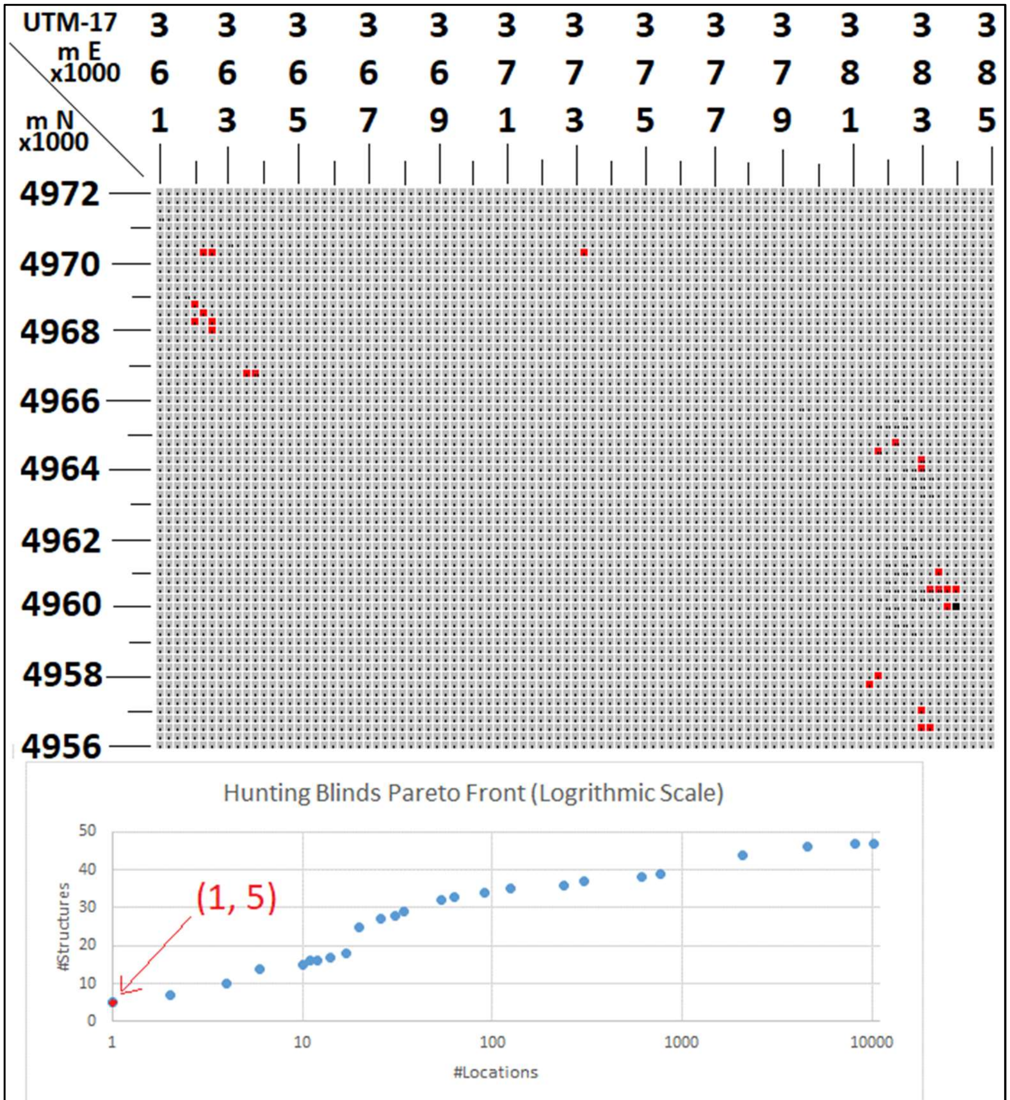
Dist to Training Set Drive Line < 159.79m

Vegetation Percentage > 75.9%

Dist to T. Set Log Camp < 1669.81m

Timestep Threshold: 3   Prediction Threshold: 10

Figure 78: Hunting Blinds Frame (Locations, Structures) = (1, 5)



Musthave Rules:

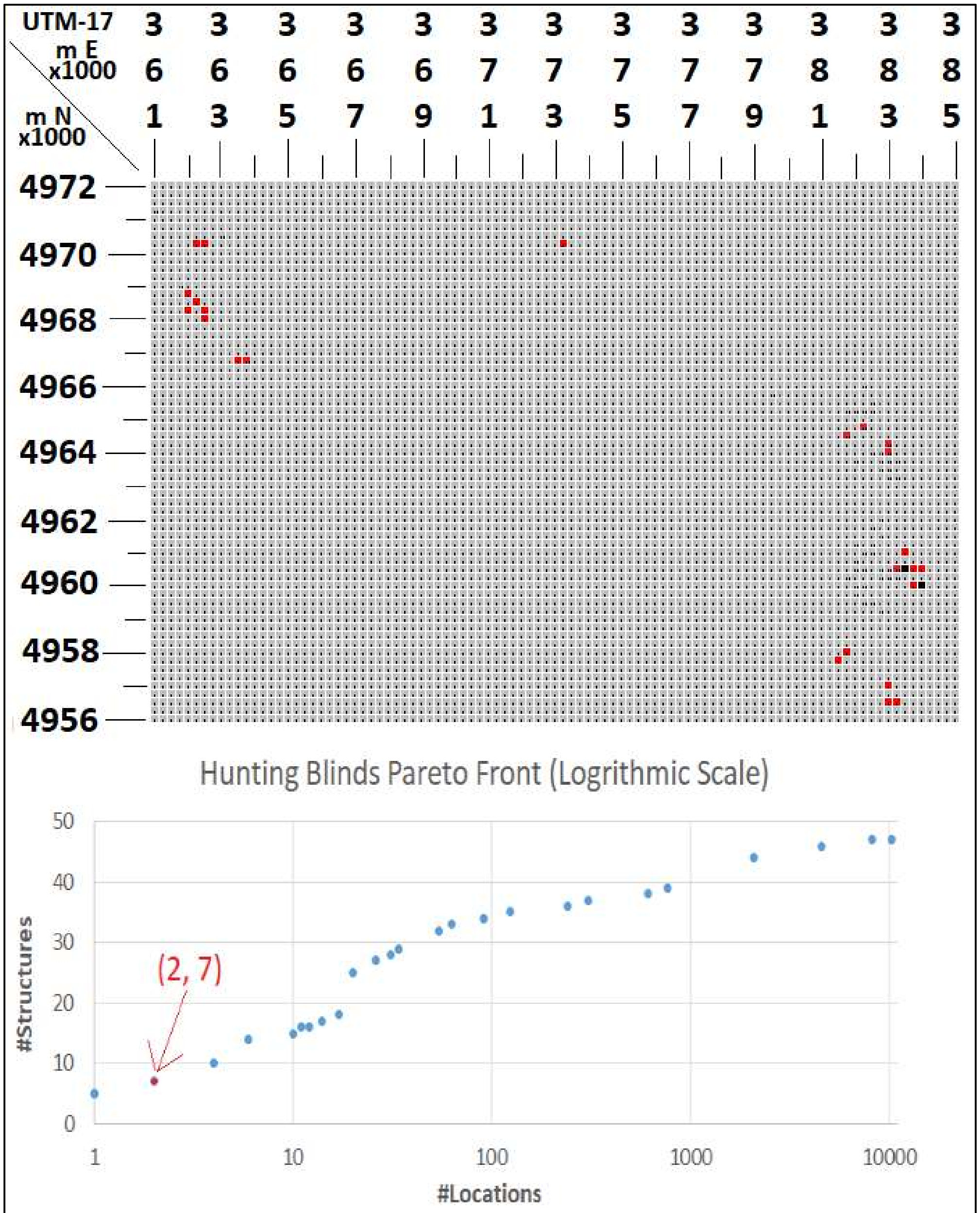
Dist to Water > 0m   Dist to Caribou < 3767.62m

Standard Rules:

|Height Above/Below Caribou| < 11.73m
Dist to Training Set Drive Line < 8.34m
Vegetation Percentage > 76.6%
Dist to T. Set Log Camp < 1080.33m

Timestep Threshold: 3   Prediction Threshold: 10

Figure 79: Hunting Blinds Frame (Locations, Structures) = (1, 5) #2



Musthave Rules:

Dist to Water > 0m   Dist to Caribou < 3995.01m

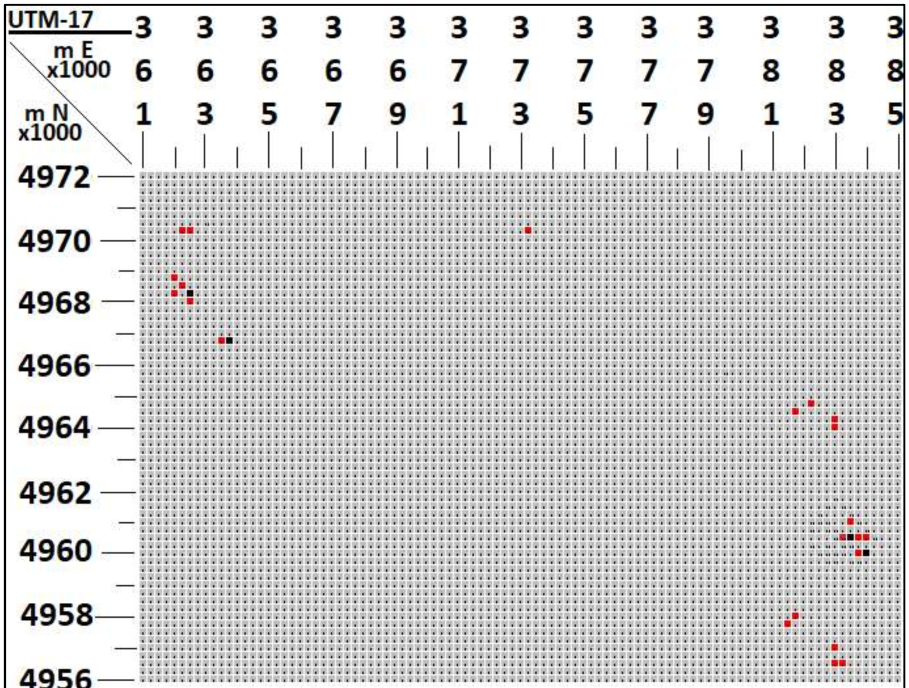
Standard Rules:

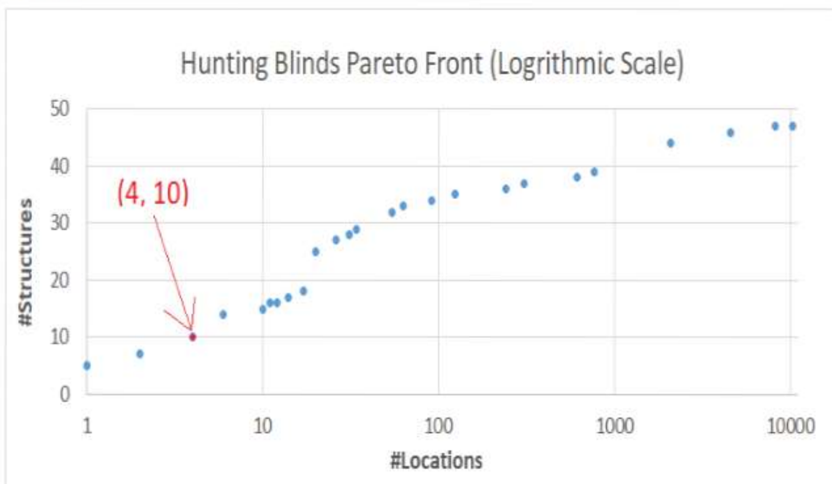
|Height Above/Below Caribou| < 7.59m   Vegetation Percentage > 41.51%

Dist to Training Set Drive Line < 83.15m   Dist to T. Set Log Camp < 1827.39m

Timestep Threshold: 3   Prediction Threshold: 9

Figure 80: Hunting Blinds Frame (Locations, Structures) = (2, 7)





Musthave Rules:

Dist to Water > 0m   Dist to Caribou < 2746.39m

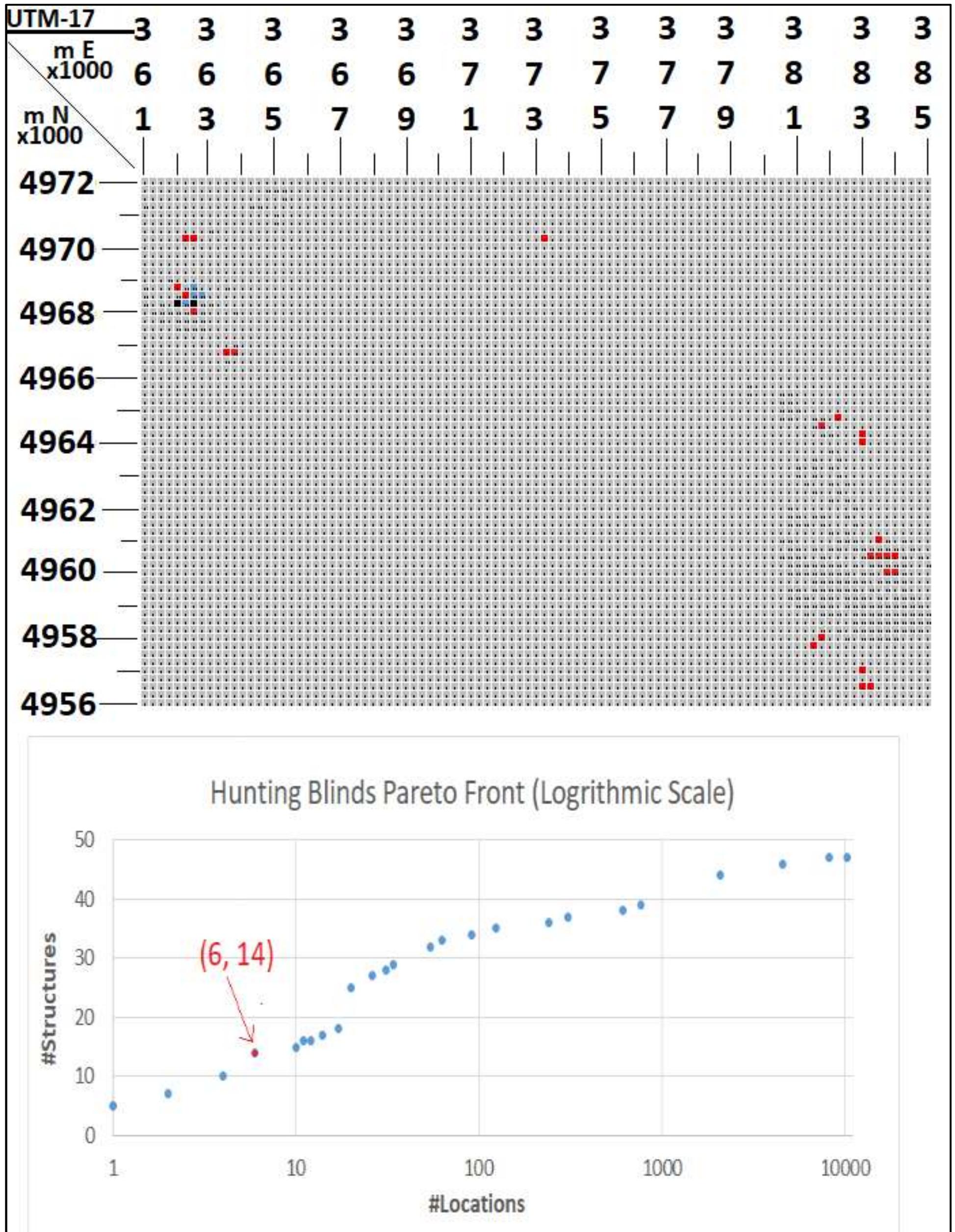
Standard Rules:

|Height Above/Below Caribou| < 11.56m   Vegetation Percentage > 42.55%

Dist to Training Set Drive Line < 174.77m   Dist to T. Set Log Camp < 1171.70m

Timestep Threshold: 3   Prediction Threshold: 6

Figure 81: Hunting Blinds Frame (Locations, Structures) = (4, 10)



Musthave Rules:

Dist to Water > 0m Dist to Caribou < 2818.98m

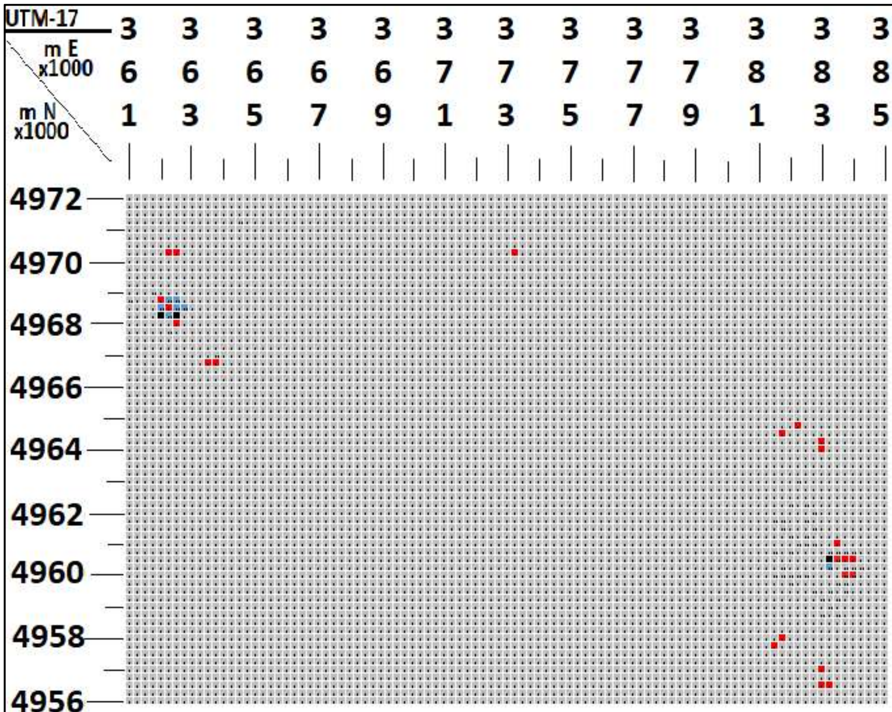
Standard Rules:

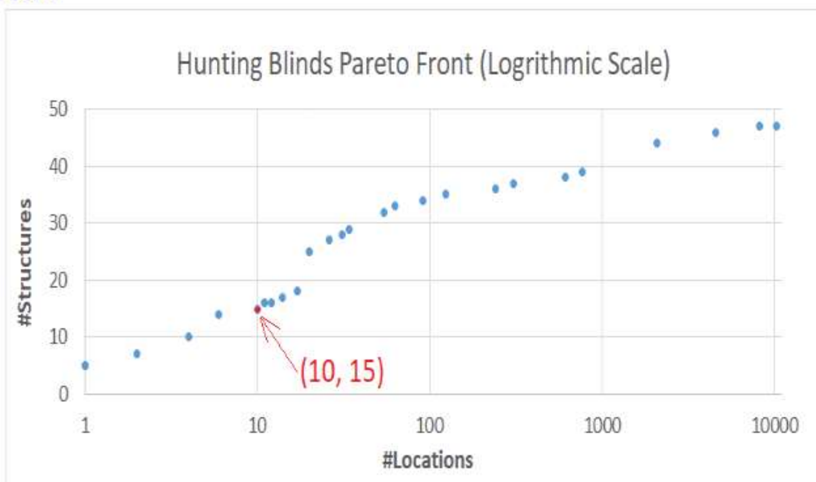
|Height Above/Below Caribou| < 9.54m Vegetation Percentage > 53.20%

Dist to Training Set Drive Line < 467.54m Dist to T. Set Log Camp < 2920.38m

Timestep Threshold: 3 Prediction Threshold: 11

Figure 82: Hunting Blinds Frame (Locations, Structures) = (6, 14)





Musthave Rules:

Dist to Water > 0m   Dist to Caribou < 2546.11m

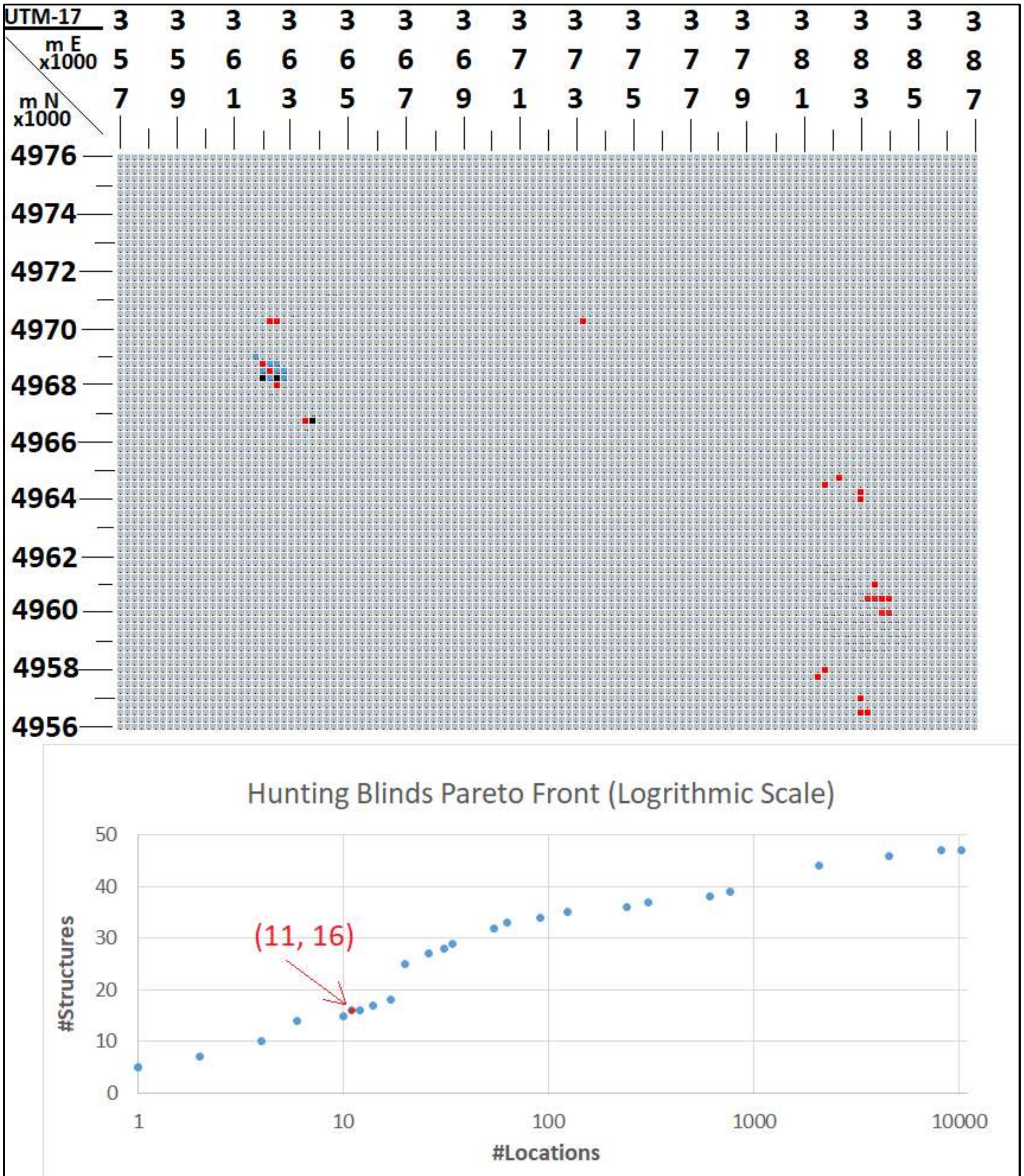
Standard Rules:

|Height Above/Below Caribou| < 2.18m   Vegetation Percentage > 64.07%

Dist to Training Set Drive Line < 474.92m   Dist to T. Set Log Camp < 2057.45m

Timestep Threshold: 3   Prediction Threshold: 6

Figure 83: Hunting Blinds Frame (Locations, Structures) = (10, 15)



Musthave Rules:

Dist to Water > 0m   Dist to Caribou < 1749.16m

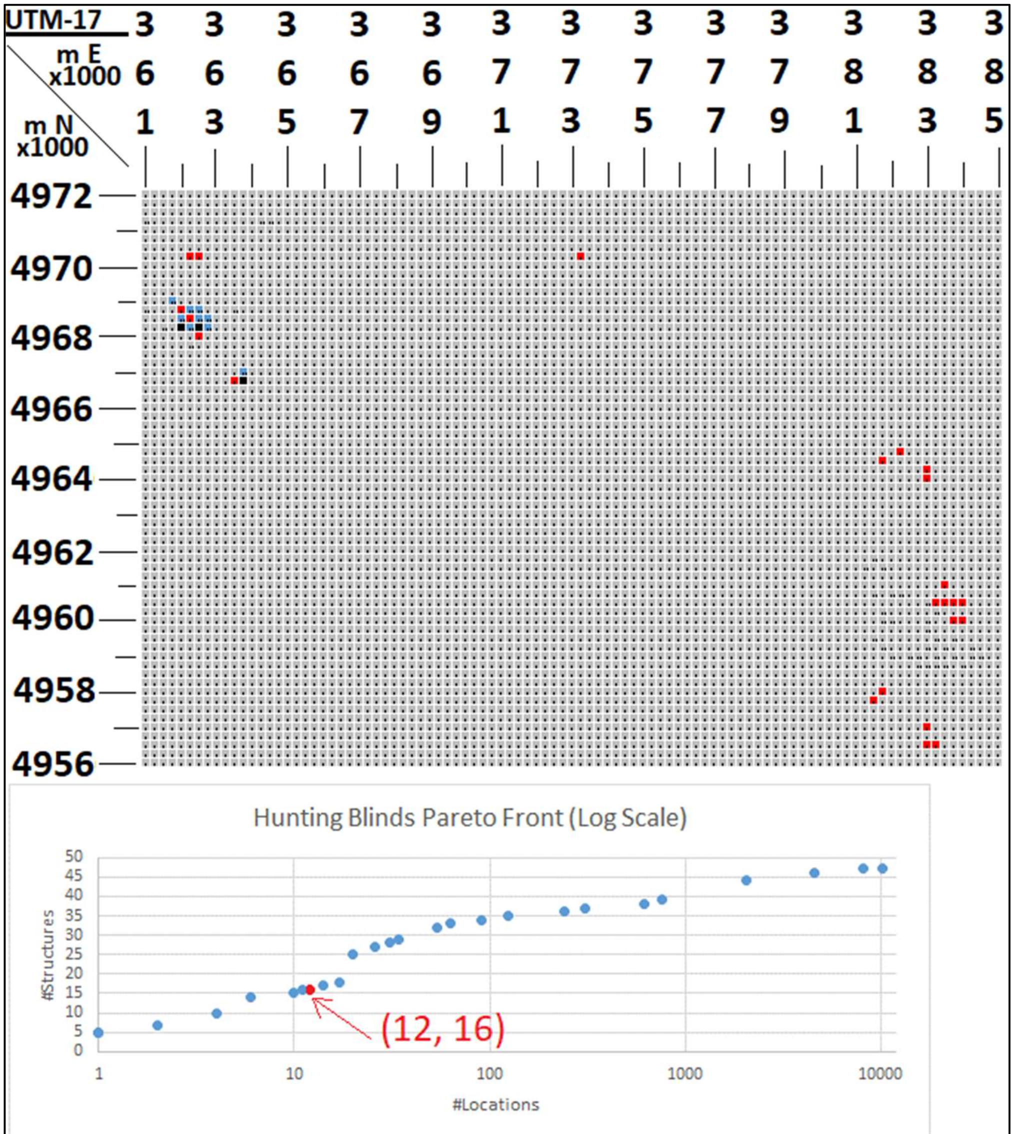
Standard Rules:

|Height Above/Below Caribou| < 19.09m Vegetation Percentage > 22.74%

Dist to Training Set Drive Line < 364.50m Dist to T. Set Log Camp < 2107.43m

Timestep Threshold: 3   Prediction Threshold: 5

Figure 84: Hunting Blinds Frame (Locations, Structures) = (11, 16)



Musthave Rules:

Dist to Water > 0m   Dist to Caribou < 1284.64m

Standard Rules:

|Height Above/Below Caribou| < 7.69m

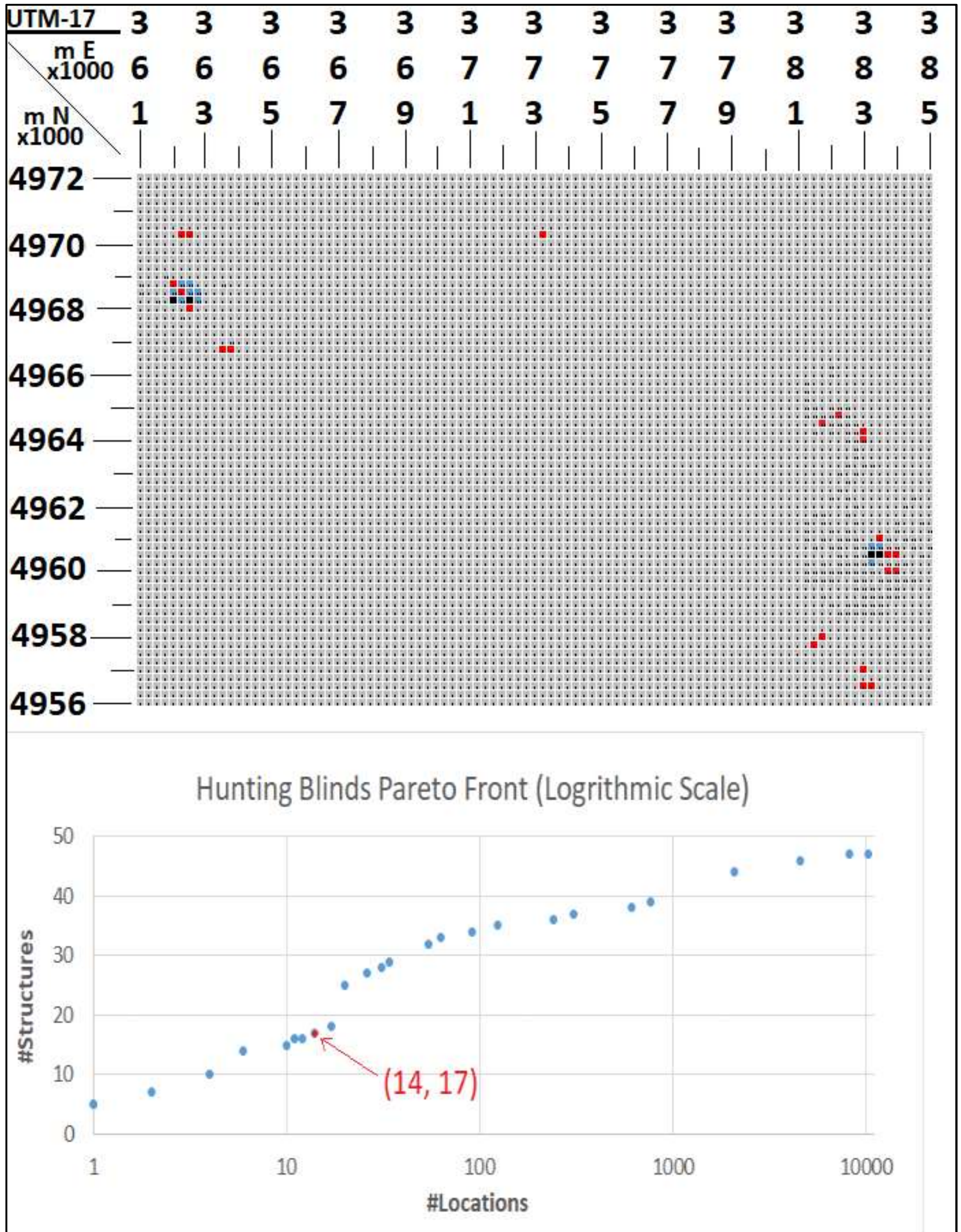
Dist to Training Set Drive Line < 460.10m

Vegetation Percentage > 57 %

Dist to Training Set Log Camp < 2199.27m

Timestep Threshold: 3   Prediction Threshold: 4

Figure 85: Hunting Blinds Frame (Locations, Structures) = (12, 16)



Musthave Rules:

Dist to Water > 0m   Dist to Caribou < 3558.71m

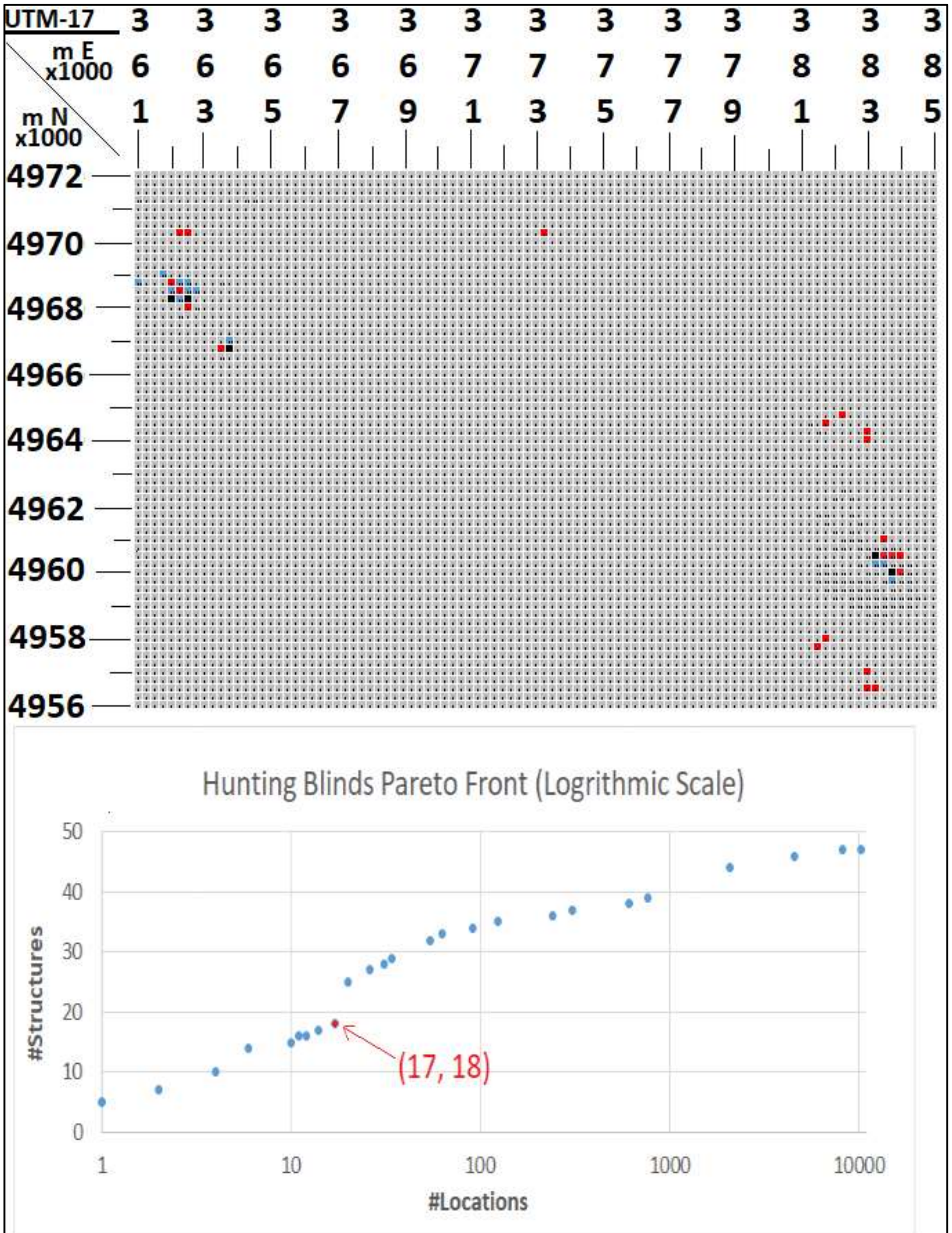
Standard Rules:

|Height Above/Below Caribou| < 2.53m   Vegetation Percentage > 30.29%

Dist to Training Set Drive Line < 479.74m   Dist to T. Set Log Camp < 2238.11m

Timestep Threshold: 3   Prediction Threshold: 6

Figure 86: Hunting Blinds Frame (Locations, Structures) = (14, 17)



Musthave Rules:

Dist to Water > 0m   Dist to Caribou < 2311.58m

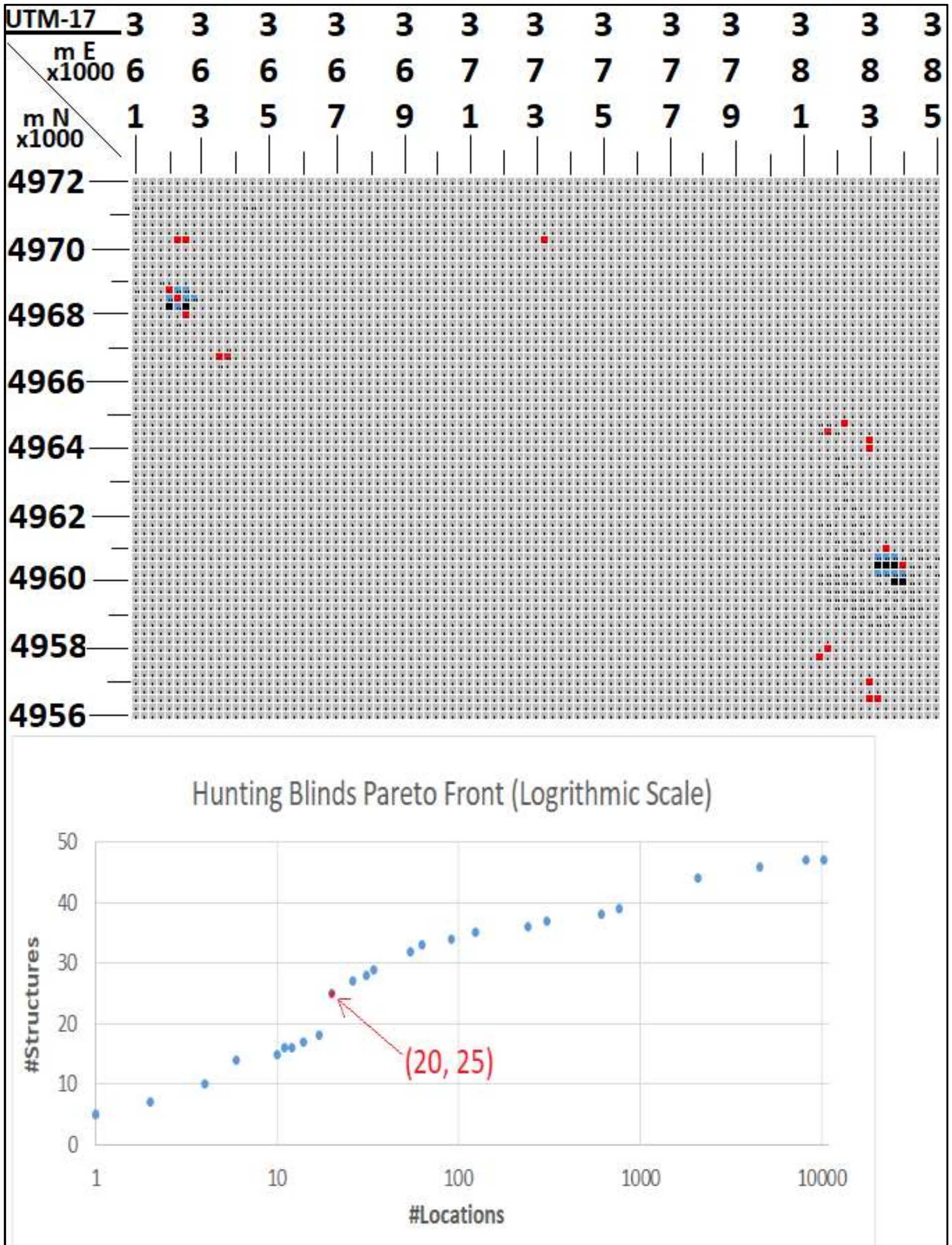
Standard Rules:

|Height Above/Below Caribou| < 7.20m   Vegetation Percentage > 9.32%

Dist to Training Set Drive Line < 399.66m Dist to T. Set Log Camp < 2027.06m

Timestep Threshold: 3   Prediction Threshold: 4

Figure 87: Hunting Blinds Frame (Locations, Structures) = (17, 18)



Musthave Rules:

Dist to Water > 0m   Dist to Caribou < 2919.07m

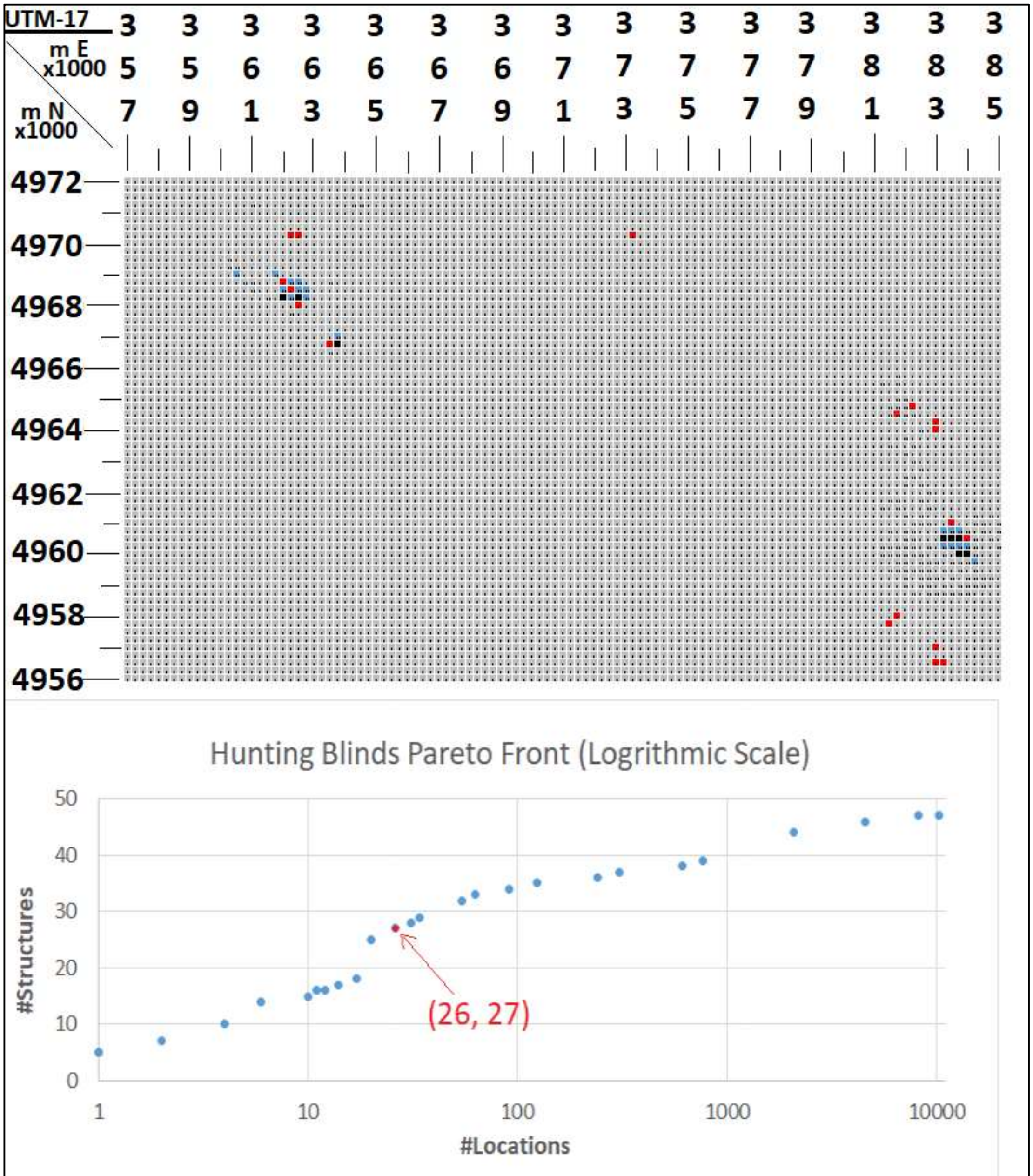
Standard Rules:

|Height Above/Below Caribou| < 6.06m   Vegetation Percentage > 59.23%

Dist to Training Set Drive Line < 452.62m   Dist to T. Set Log Camp < 2026.84m

Timestep Threshold: 3   Prediction Threshold: 6

Figure 88: Hunting Blinds Frame (Locations, Structures) = (20, 25)



Musthave Rules:

Dist to Water > 0m   Dist to Caribou < 3059.79m

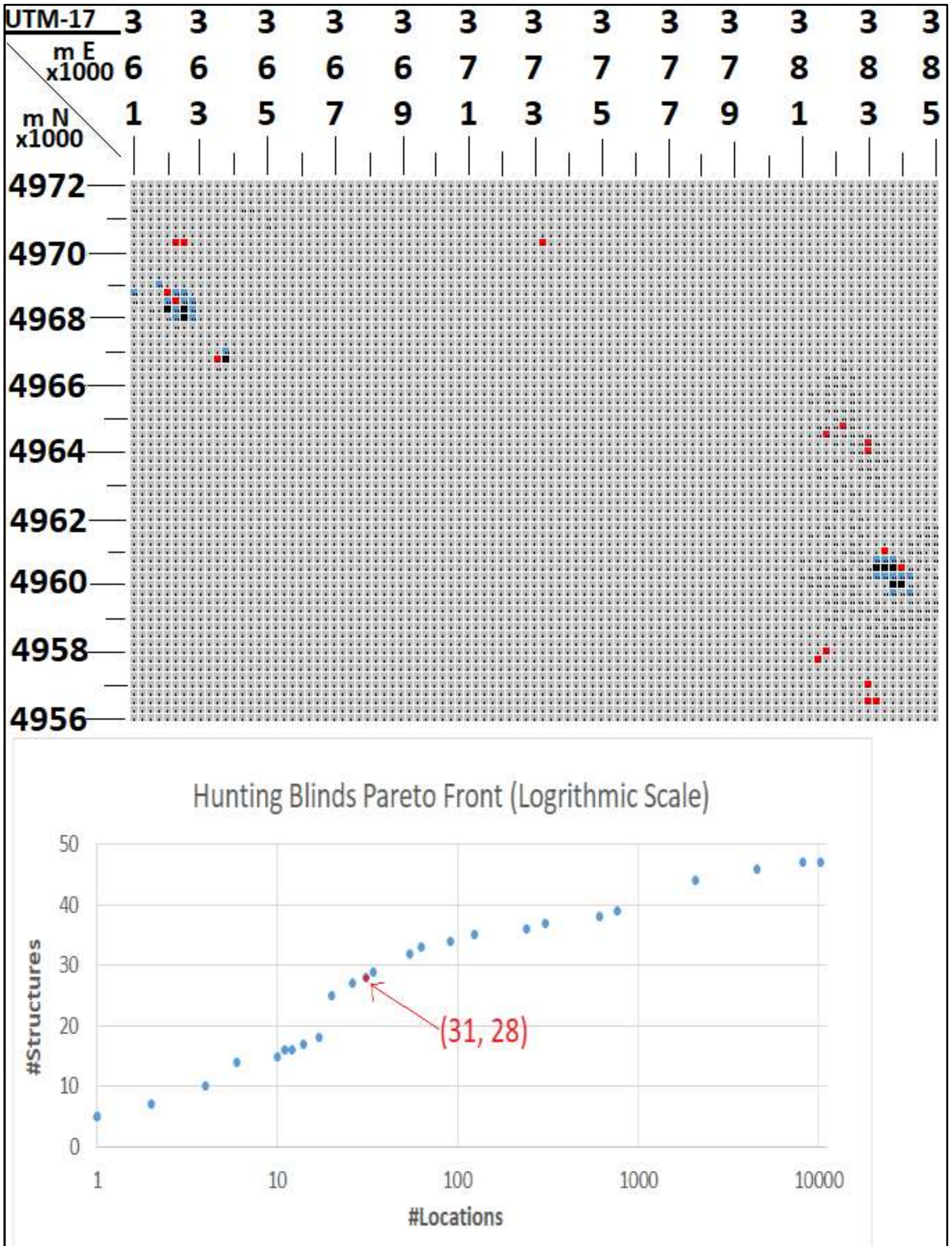
Standard Rules:

|Height Above/Below Caribou| < 6.90m   Vegetation Percentage > 59.85%

Dist to Training Set Drive Line < 397.79m   Dist to T. Set Log Camp < 2242.21m

Timestep Threshold: 3   Prediction Threshold: 5

Figure 89: Hunting Blinds Frame (Locations, Structures) = (26, 27)



Musthave Rules:

Dist to Water > 0m   Dist to Caribou < 3884.13m

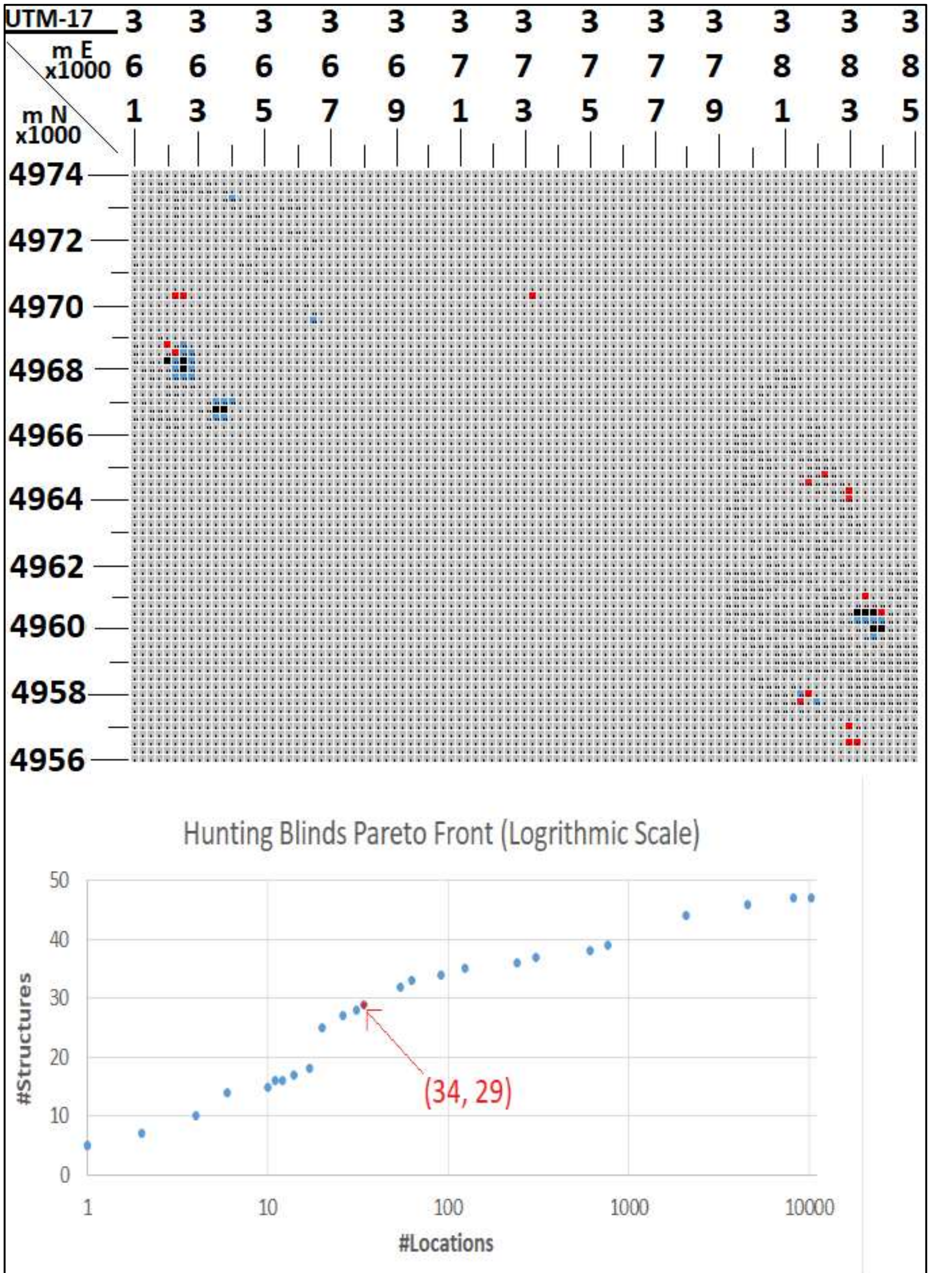
Standard Rules:

|Height Above/Below Caribou| < 7.36m   Vegetation Percentage > 64.77%

Dist to Training Set Drive Line < 453.64m   Dist to T. Set Log Camp < 2350.24m

Timestep Threshold: 3   Prediction Threshold: 4

Figure 90: Hunting Blinds Frame (Locations, Structures) = (31, 28)



Musthave Rules:

Dist to Water > 0m   Dist to Caribou < 3551.63m

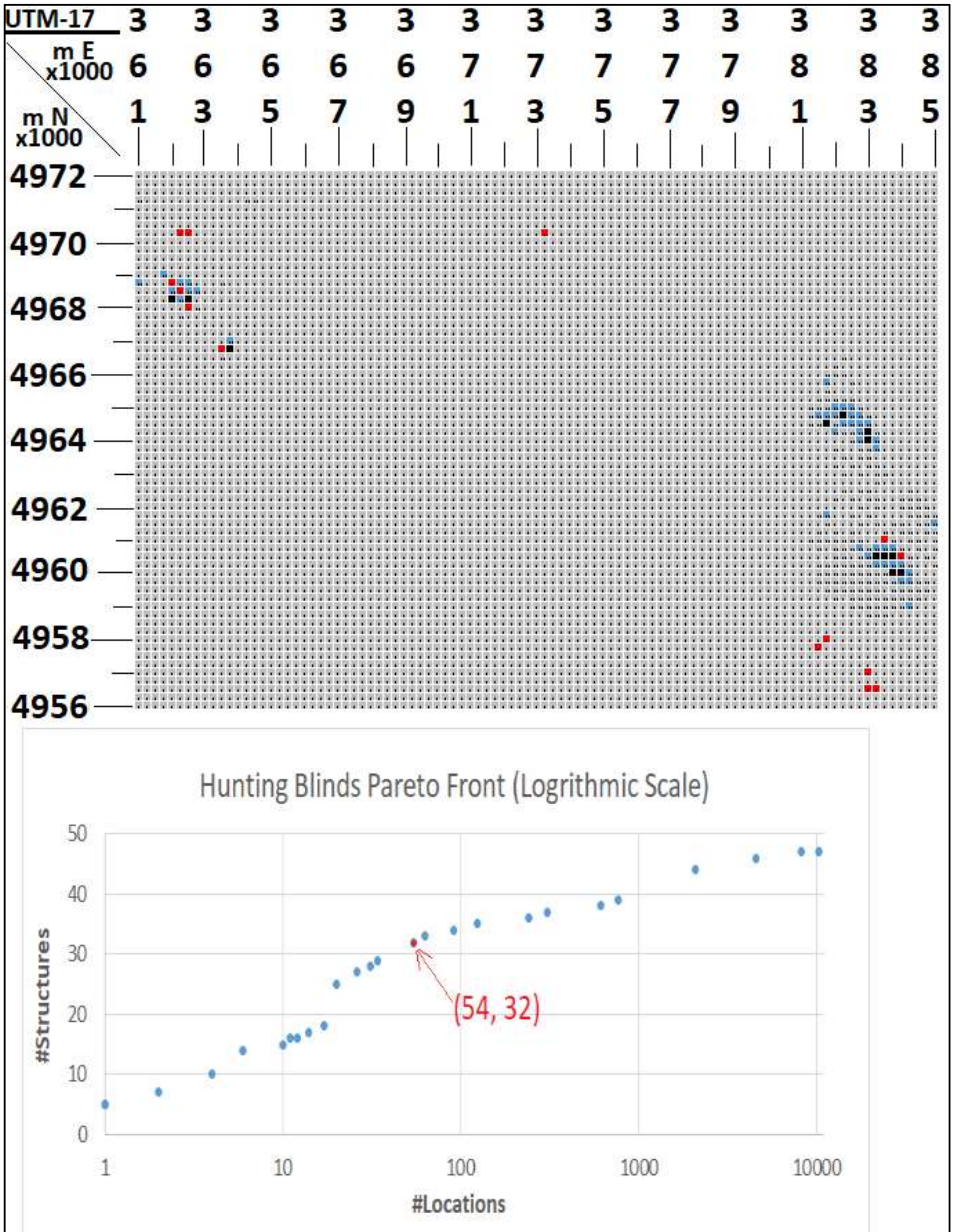
Standard Rules:

|Height Above/Below Caribou| < 7.85m   Vegetation Percentage > 59.67%

Dist to Training Set Drive Line < 401.00m   Dist to T. Set Log Camp < 4158.51m

Timestep Threshold: 3   Prediction Threshold: 8

Figure 91: Hunting Blinds Frame (Locations, Structures) = (34, 29)



Musthave Rules:

Dist to Water > 0m   Dist to Caribou < 3819.48m

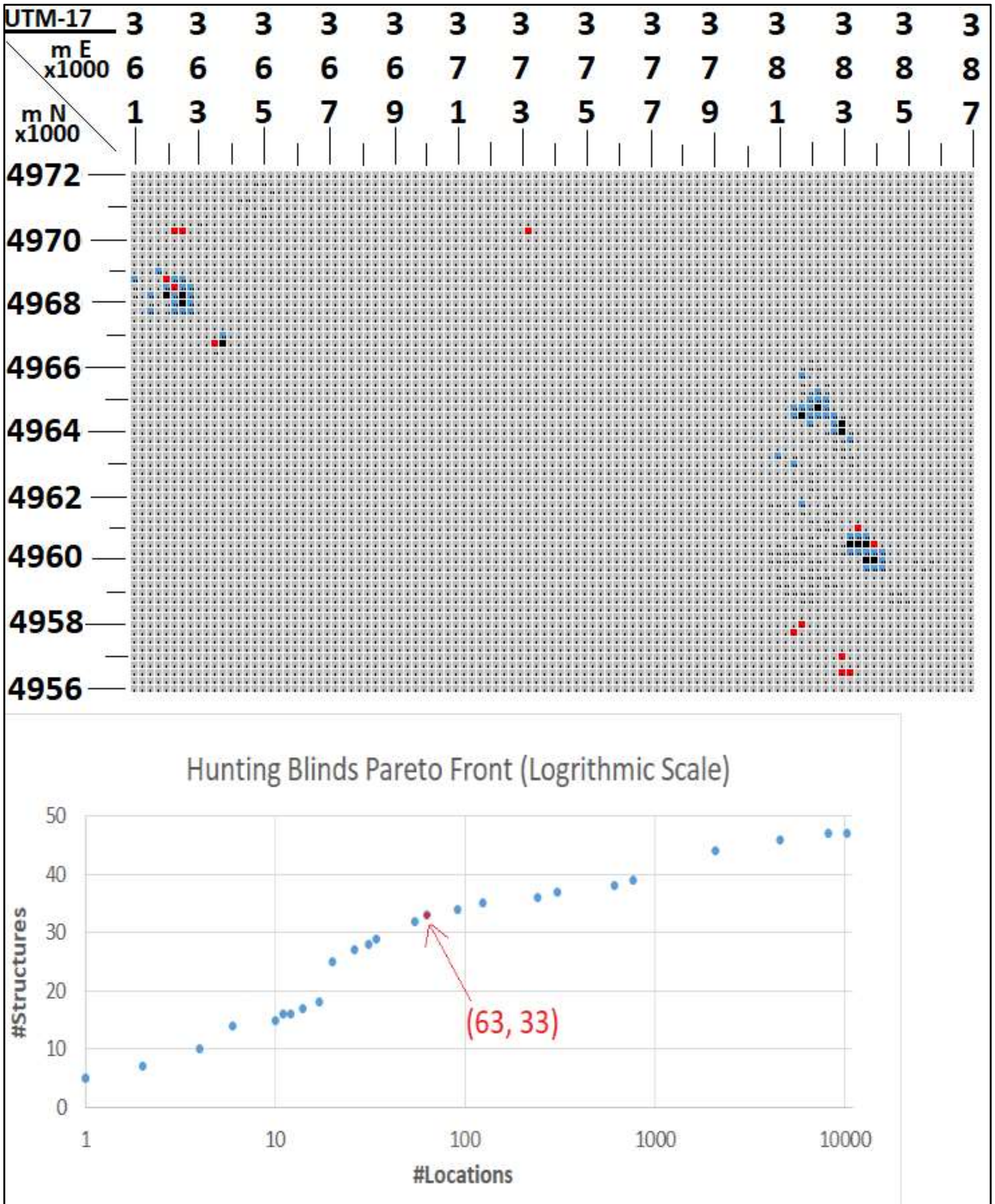
Standard Rules:

|Height Above/Below Caribou| < 6.43m   Vegetation Percentage > 36.64%

Dist to Training Set Drive Line < 422.93m   Dist to T. Set Log Camp < 2047.37m

Timestep Threshold: 3   Prediction Threshold: 3

Figure 92: Hunting Blinds Frame (Locations, Structures) = (54, 32)



Musthave Rules:

Dist to Water > 0m   Dist to Caribou < 3782.30m

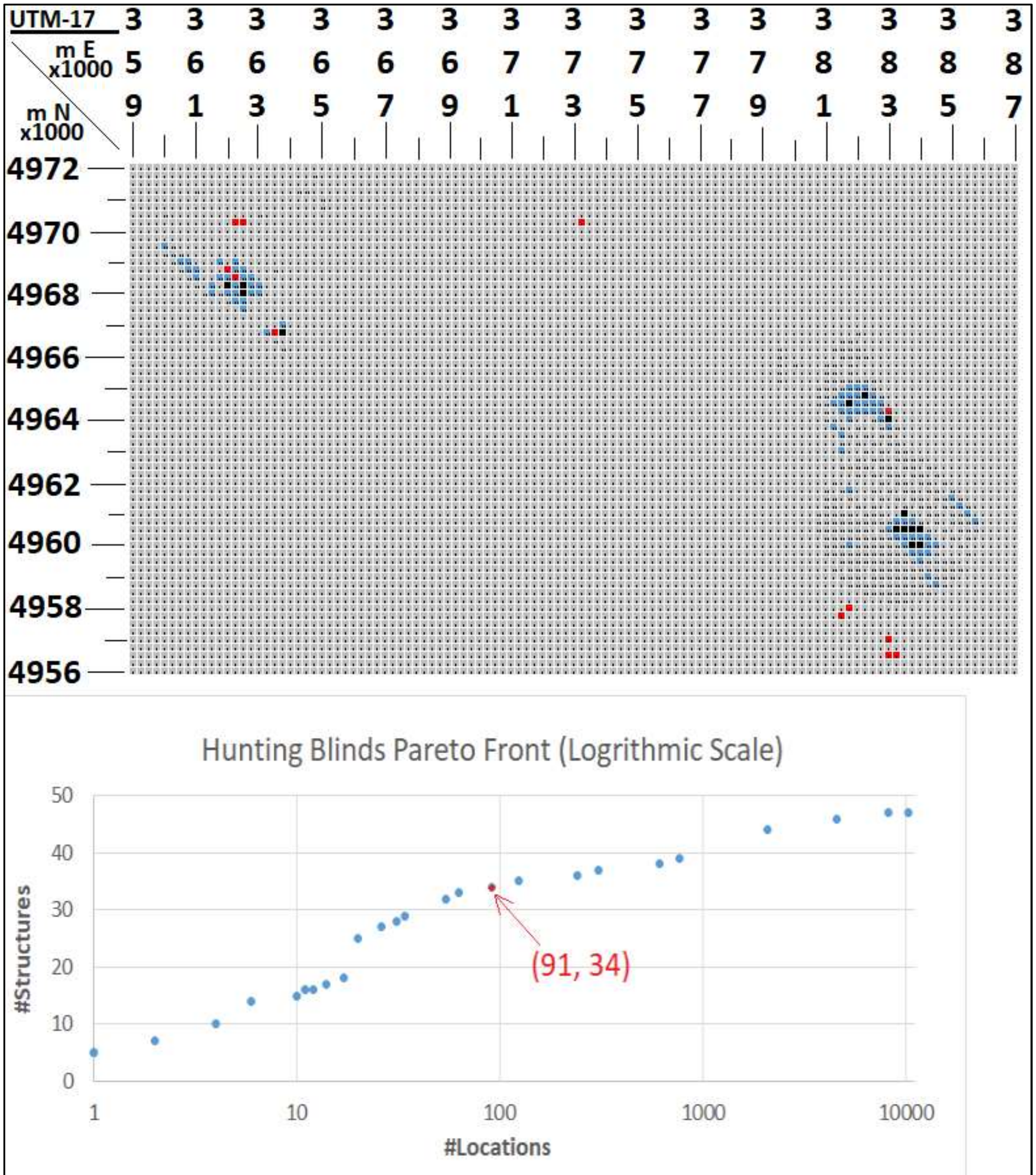
Standard Rules:

|Height Above/Below Caribou| < 8.43m   Vegetation Percentage > 76.80%

Dist to Training Set Drive Line < 354.56m   Dist to T. Set Log Camp < 2722.53m

Timestep Threshold: 3   Prediction Threshold: 3

Figure 93: Hunting Blinds Frame (Locations, Structures) = (63, 33)



Musthave Rules:

Dist to Watre > 0m Dist to Caribou < 3677.32m

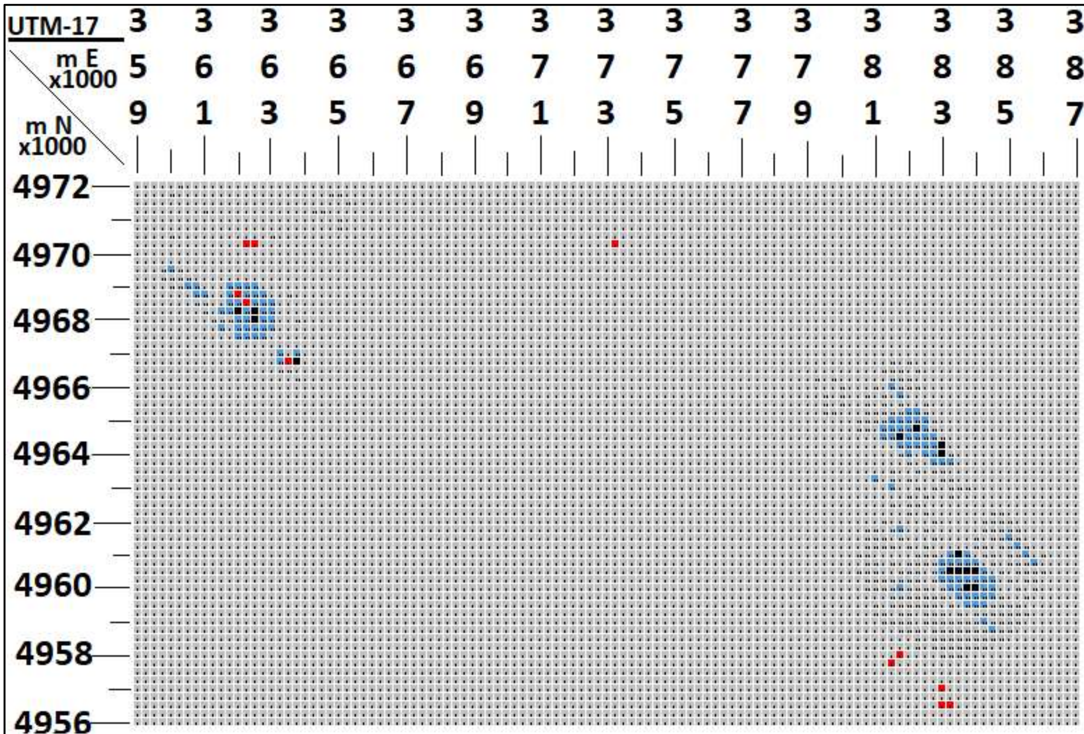
Standard Rules:

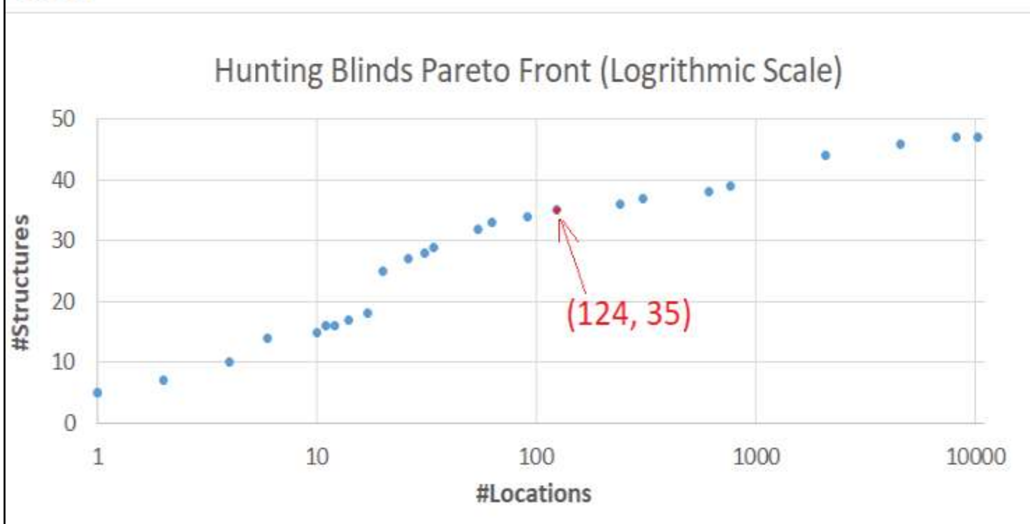
|Height Above/Below Caribou| < 14.47m Vegetation Percentage > 20.37%

Dist to Training Set Drive Line < 524.93m Dist to T. Set Log Camp < 2539.85m

Timestep Threshold: 3 Prediction Threshold: 3

Figure 94: Hunting Blinds Frame (Locations, Structures) = (91, 34)





Musthave Rules:

Dist to Water > 0m   Dist to Caribou < 3733.27m

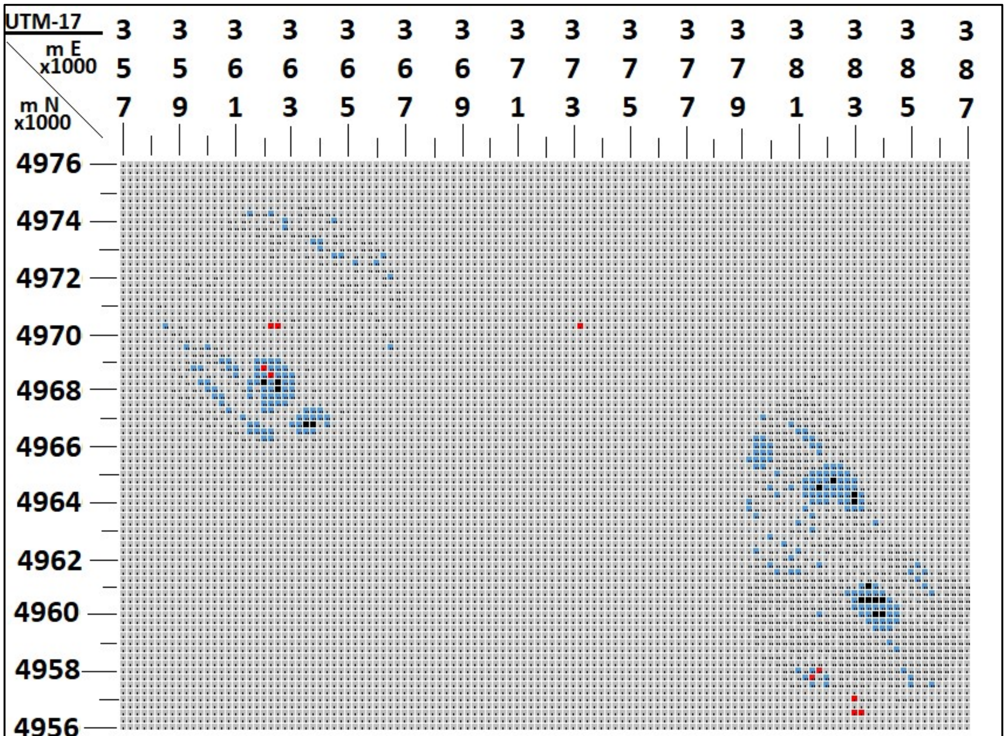
Standard Rules:

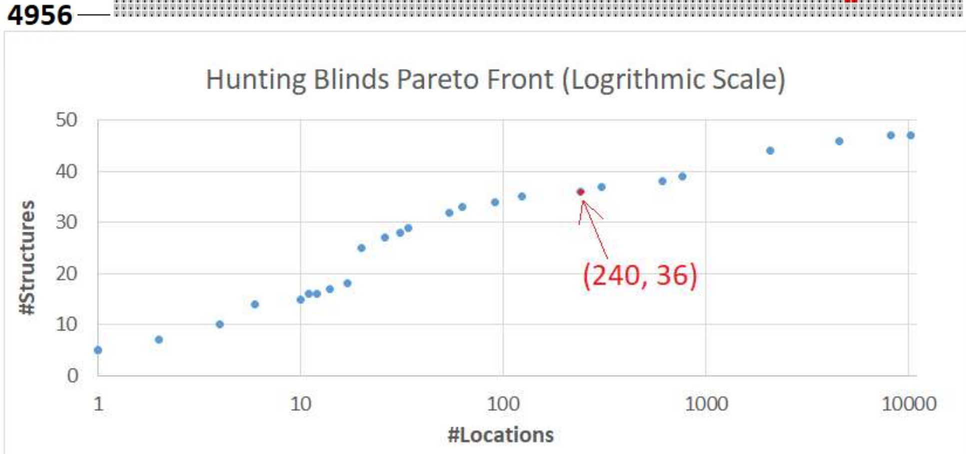
|Height Above/Below Caribou| < 9.10m   Vegetation Percentage > 50.81%

Dist to Training Set Drive Line < 688.28m   Dist to T. Set Log Camp < 2779.22m

Timestep Threshold: 3   Prediction Threshold: 3

Figure 95: Hunting Blinds Frame (Locations, Structures) = (124, 35)





Musthave Rules:

Dist to Water > 0m   Dist to Caribou < 3991.75m

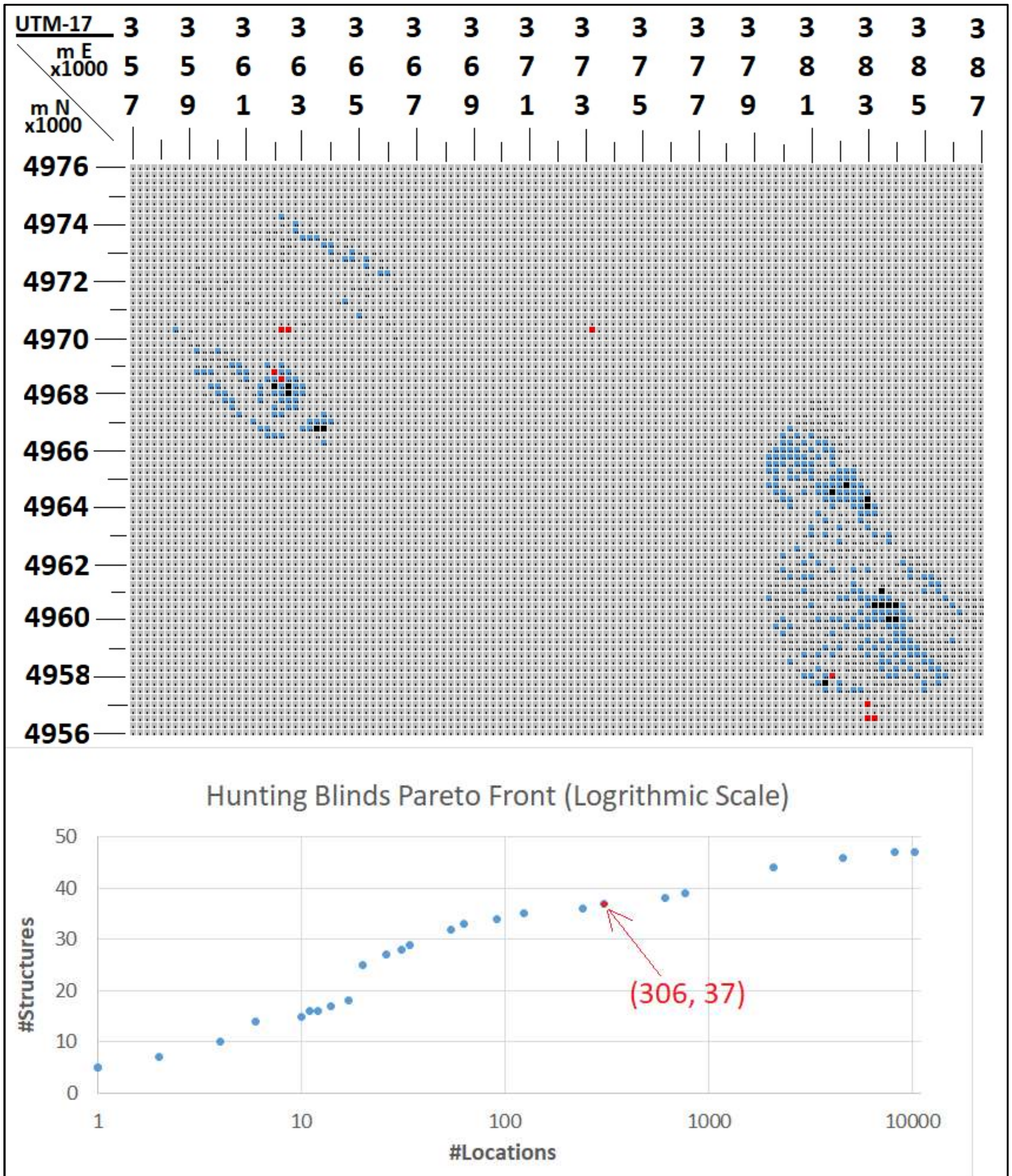
Standard Rules:

|Height Above/Below Caribou| < 15.10m Vegetation Percentage > 44.68%

Dist to Training Set Drive Line < 629.87m Dist to T. Set Log Camp < 4187.95m

Timestep Threshold: 3   Prediction Threshold: 3

Figure 96: Hunting Blinds Frame (Locations, Structures) = (240, 36)



Musthave Rules:

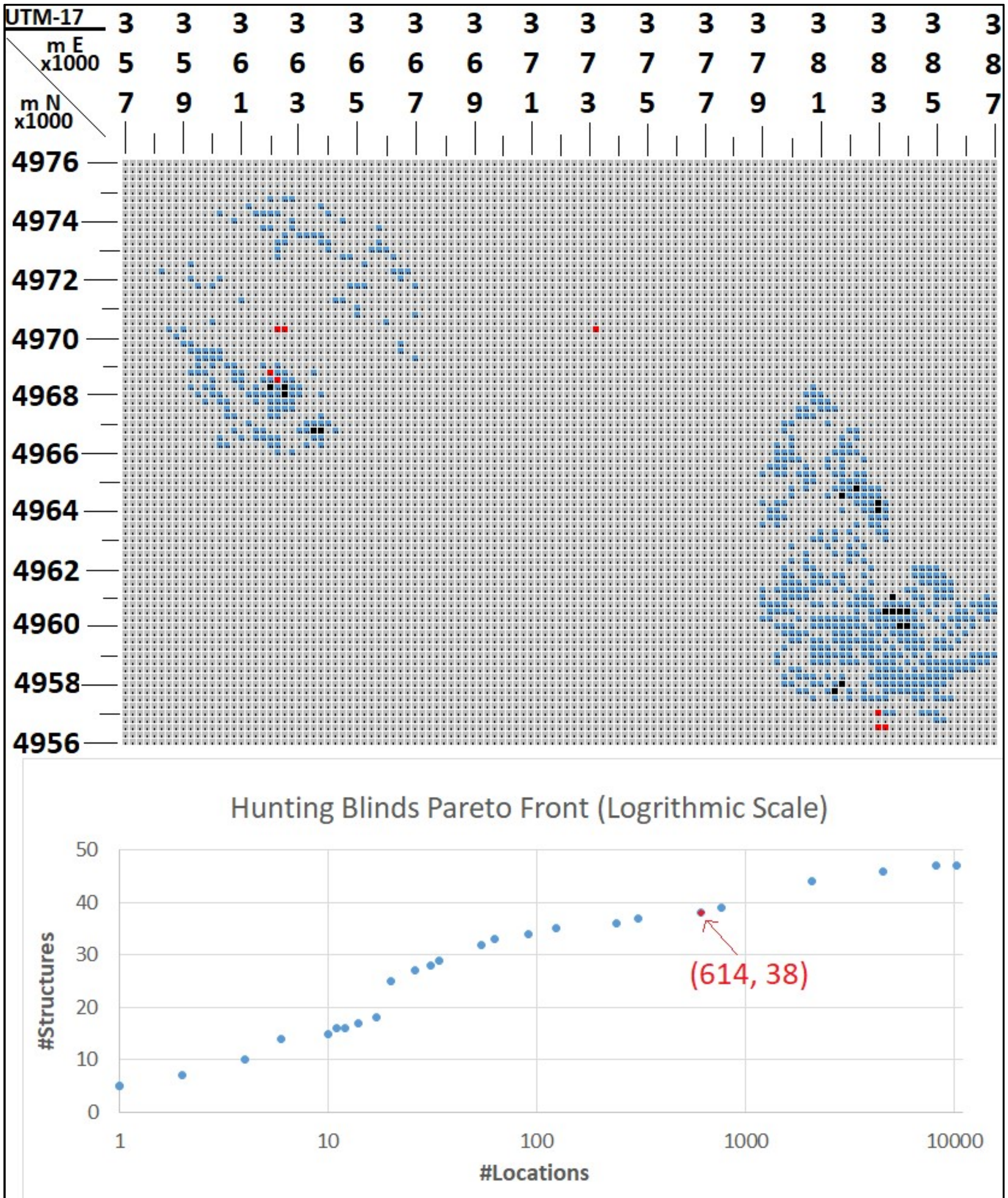
Dist to Water > 0m Dist to Caribou < 3773.66m

Standard Rules:

|Height Above/Below Caribou| < 15.04m Vegetation Percentage > 44.02%
Dist to Training Set Drive Line < 521.42m Dist to T. Set Log Camp < 4187.95m

Timestep Threshold: 3 Prediction Threshold: 2

Figure 97: Hunting Blinds Frame (Locations, Structures) = (306, 37)



Musthave Rules:

Dist to Water > 0m   Dist to Caribou < 3826.32m

Standard Rules:

|Height Above/Below Caribou| < 5.68m

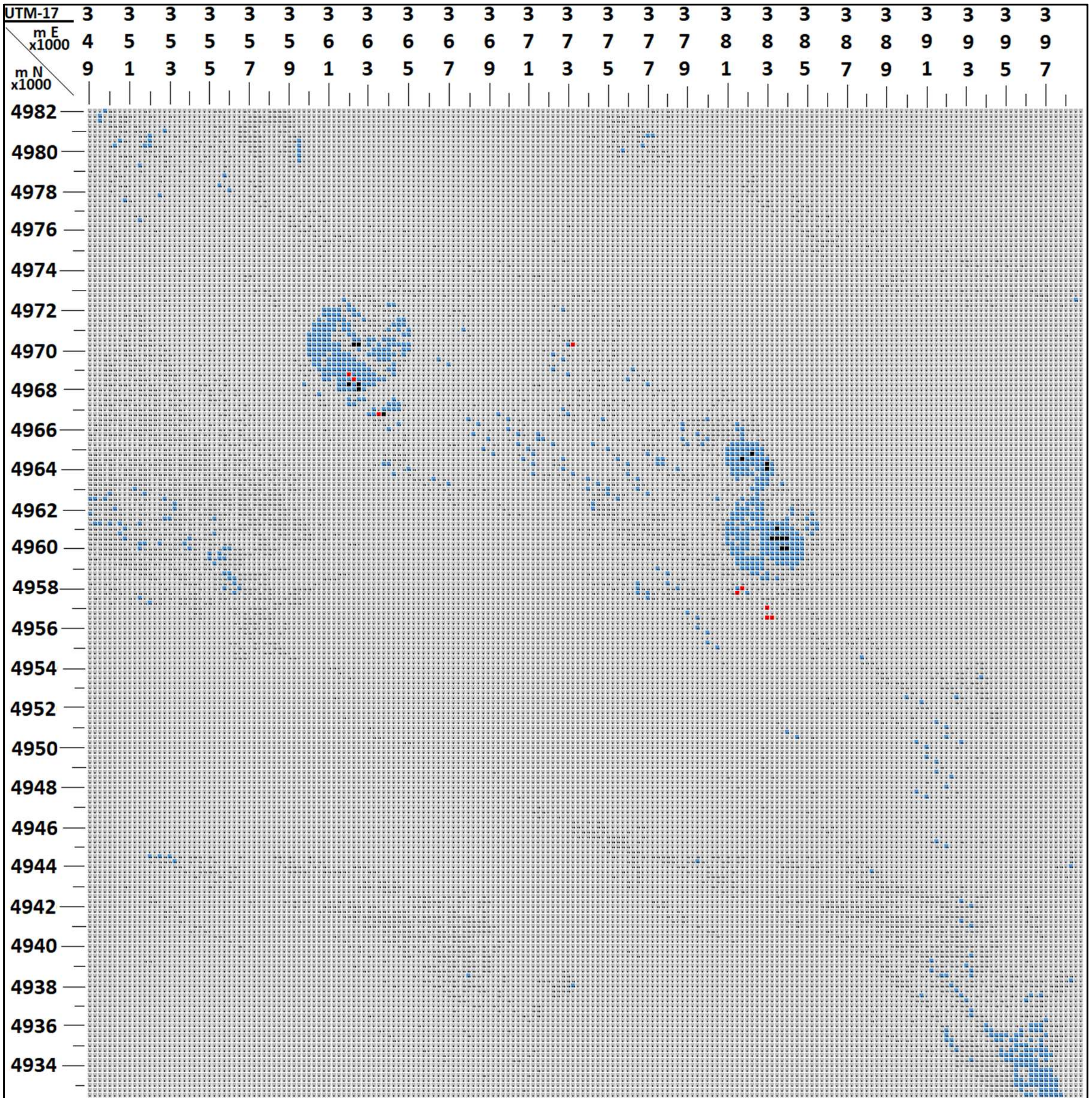
Dist to Training Set Drive Line < 531.26m

Vegetation Percentage > 56.60%

Dist to T. Set Log Camp < 4479.55m

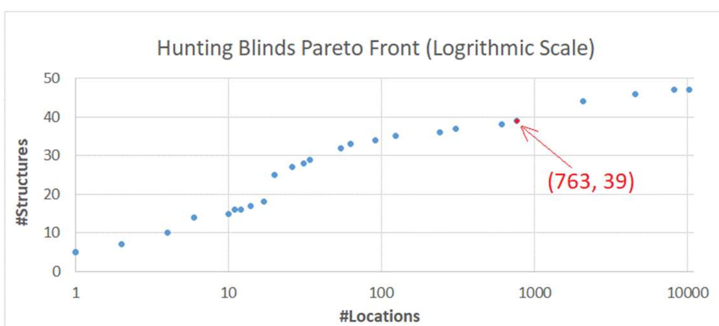
Timestep Threshold: 3   Prediction Threshold: 1

Figure 98: Hunting Blinds Frame (Locations, Structures) = (614, 38)



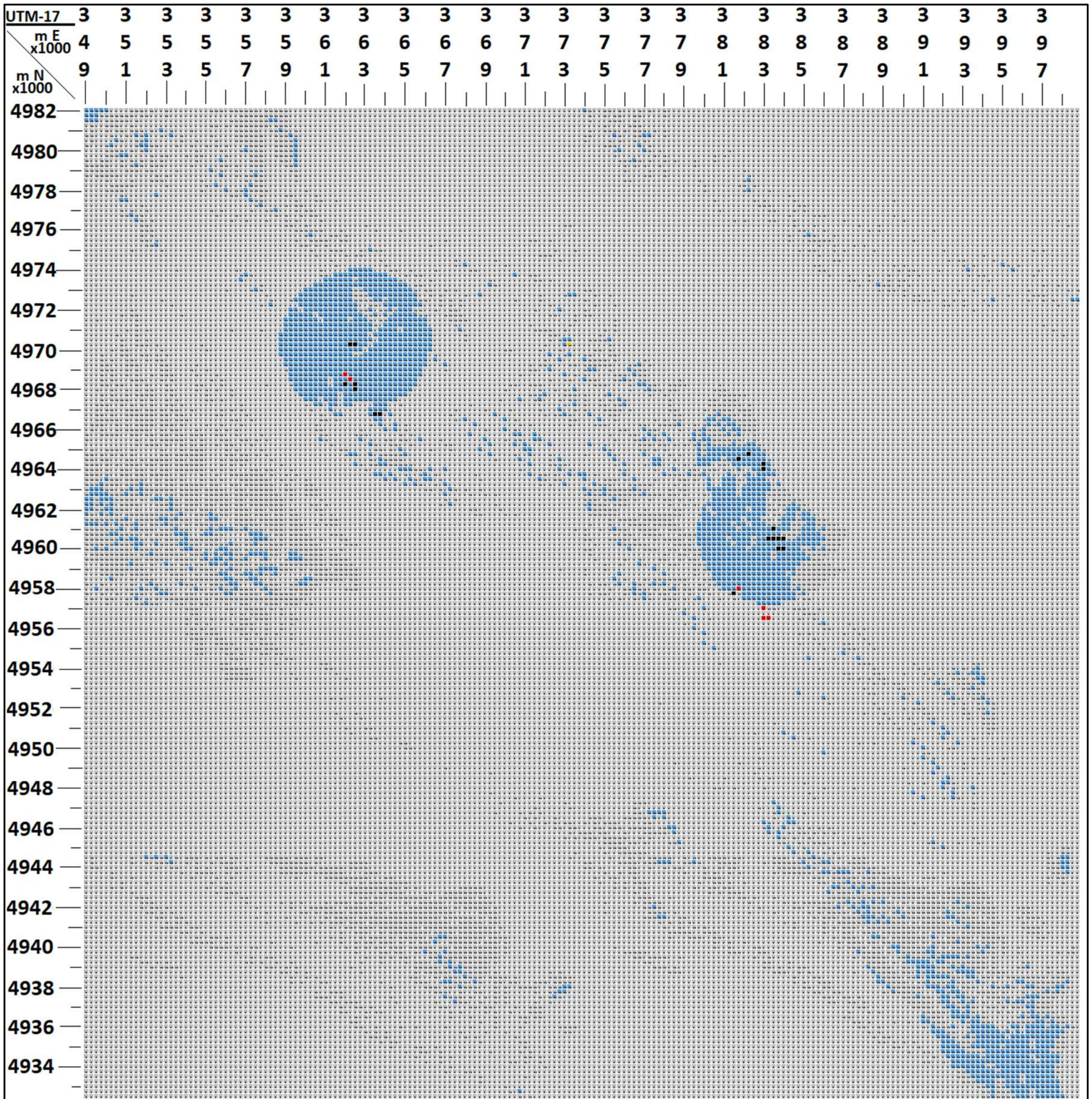
Musthave Rules:
Dist to Water > 0m
Dist to Caribou < 3914.95m

Standard Rules:
|Height Above/Below Caribou| < 1.41m
Vegatation Percentage > 45.30%
Dist to Training Set Drive Line < 963.12m
Dist to Training Set Log Camp < 2310.22m



Timestep Threshold: 2 **Prediction Threshold: 3**

Figure 99: Hunting Blinds Frame (Locations, Structures) = (763, 39)



Musthave Rules

Dist to Water > 0m

Dist to Caribou < 3972.58m

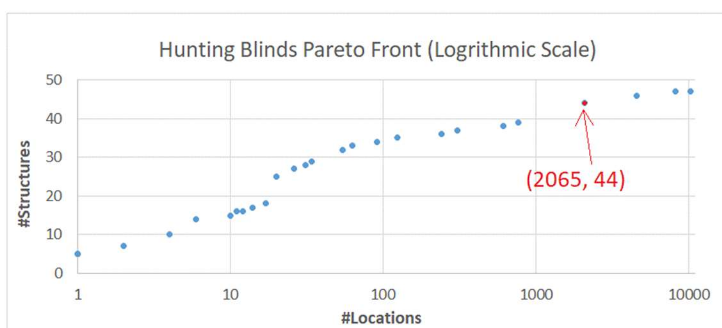
Standard Rules

|Height Above/Below Caribou| < 15.10m

Vegetation Percentage > 28.95%

Dist to Training Set Drive Line < 538.89m

Dist to Training Set Log Camp < 3557.65m



Timestep Threshold: 2 Prediction Threshold: 3

Figure 100: Hunting Blinds Frame (Locations, Structures) = (2065, 44)

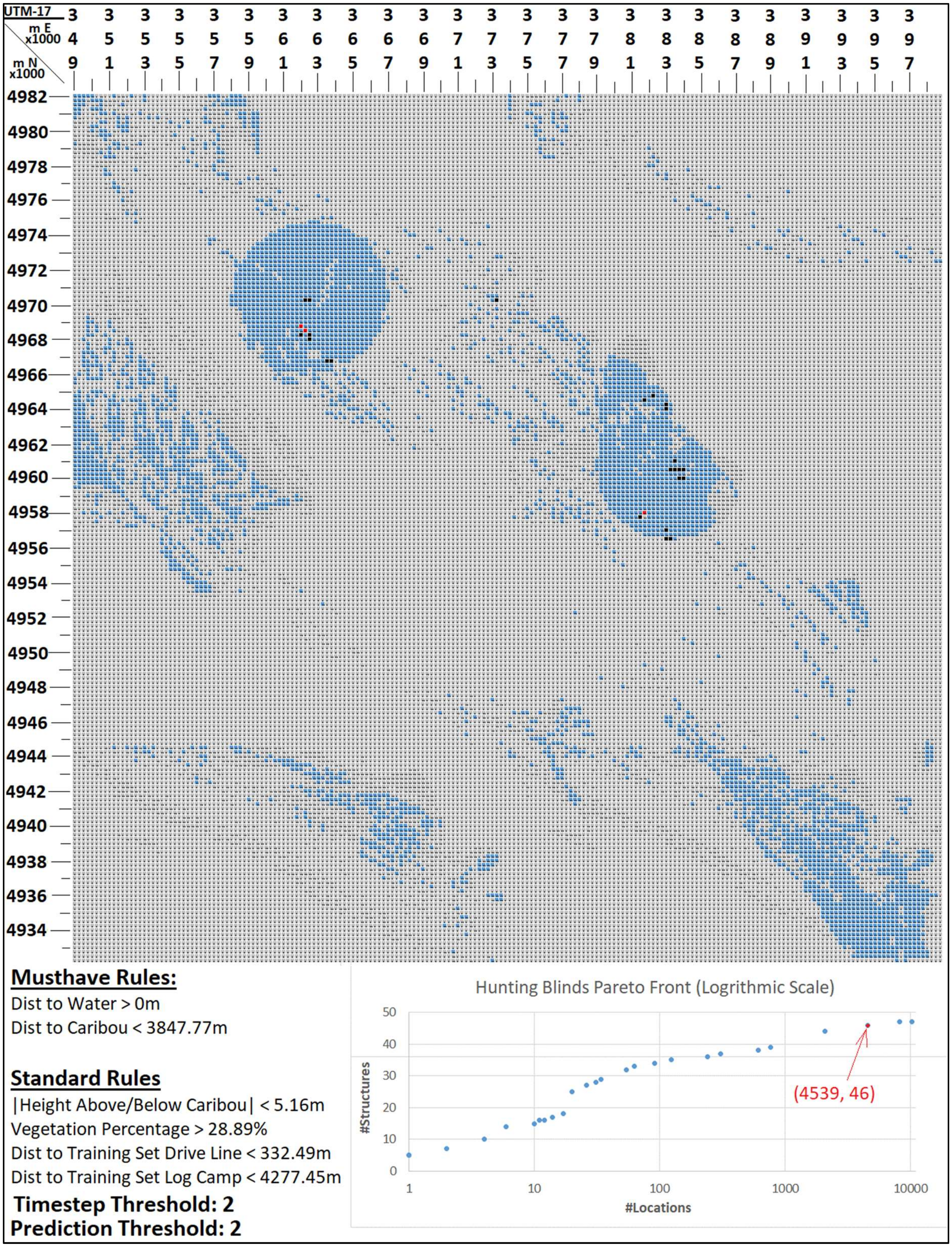
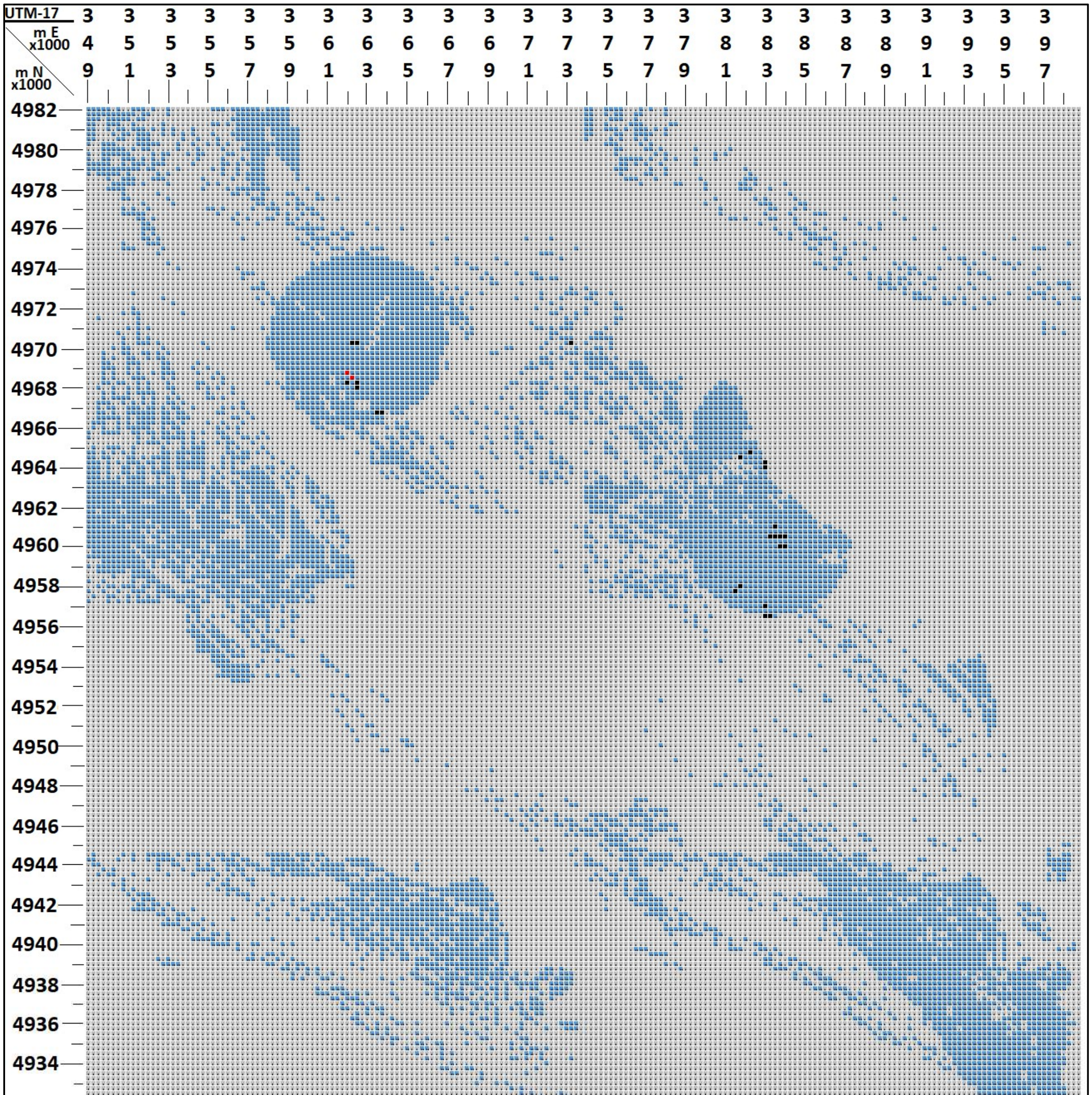


Figure 101: Hunting Blinds Frame (Locations, Structures) = (4539, 46)



Musthave Rules:

Dist to Water > 0m
Dist to Caribou < 3609.20m

Standard Rules:

|Height Above/Below Caribou| < 5.94m
Vegetation Percentage > 34.36%
Dist to Training Set Drive Line < 63.50m
Dist to Training Set Log Camp < 4261.34

Timestep Threshold: 2
Prediction Threshold: 1

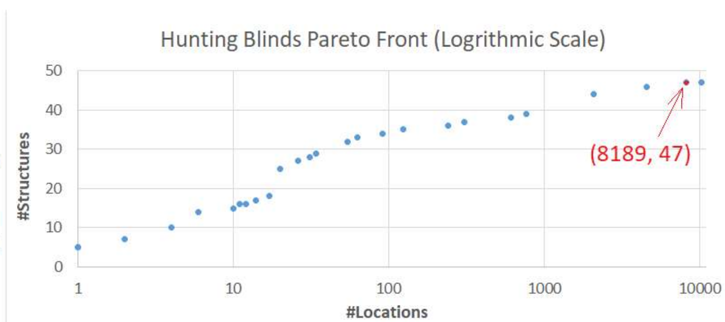
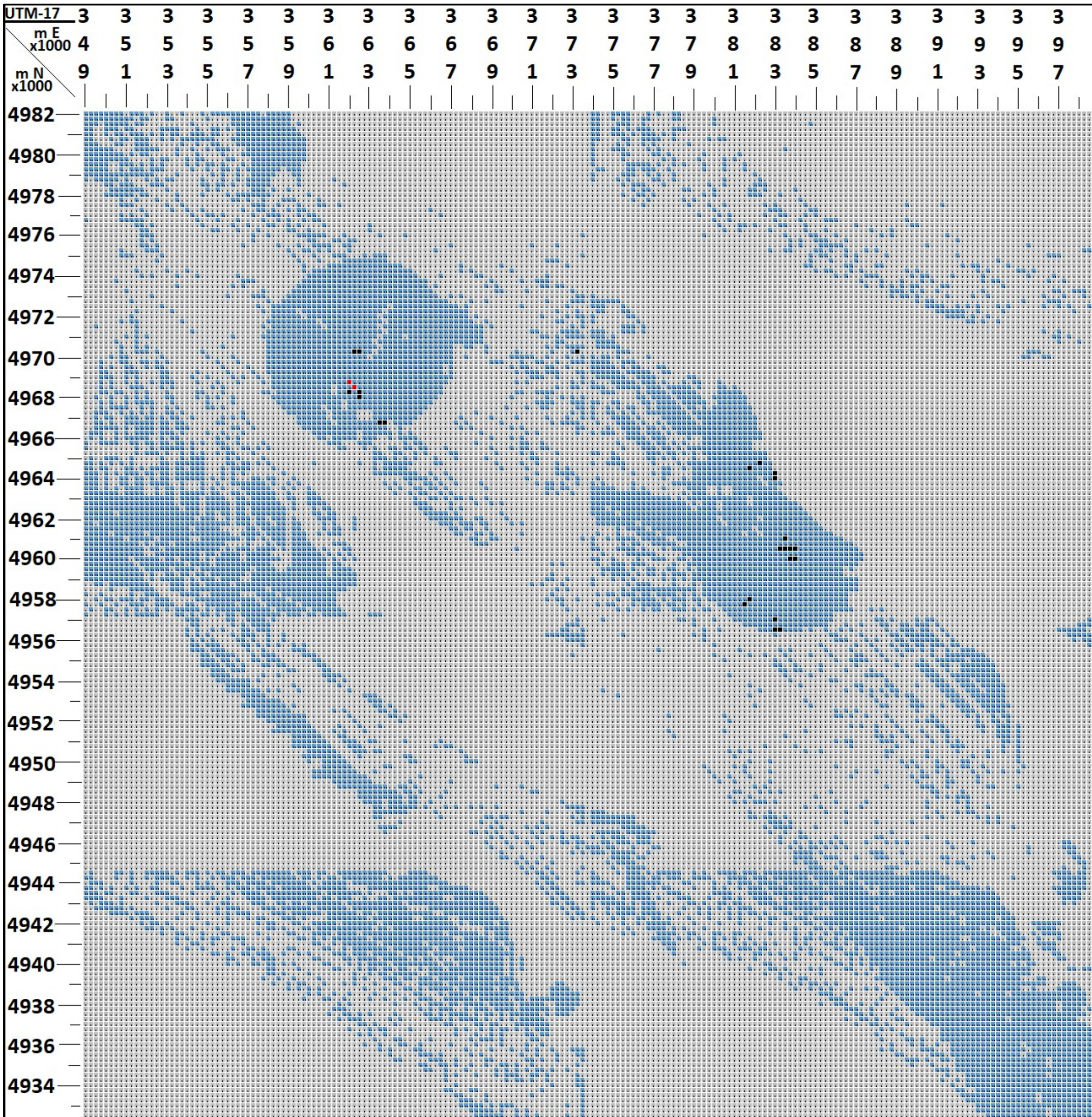


Figure 102: Hunting Blinds Frame (Locations, Structures) = (8189, 47)



Musthave Rules:

Dist to Water > 0m
Dist to Caribou < 3665.22m

Standard Rules

|Height Above/Below Caribou| < 15.07m
Vegetation Percentage > 41.57%
Dist to Training Set Drive Line < 700.91m
Dist to Training Set Log Camp < 4376.57m

Timestep Threshold: 2
Prediction Threshold: 1

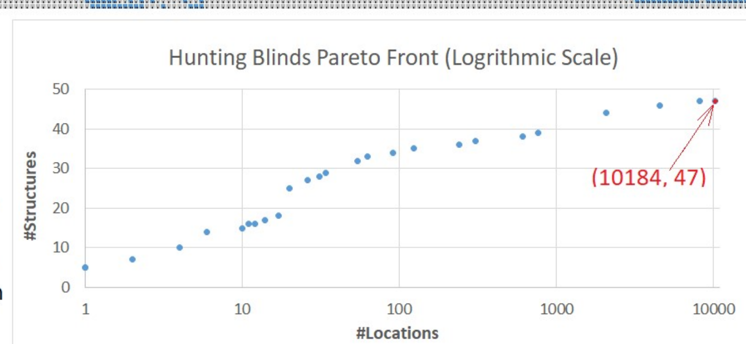


Figure 103: Hunting Blinds Frame (Locations, Structures) = (10184, 47)

8.2 Drive Lines

8.2.1 CAPSO Output for Drive Lines Structure Type

Table 14 contains the Pareto Front along with corresponding parameter values for the Drive Line structure type. It took 19.833 hours for CAPSO to produce these results.

Locations	Structures	Rule Thresholds:	(Musthave) Dist to Caribou <	(Standard) Dist to T. Set Hunting Blind <	(Standard) Dist to T. Set Logistical Camp <	Timestep Threshold	Prediction Threshold
3	2	Rule Thresholds:	864.8535	202.763782	2317.5791	2	5
4	3	Rule Thresholds:	2021.395	229.069223	2492.8265	2	10
8	4	Rule Thresholds:	1660.73	64.4937703	5416.432	2	8
9	6	Rule Thresholds:	5859.082	247.879092	2451.948	2	11
13	7	Rule Thresholds:	2099.277	195.377581	4047.5302	2	3
14	8	Rule Thresholds:	3474.944	95.420956	2817.5381	2	1
16	9	Rule Thresholds:	5724.179	217.618003	2568.0025	2	6
20	10	Rule Thresholds:	3735.675	148.299762	3847.7952	2	3
88	10	Rule Thresholds:	4319.658	285.992888	6520.9502	2	3
312	11	Rule Thresholds:	6366.46	320.81607	1296.3617	1	6
1489	12	Rule Thresholds:	5231.485	2828.69317	4196.2472	2	3

Table 14: CAPSO's Output – Drive Line Structure Type

8.2.2 Pareto Front, Learning Curve, and Knowledge Source Dominance Graphs

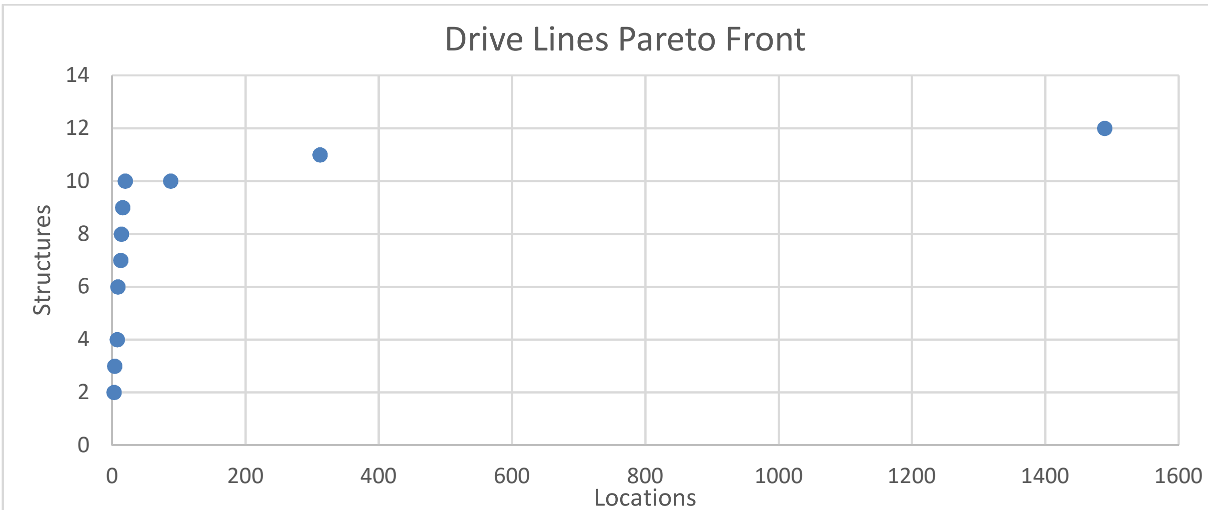


Figure 104: Drive Lines Pareto Front

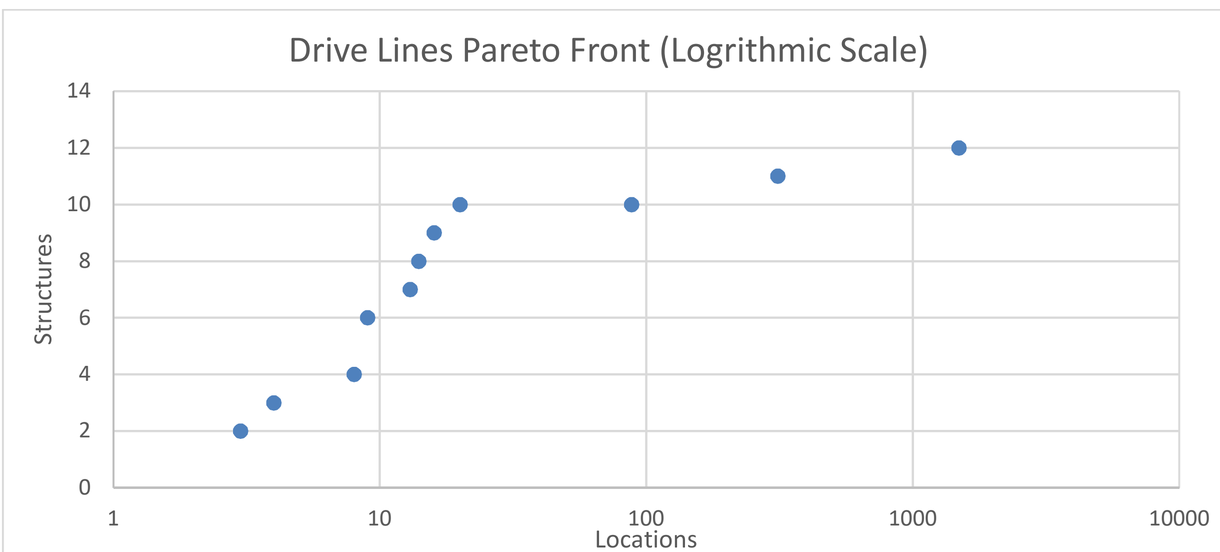


Figure 105: Drive Lines Pareto Front (Logrithmic Scale)

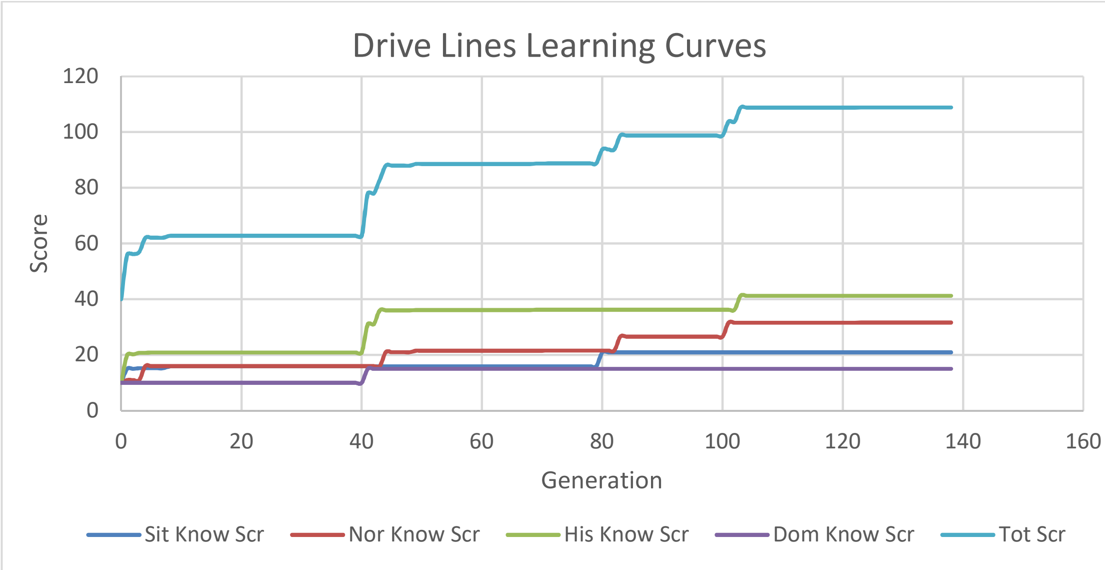


Figure 106: Drive Lines Learning Curves

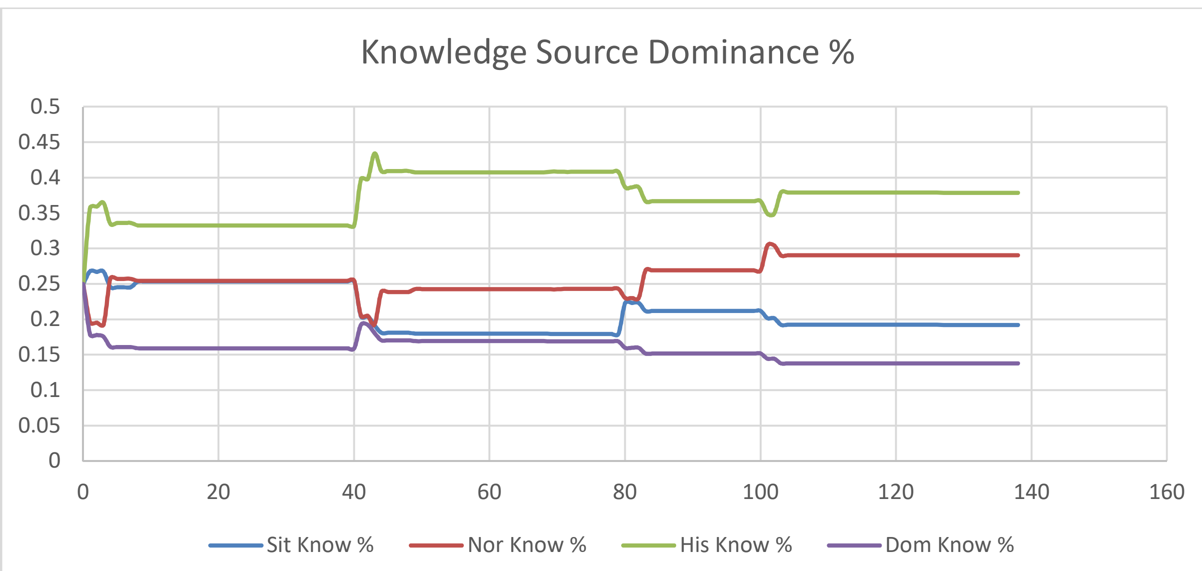


Figure 107: Drive Lines Knowledge Source Dominance

Out of the four structure types, the Drive Line type was the only one where an exploitative knowledge source (Historical Knowledge) dominated from start to finish. The smaller ruleset here may have been the reason for this. However, Normative Knowledge, an explorative knowledge source, was the second-most dominant.

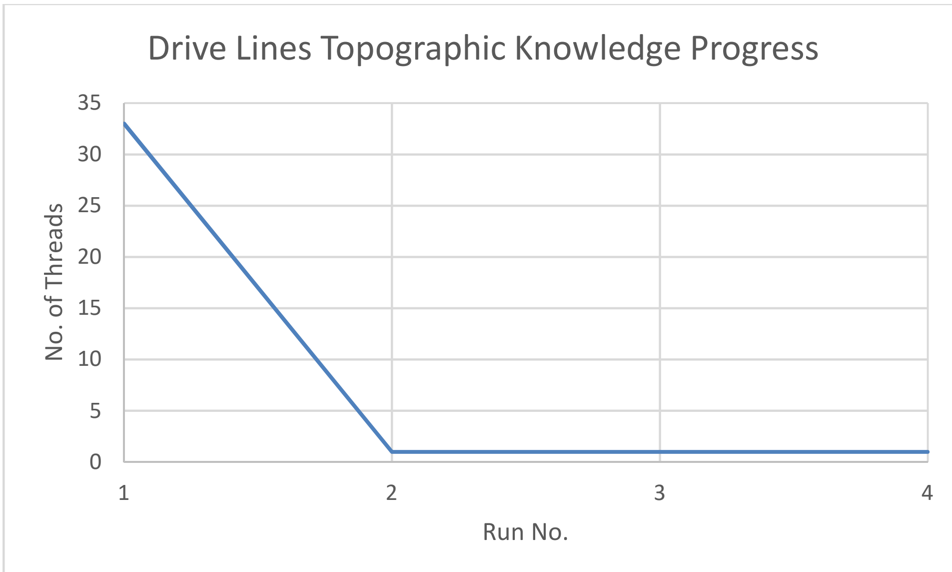


Figure 108: Drive Lines Topographic Knowledge Progress

8.2.3 Drive Lines Frames

We can use a data visualizer system to convert each of the entries in *Table 14: CAPSO's Output – Drive Line Structure Type* into a geographical heatmap corresponding to each entry. Said heatmaps can be found in Figure 109 though Figure 119.

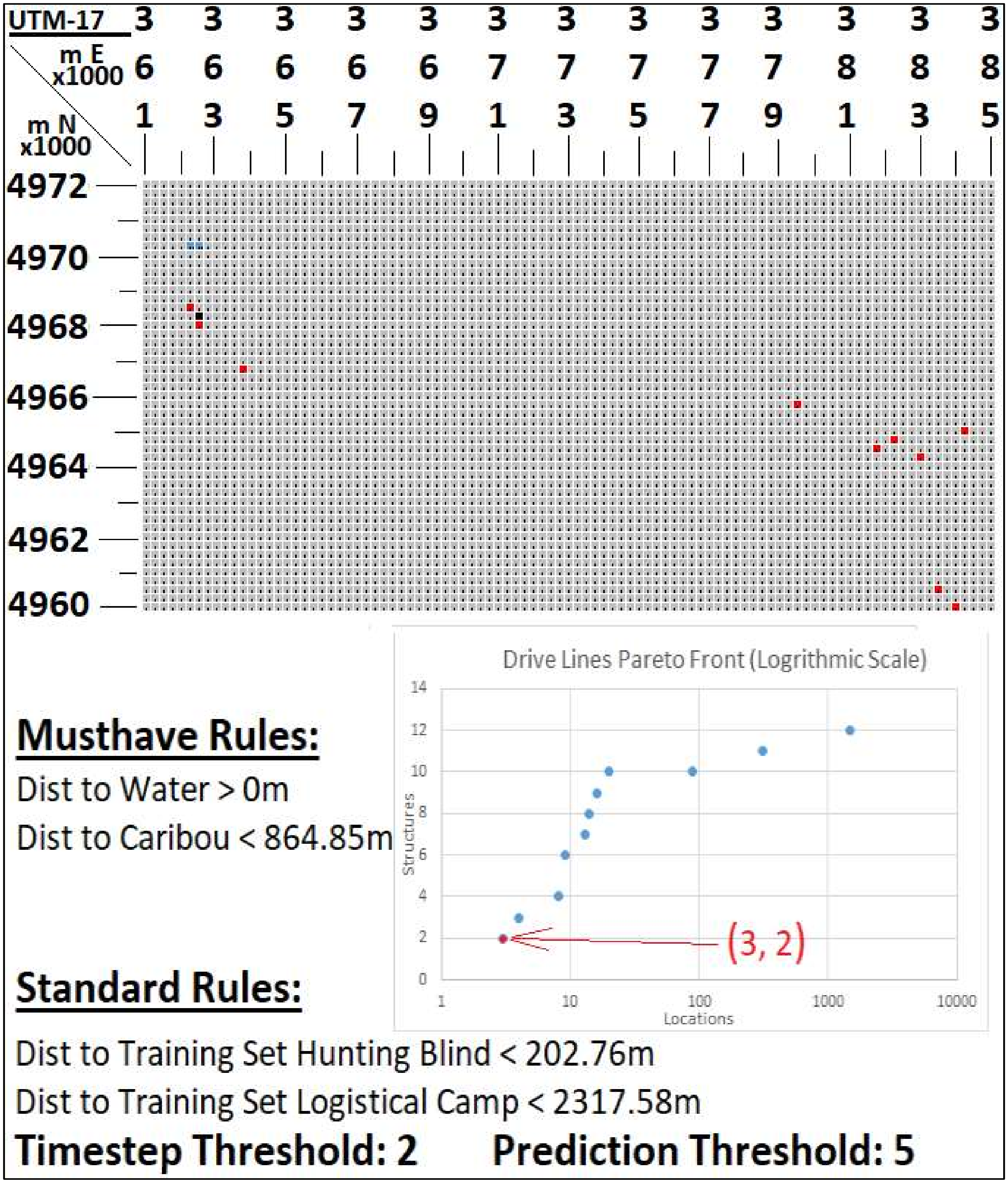


Figure 109: Drive Lines Frame (Locations, Structures) = (3, 2)

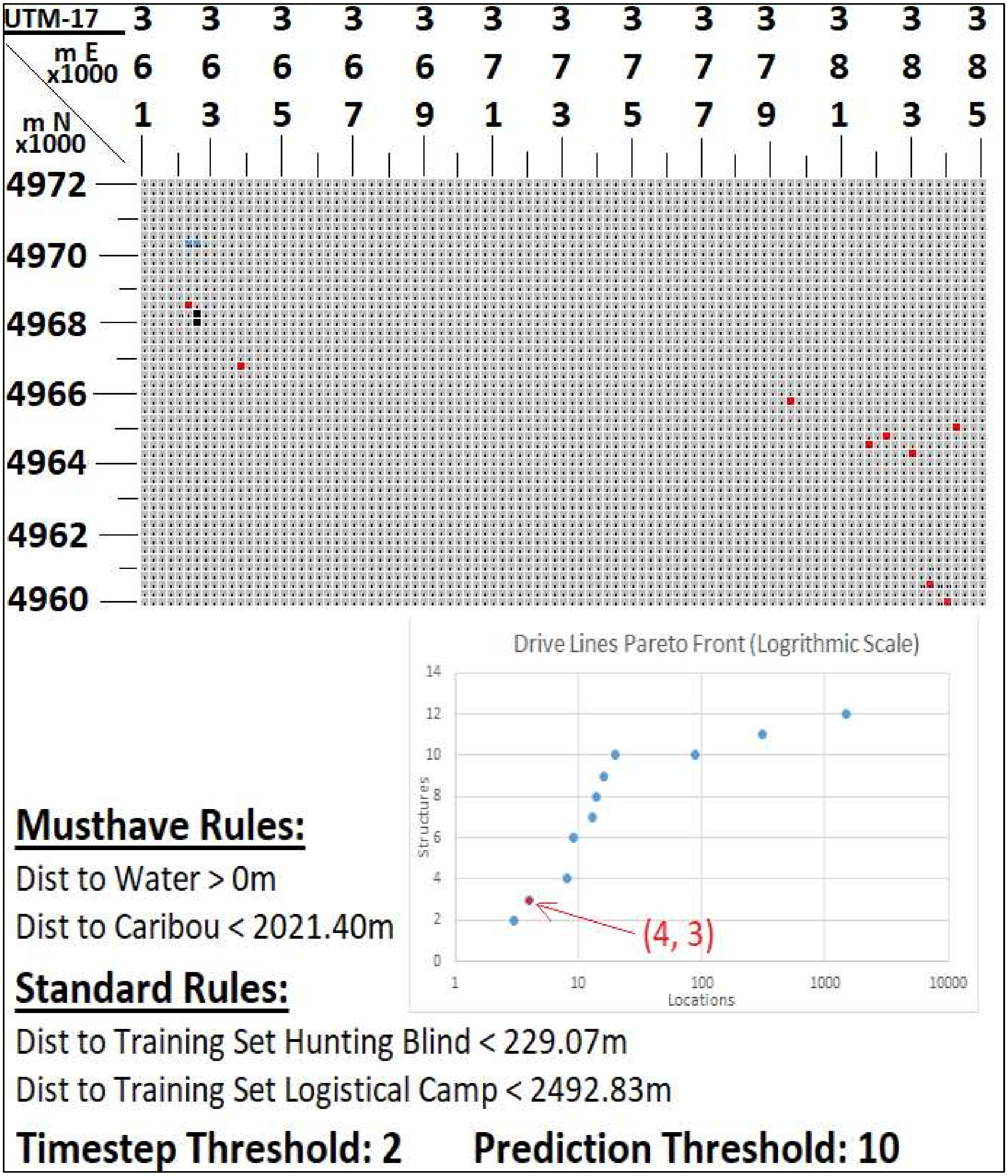


Figure 110: Drive Lines Frame (Locations, Structures) = (4, 3)

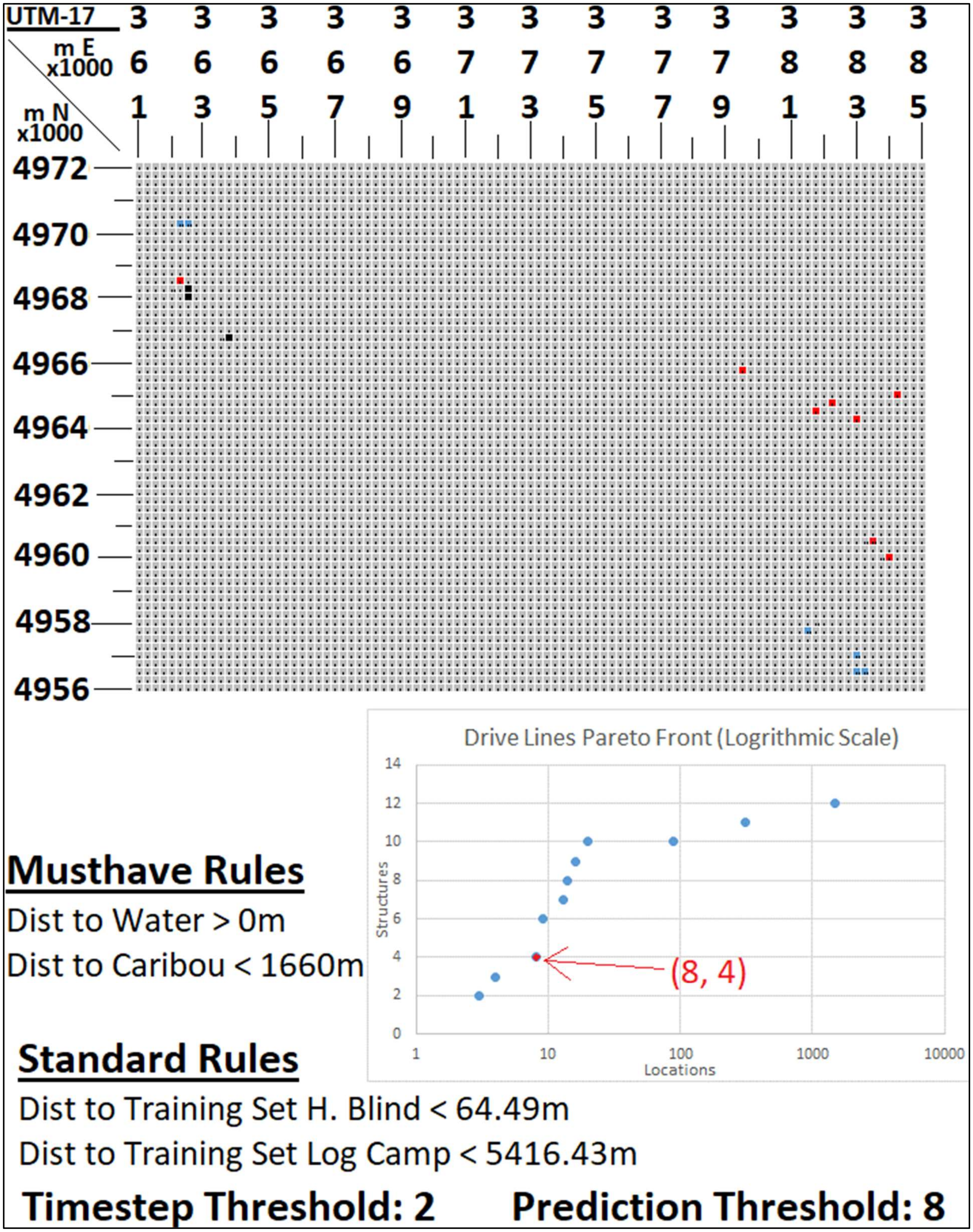


Figure 111: Drive Lines Structure (Locations, Structures) = (8, 4)

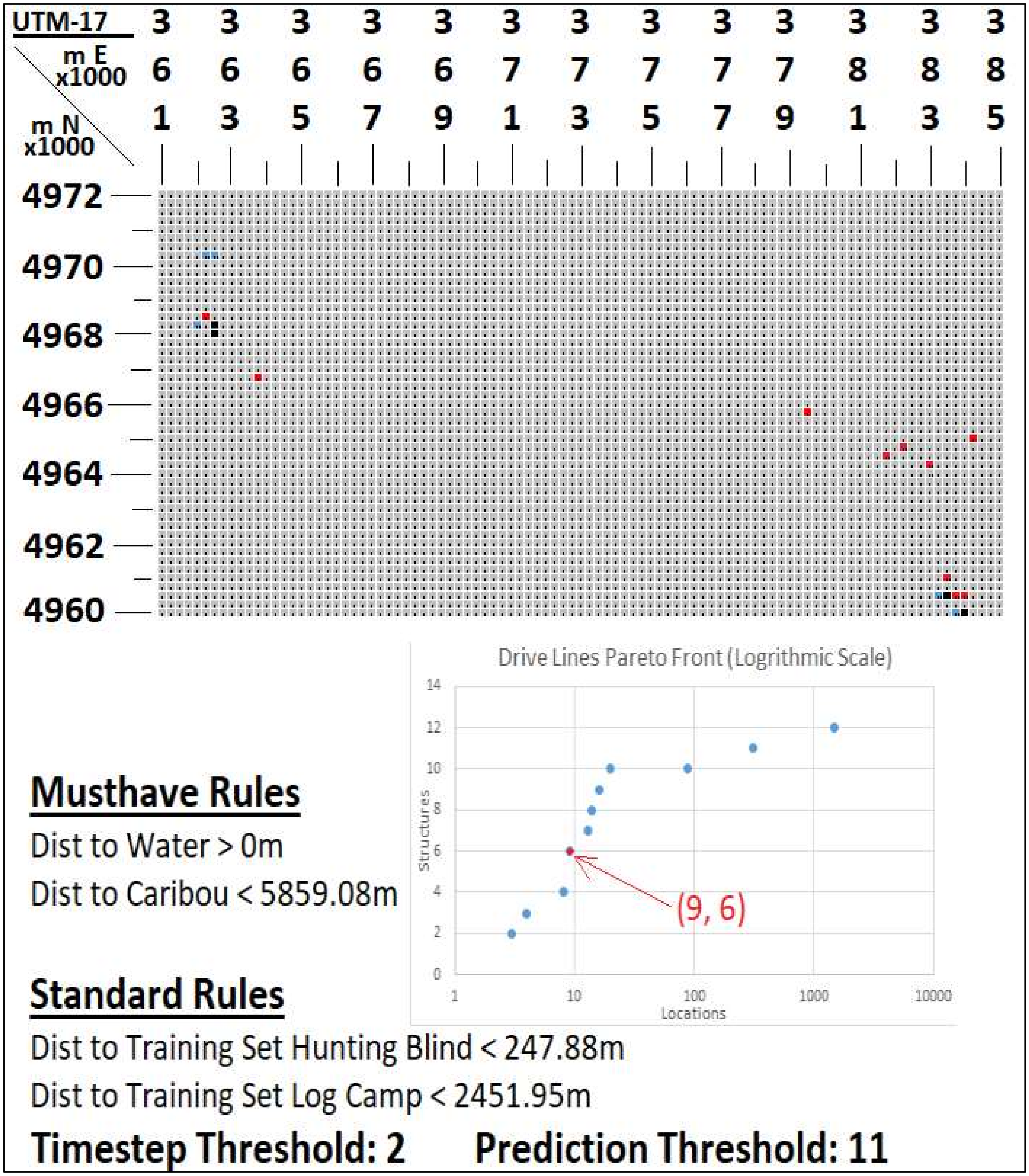


Figure 112 = Drive Lines Frame (Locations, Structures) = (9, 6)

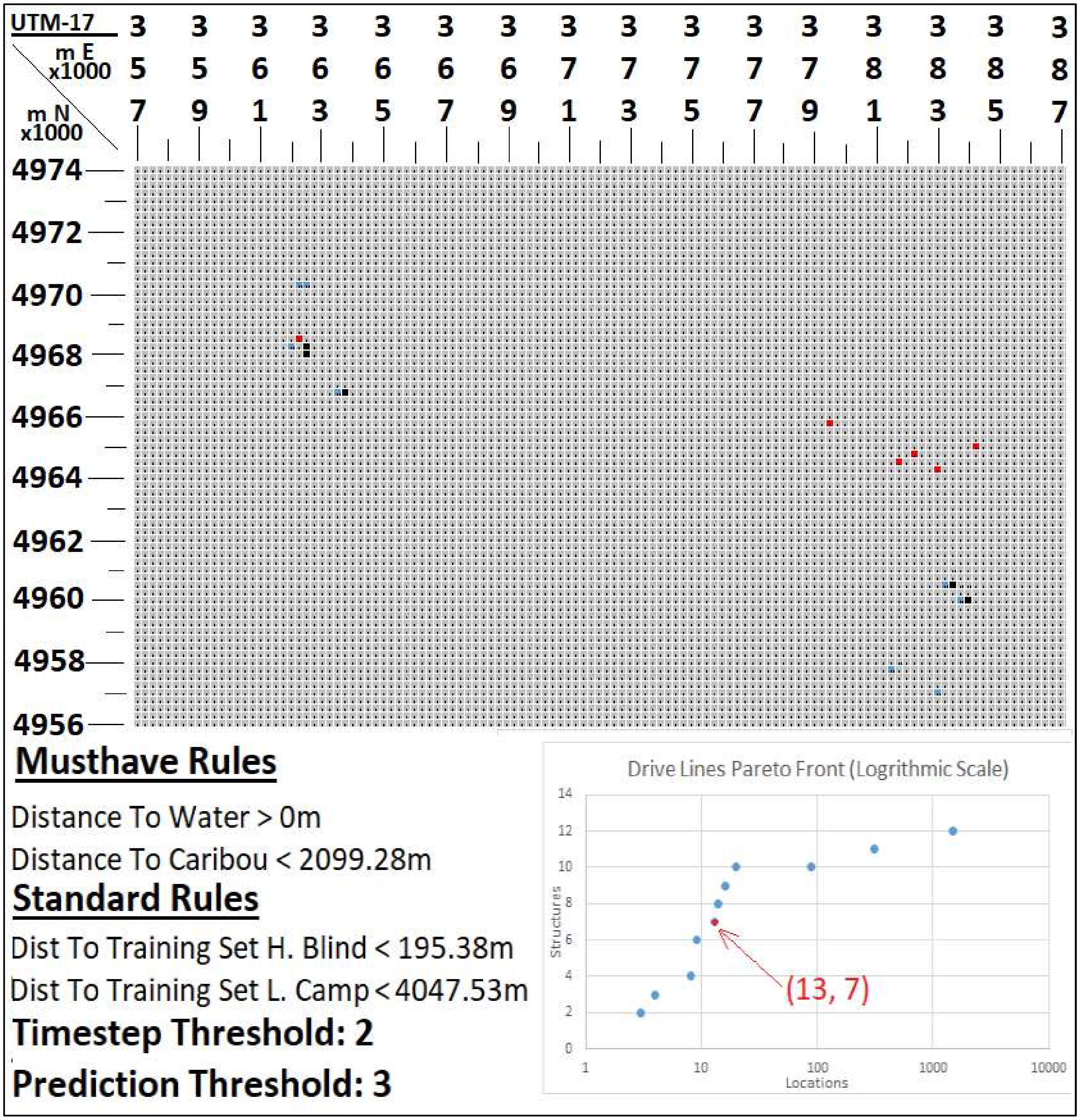


Figure 113: Drive Lines Frame (Structures, Locations) = (13, 7)

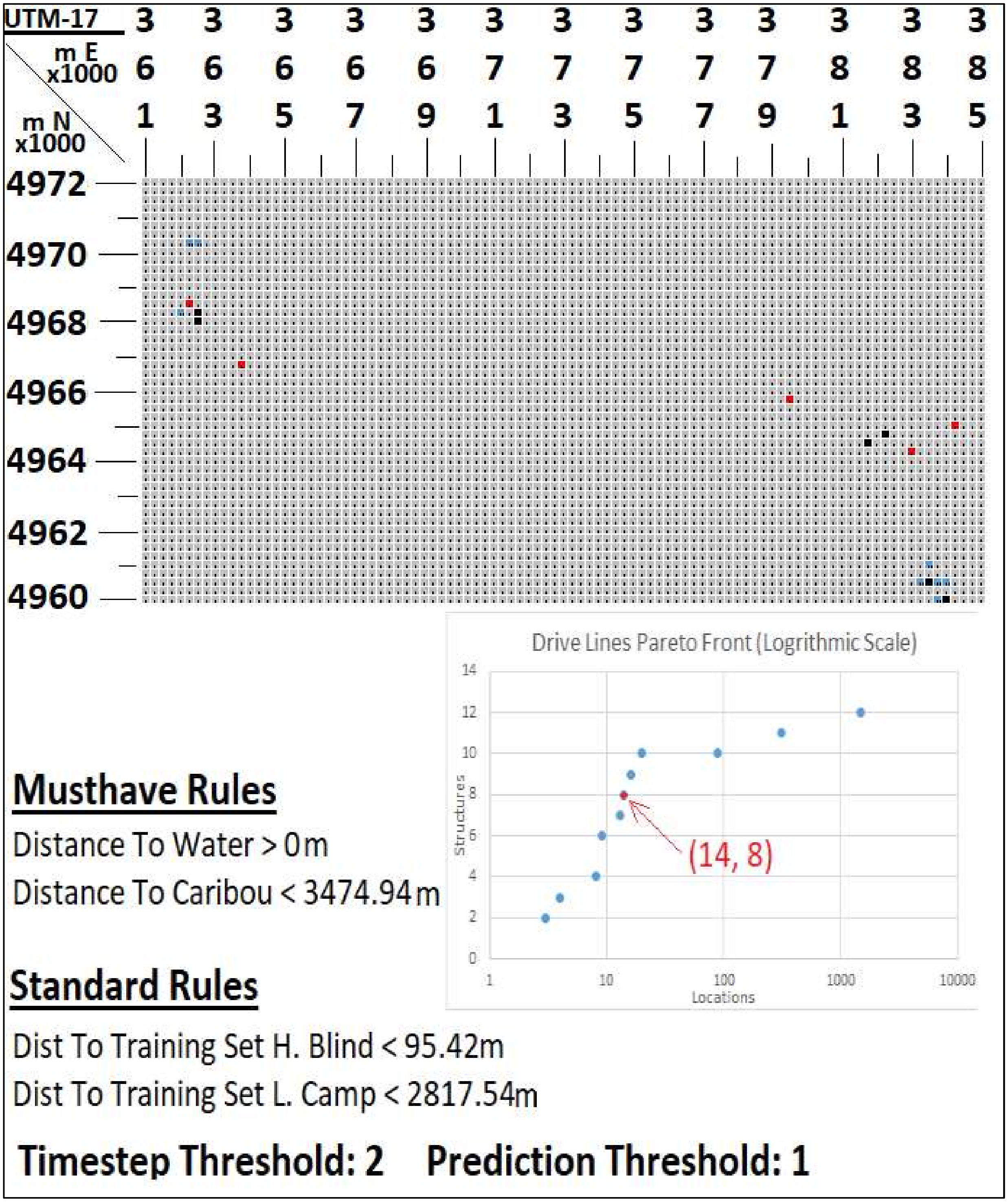


Figure 114: Drive Lines Frame (Locations, Structures) = (14, 8)

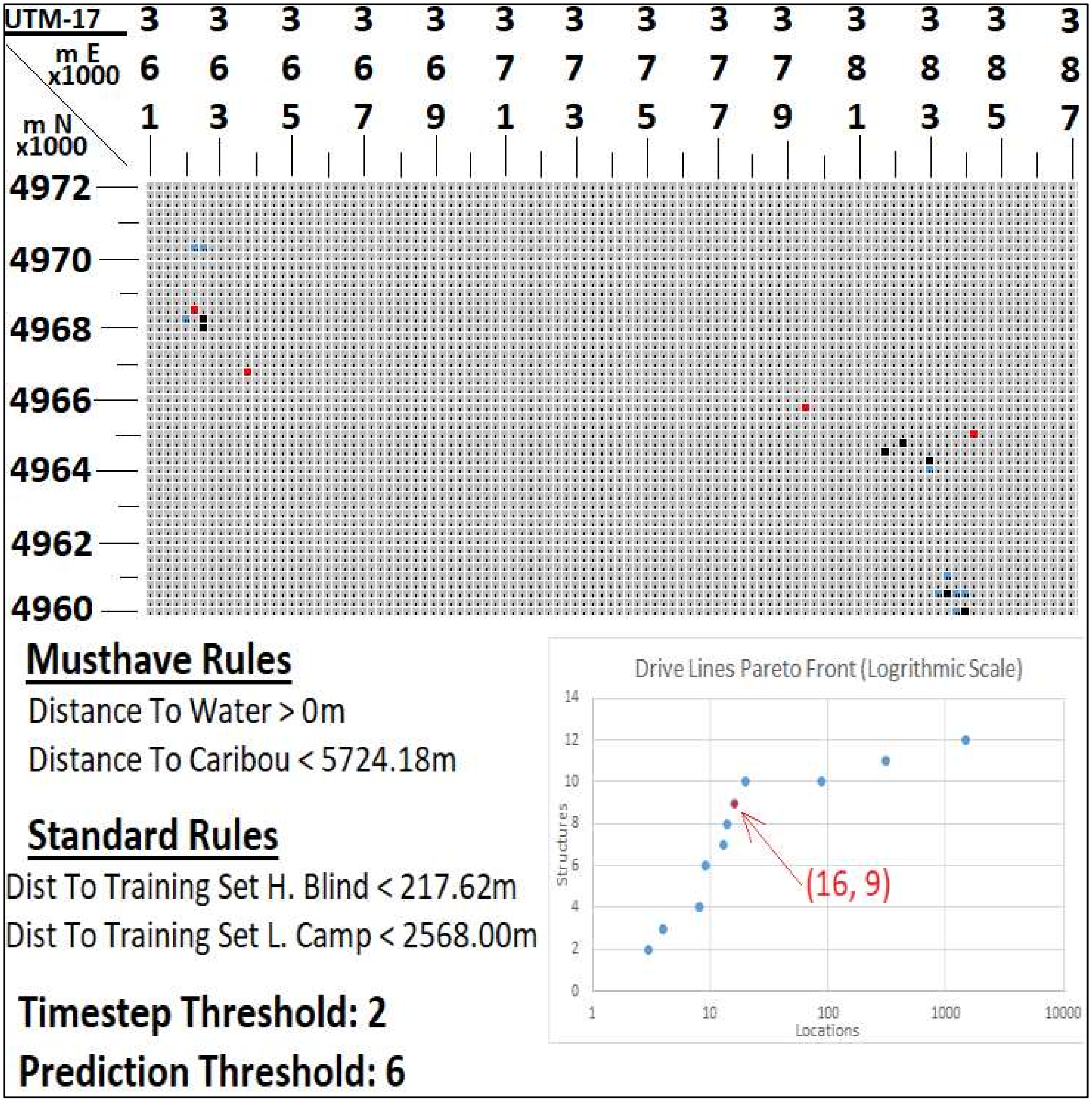


Figure 115: Drive Lines (Locations, Structures) = (16, 9)

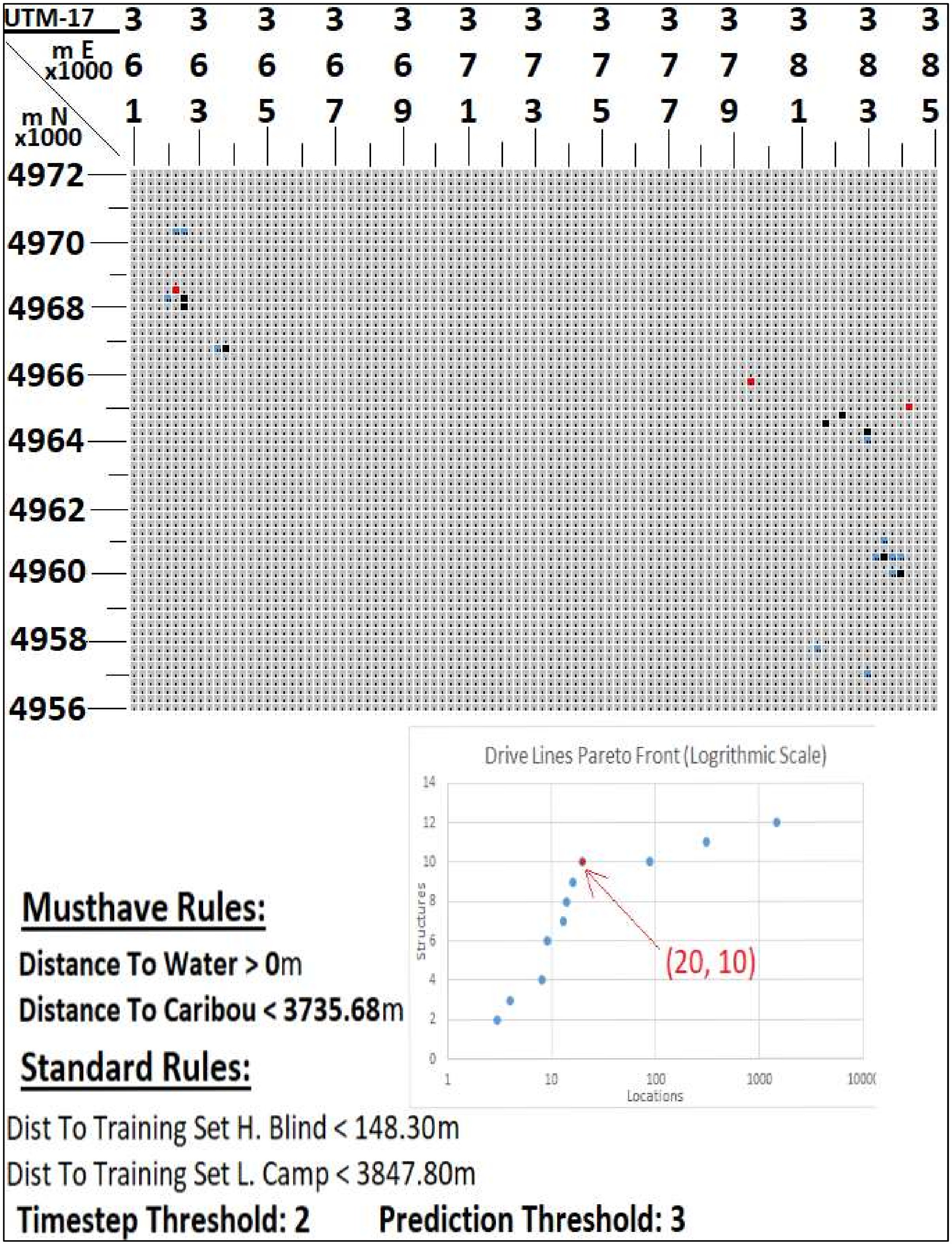


Figure 116: Drive Lines Frame (Locations, Structures) = (20, 10)

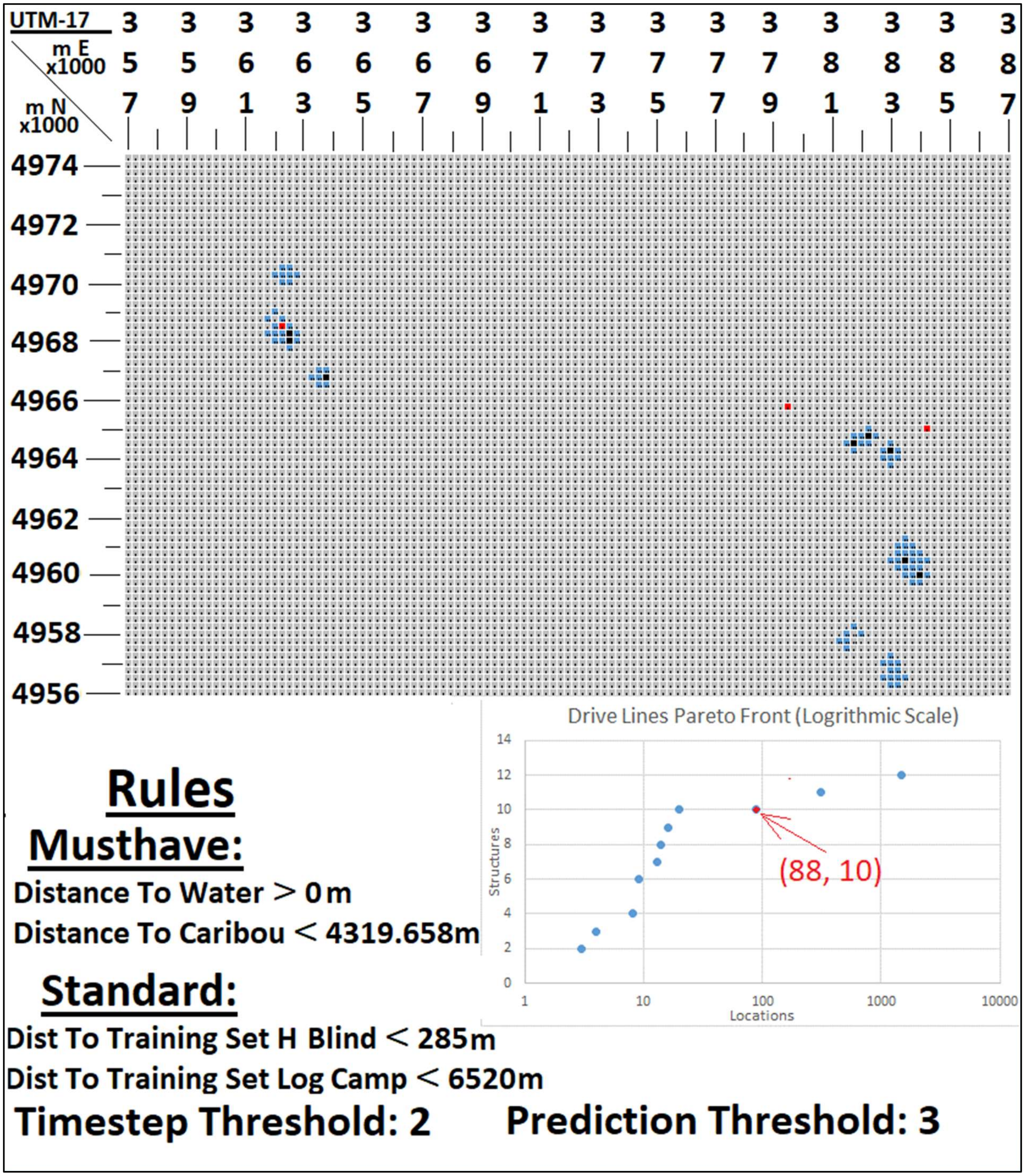


Figure 117: Drive Lines Frame (Locations, Structures) = (88, 10)

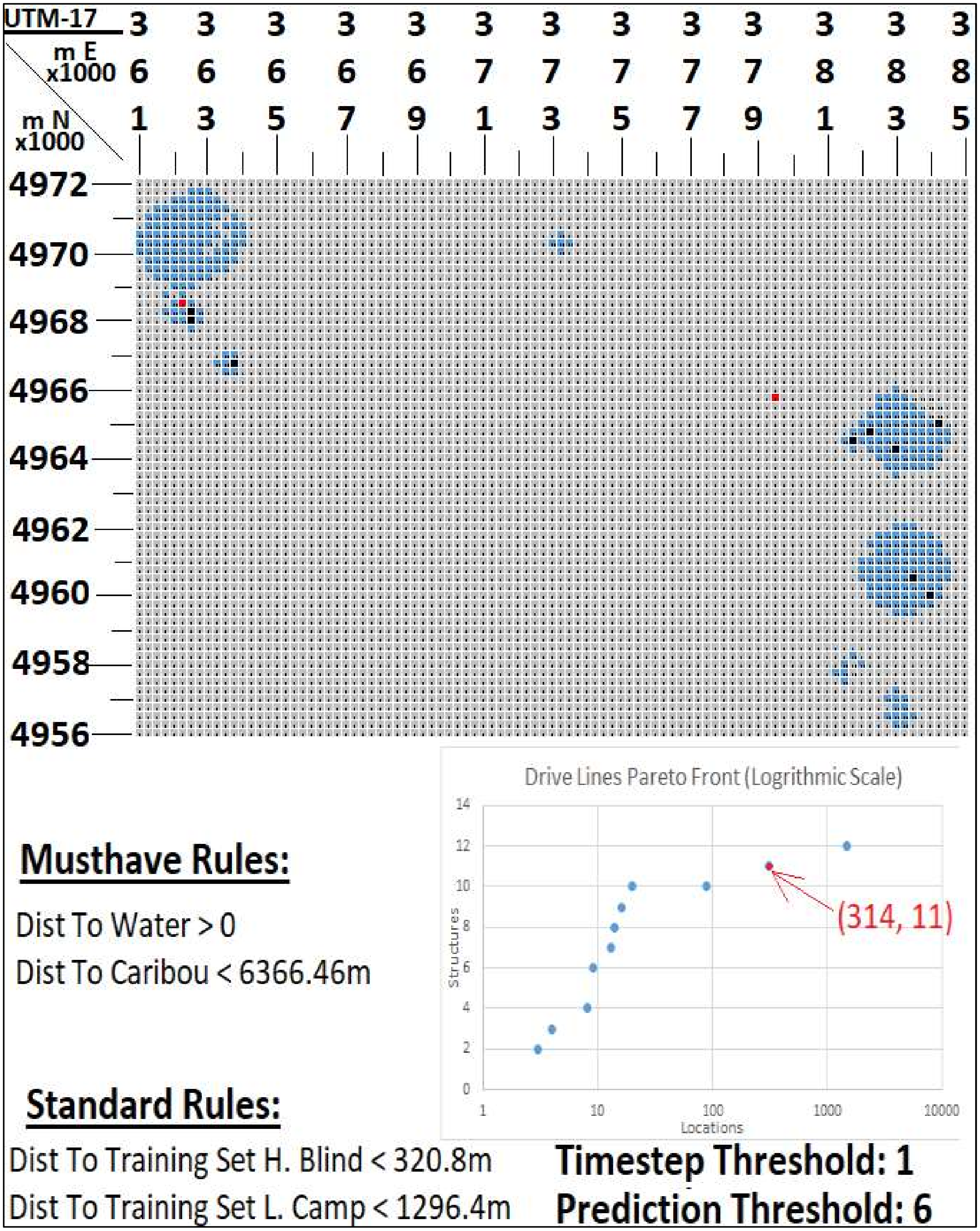


Figure 118: Drive Lines Frame (314, 11)

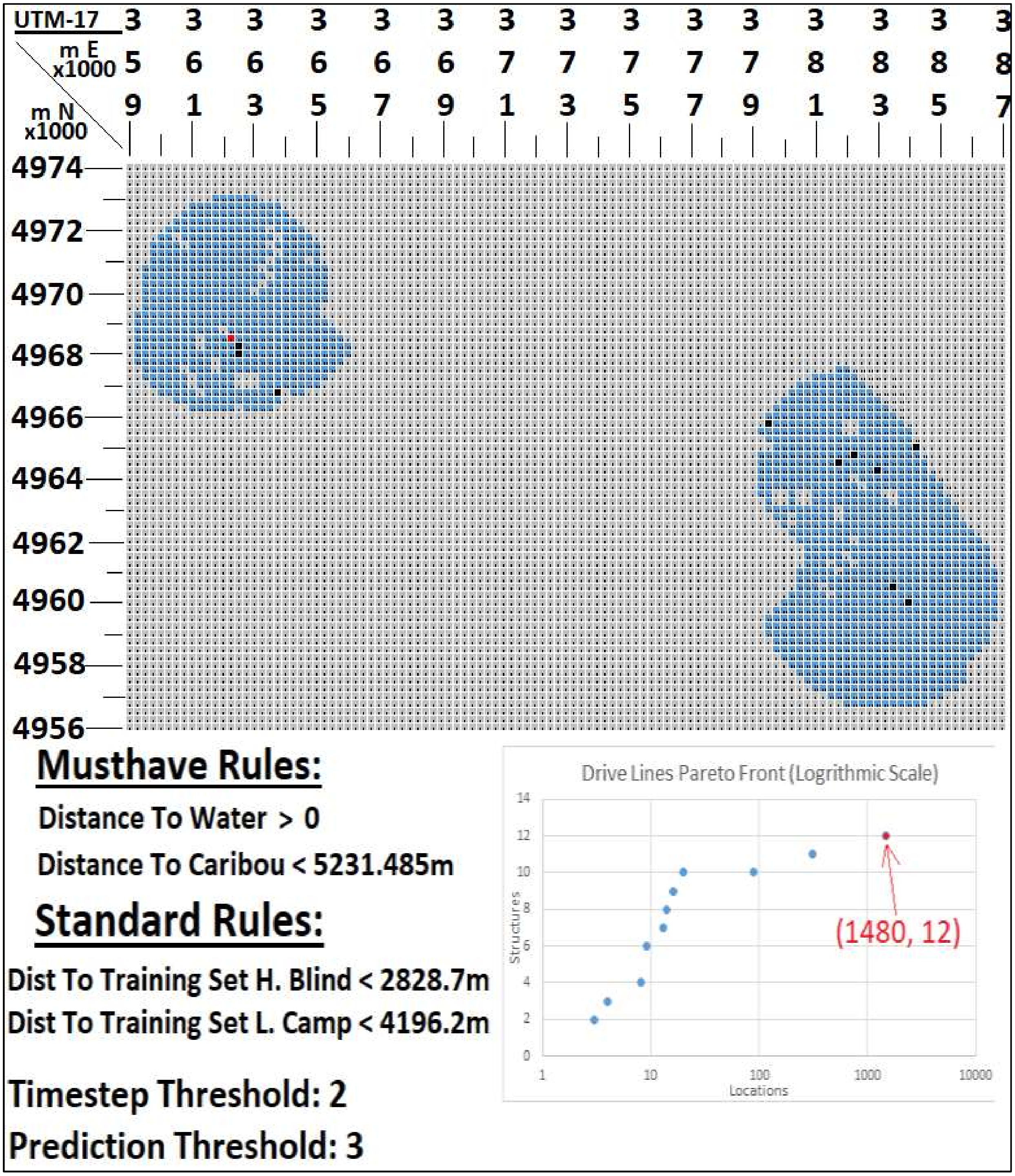


Figure 119: Drive Lines Frame (1480, 12)

8.3 Caches

8.3.1 CAPSO Output for Cache Structure Type

Table 15 contains the Pareto Front along with corresponding parameter values for the Cache structure type. It took 6.002 hours for CAPSO to produce these results.

#Locations	#Structures	Rules:	(Musthave) Dist to Caribou <	(Standard) Dist to T. Set Hunting Blind <	(Standard) Dist to T. Set Logistical Camp <	Timestep Threshold	Prediction Threshold
1	1	Rule Thresholds:	789.389	148.3922294	1348.921106	2	7
4	2	Rule Thresholds:	4257.448	137.369015	2187.217972	2	10
7	3	Rule Thresholds:	6117.35	242.8696907	897.416077	2	8
11	4	Rule Thresholds:	3674.529	222.7980764	2164.126084	2	3
65	5	Rule Thresholds:	4251.174	338.0939142	4013.202386	2	6

Table 15: CAPSO's Outputs for Cache Structure Type

8.3.2 Pareto Front, Learning Curve, and Knowledge Source Dominance Graphs

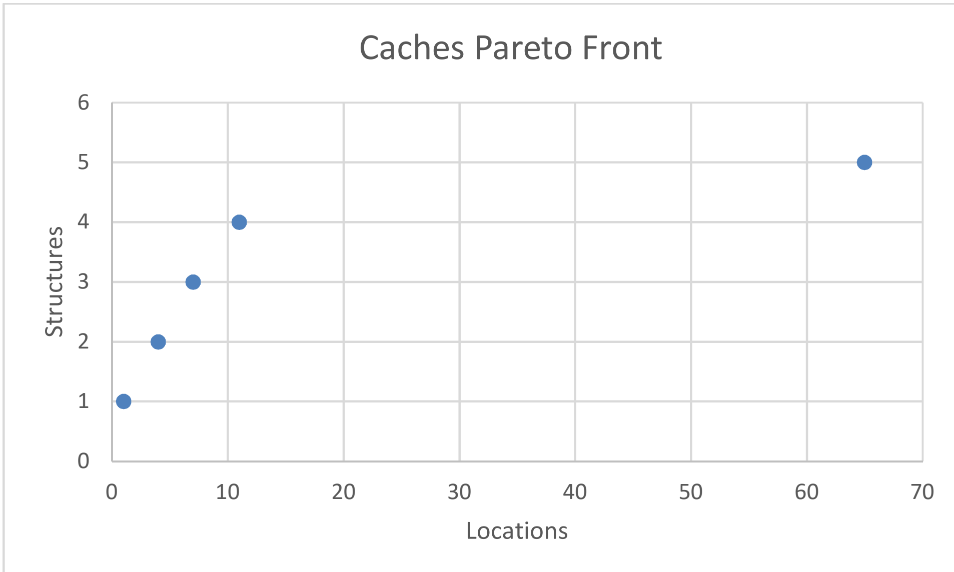


Figure 120: Caches Pareto Front

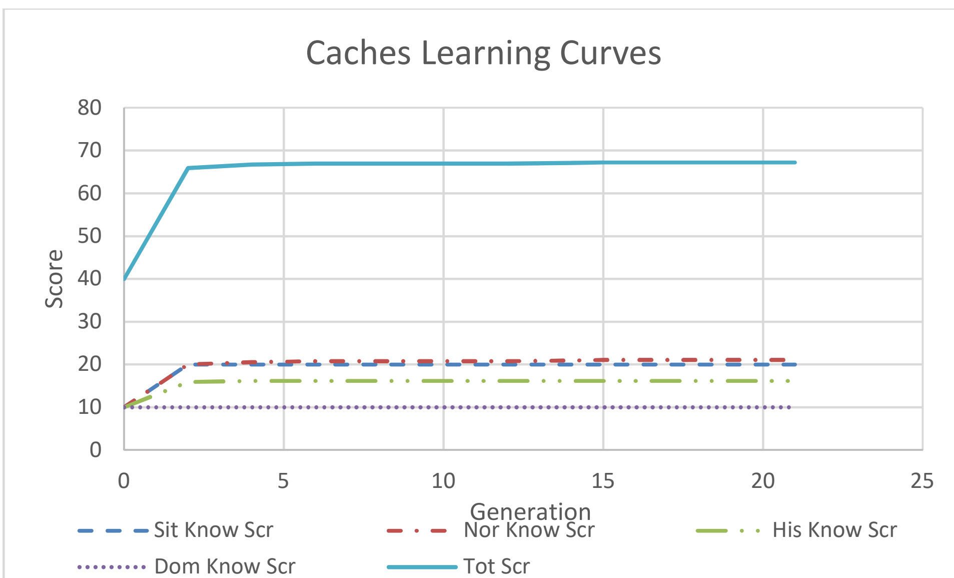


Figure 121: Caches Learning Curves

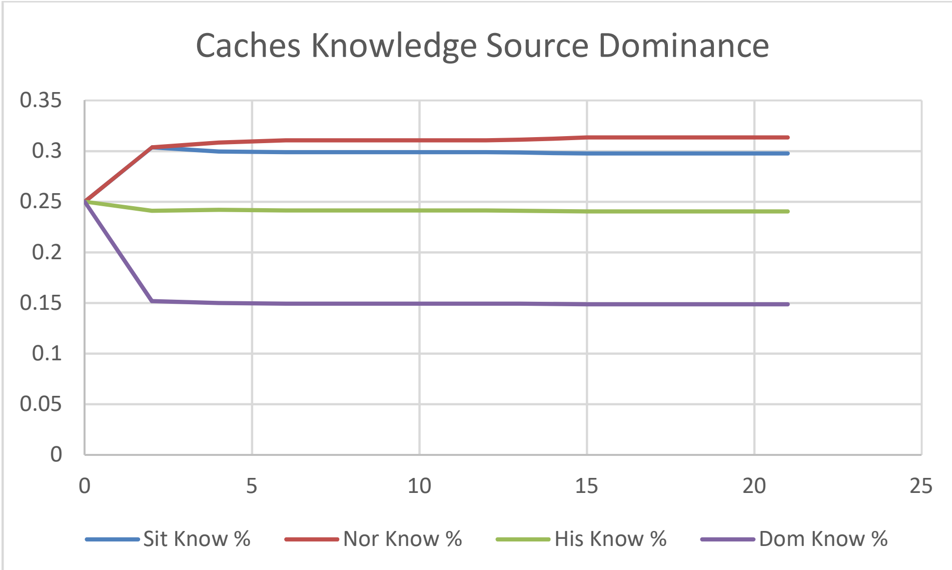


Figure 122: Caches Knowledge Source Dominance Plot

We can see that Normative Knowledge, an explorative knowledge source, dominated the entire time for the Cache occupational structure type. This may be due to the fact that there is a tough musthave rule in its ruleset (Distance to Fall Caribou is a lot tougher than Distance to Overall Caribou, i.e., either spring or fall).

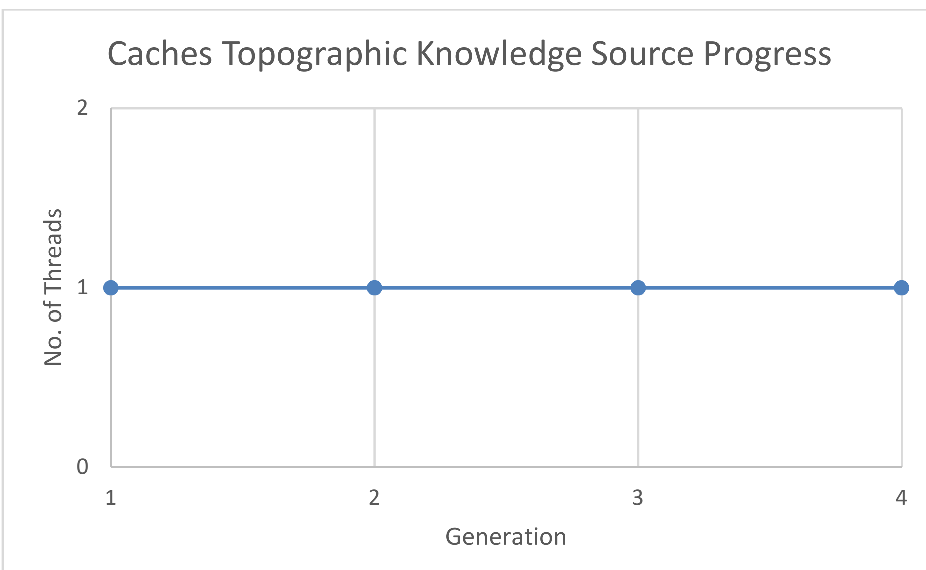


Figure 123: Caches Topographic Knowledge Progress

Regarding Figure 123 above, the particular paucity of data for the Cache structure type (i.e., there were only 5 structures in the Cache structure category) meant that the task of multi-objective optimization for Caches was a much smaller and easier task than for the previous structure types. Because of this, CAPSO did not feel it necessary to subdivide and parallelize the search process, hence the search process proceeded serially and thus Topographic Knowledge was not used in this particular case.

8.3.3 Frames

We can use a data visualizer system to convert each of the entries in *Table 15: CAPSO's Outputs for Cache Structure Type* into a geographical heatmap corresponding to each entry. Said maps can be found in Figure 124 through Figure 128.

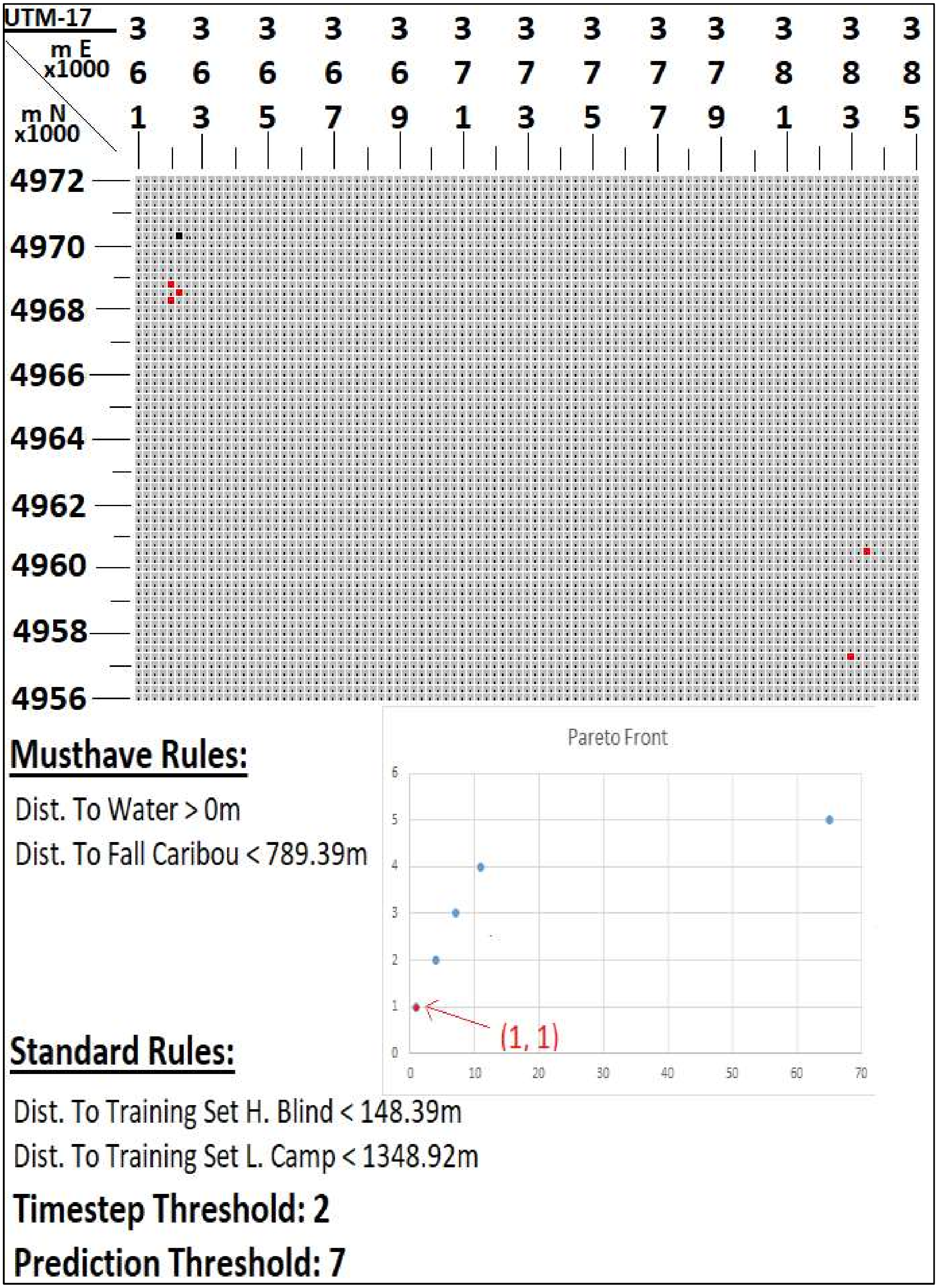


Figure 124: Caches Frame (Locations, Structures) = (1, 1)

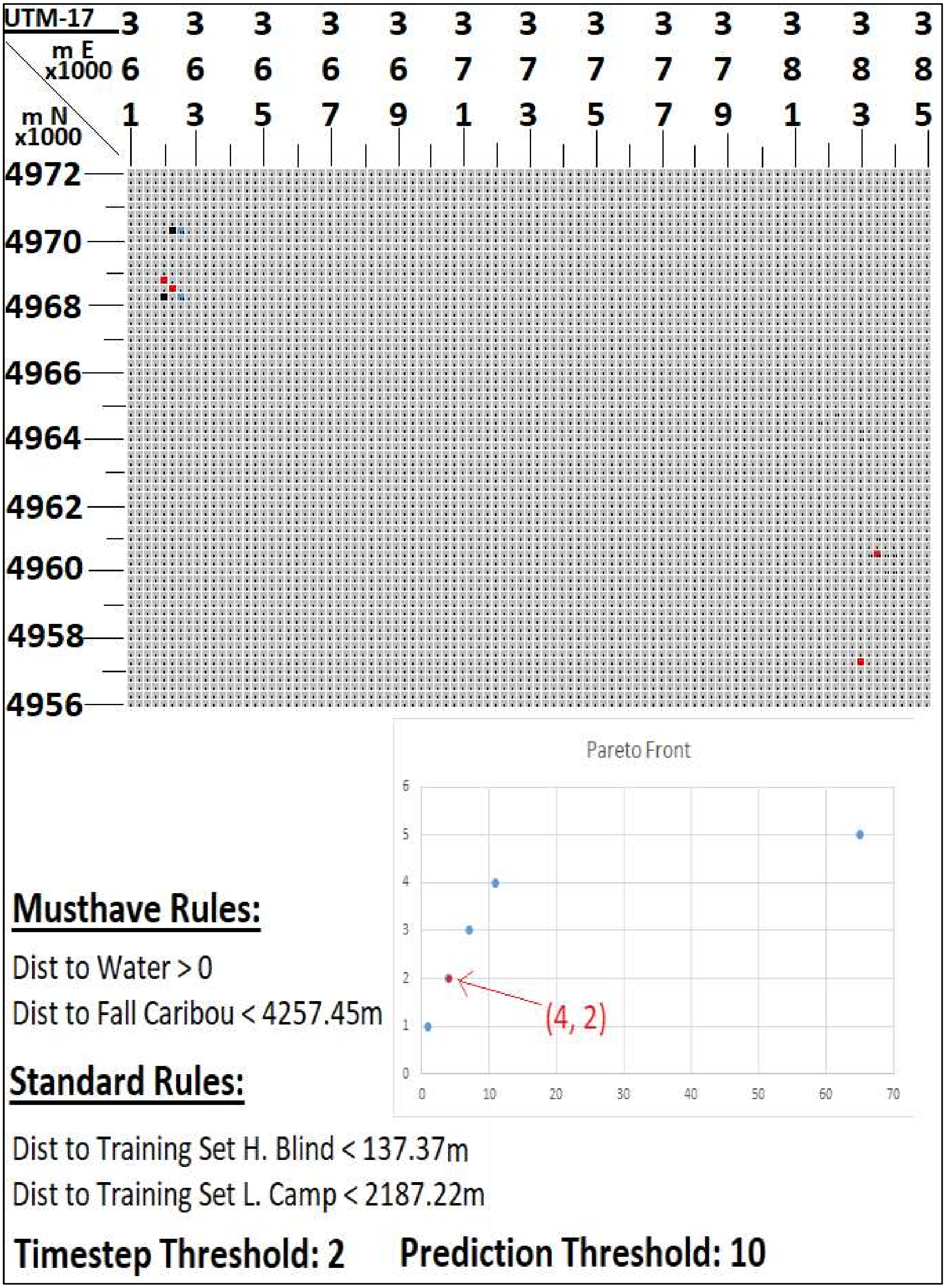


Figure 125: Caches Frame (Locations, Structures) = (4, 2)

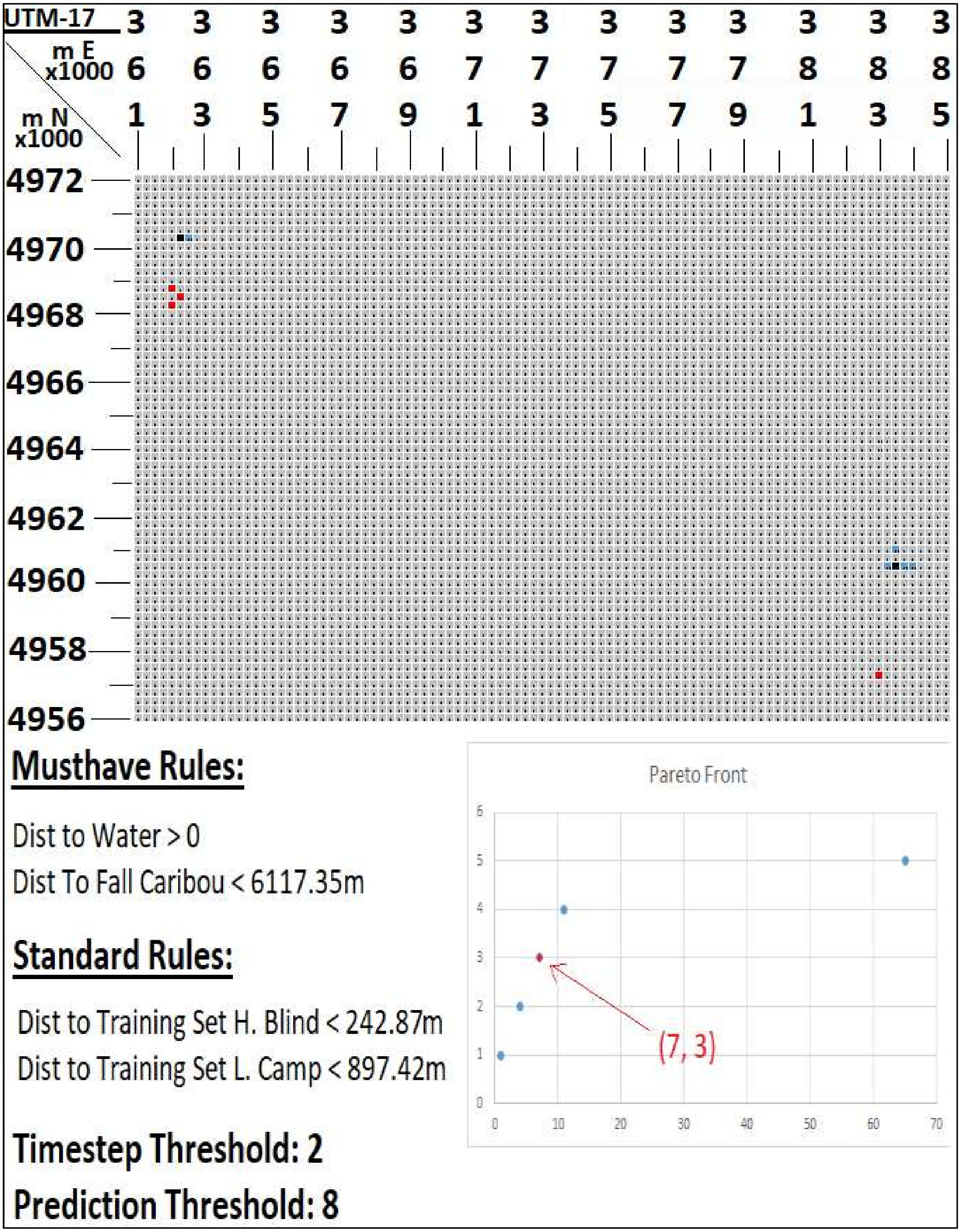


Figure 126: Caches Frame (Locations, Structures) = (7, 3)

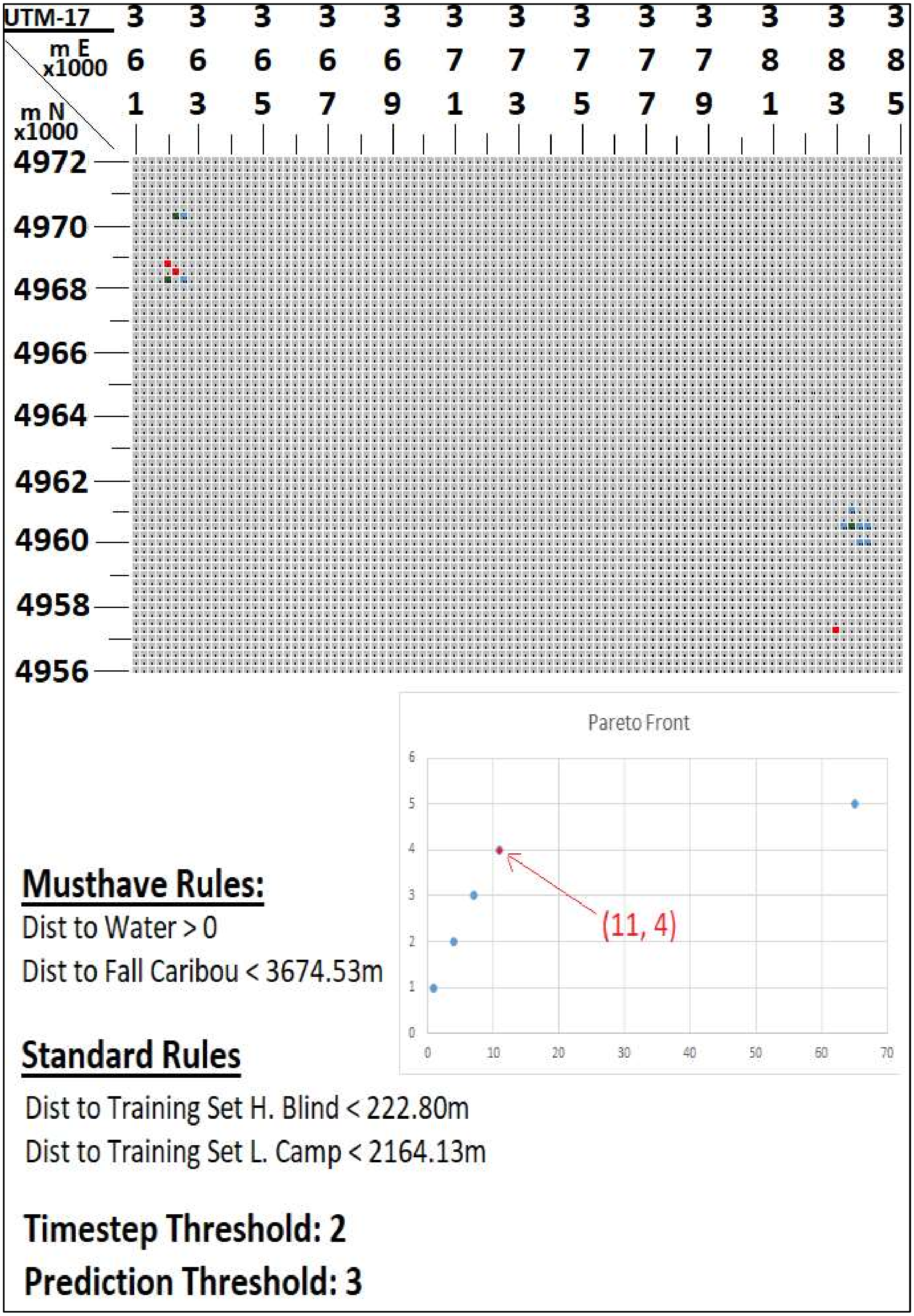


Figure 127: Caches Frame (Locations, Structures) = (11, 4)

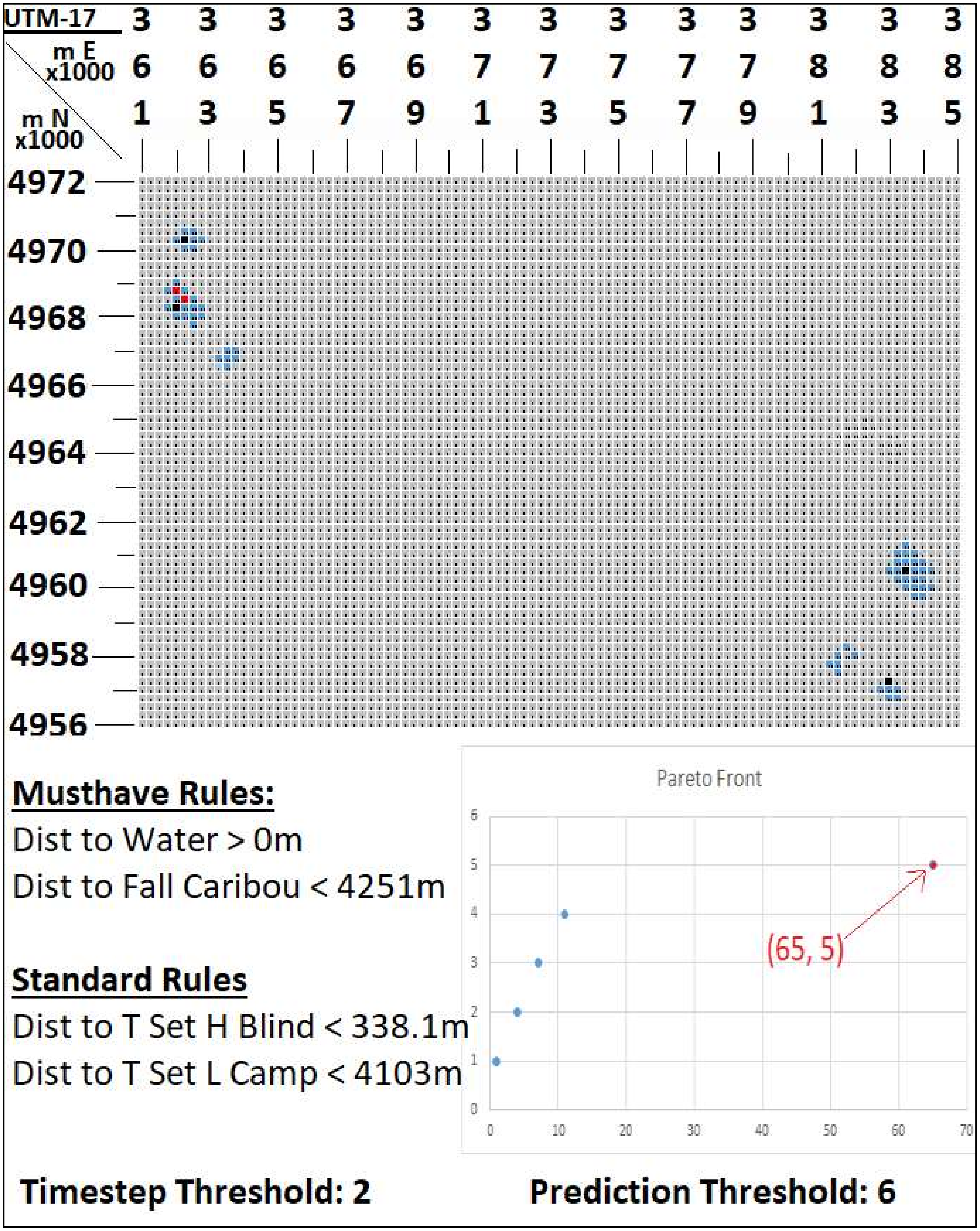


Figure 128: Caches Frame (Locations, Structures) = (65, 5)

8.4 Logistical Camps

8.4.1 CAPSO's Output for Logistical Camp Structure Type

Table 16 contains the Pareto Front along with corresponding parameter values for the Logistical Camp structure type. It took 7.673 hours for CAPSO to produce these results.

#Locations	#Structures	Rule Thresholds:	Dist to Caribou <	Dist to T Set H. Blind <	Dist to T. Set Cache <	Veg % >	Timestep Thresh	Prediction Thresh
2	1	Rule Thresholds:	5164.958908	1785.967536	228.1640564	0.1211432	3	2
10	3	Rule Thresholds:	792.7351219	449.3953659	5312.476698	0.98548791	2	11
56	4	Rule Thresholds:	5337.35623	399.2700423	6568.208711	0.39096809	2	18
113	5	Rule Thresholds:	4741.39574	1022.381853	584.804733	0.06654895	2	12

Table 16: CAPSO's Output for Logistical Camp Structure Type

8.4.2 Pareto Front, Learning Curve, and Knowledge Source Dominance Graphs

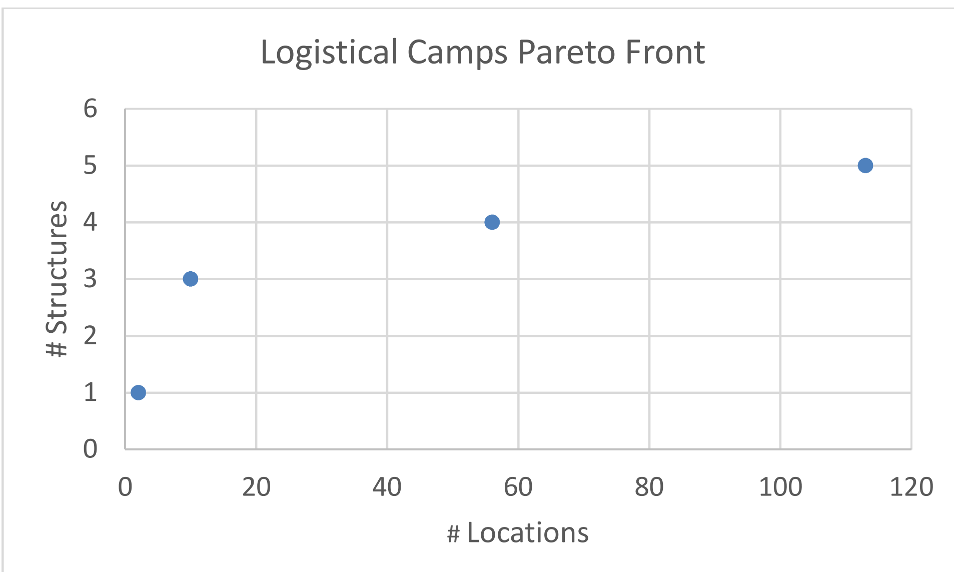


Figure 129: Logistical Camps Pareto Front

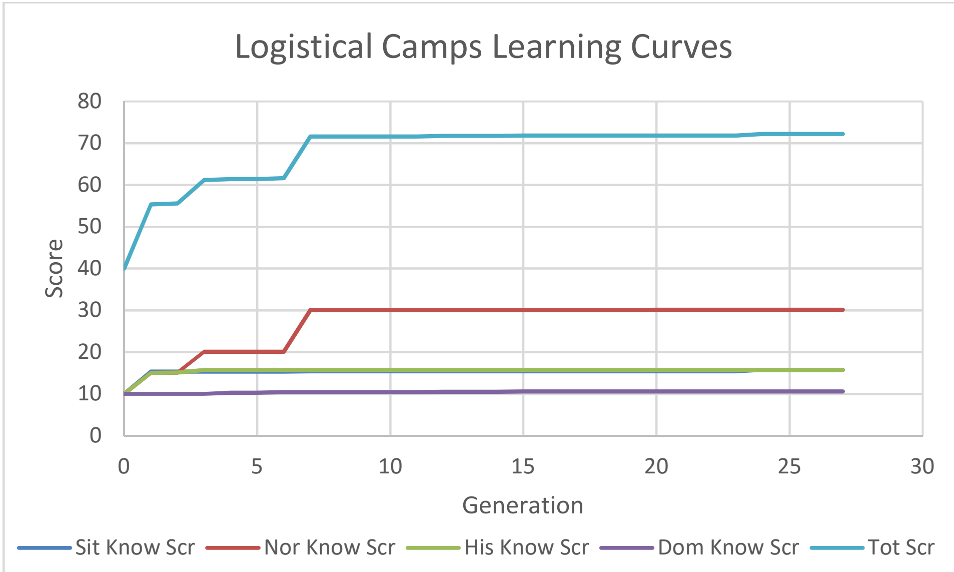


Figure 130: Logistical Camps Learning Curves

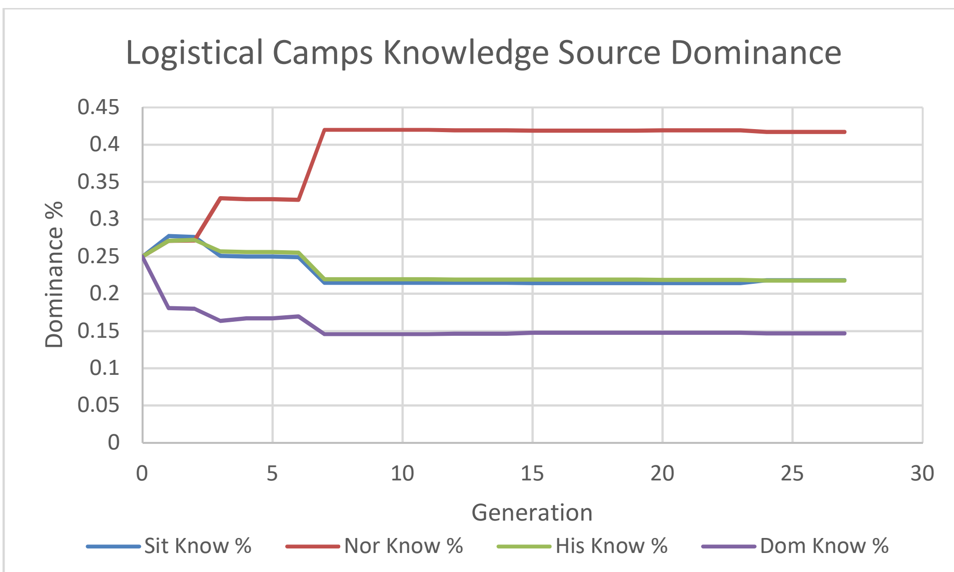


Figure 131: Logistical Camps Knowledge Source Dominance Plot

Looking at Figure 131, we can see that Normative Knowledge, an explorative knowledge source, dominated for the Logistical Camp occupational structure type.

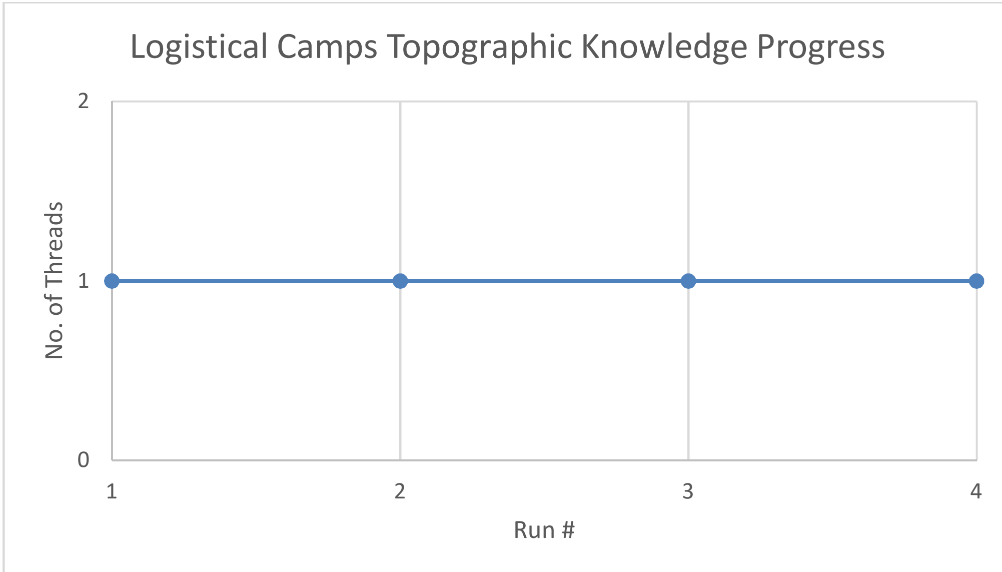
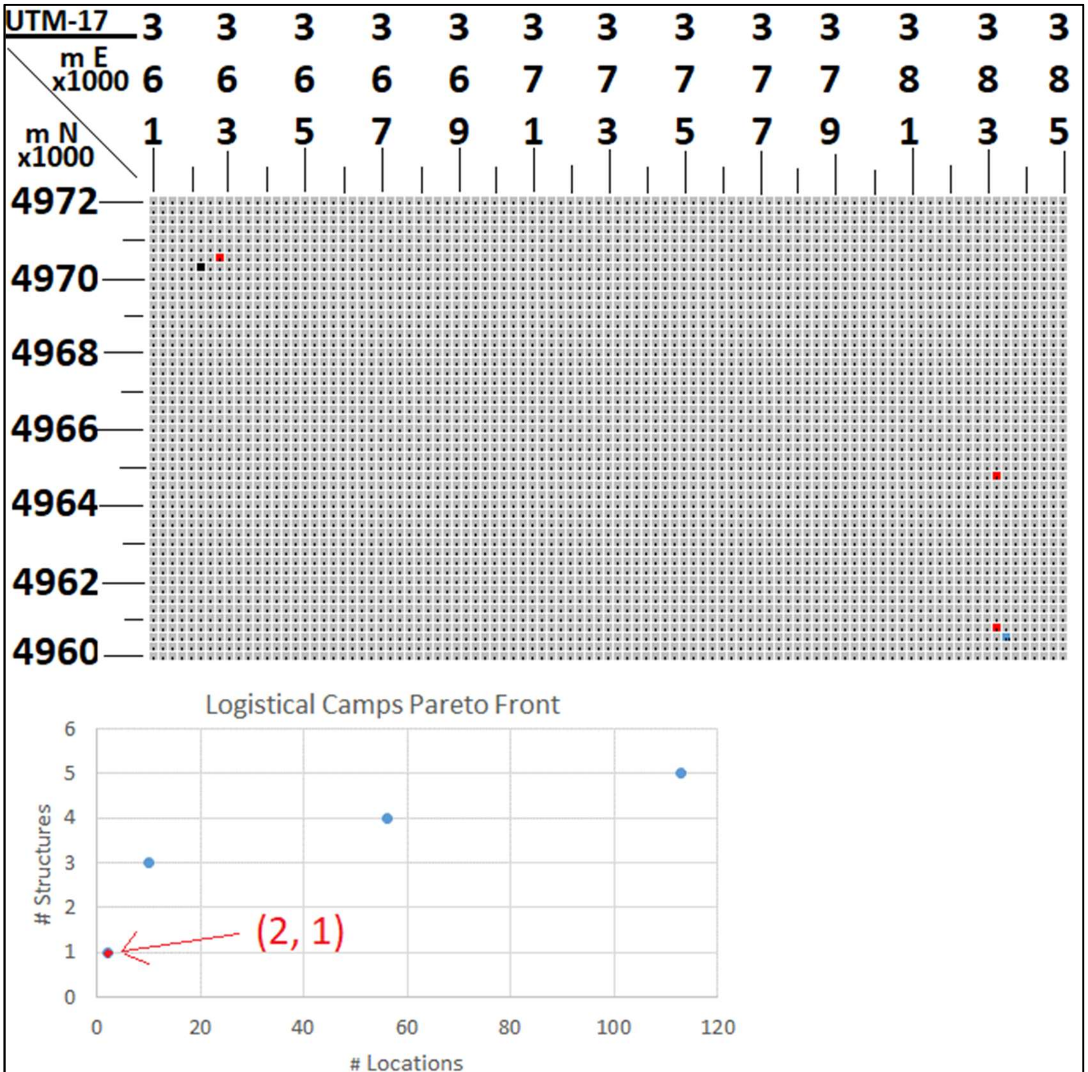


Figure 132: Logistical Camps Topographic Knowledge Progress

Regarding Figure 132 above, the particular paucity of data for the Logistical Camp structure type (i.e., there were only 4 structures in the Logistical Camp category) meant that the task of multi-objective optimization for Logistical Camps was a much smaller and easier task than for more numerous structure types such as Hunting Blinds and Drive Lines. Because of this, CAPSO did not feel it necessary to subdivide and parallelize the search process, hence the search process proceeded serially and thus Topographic Knowledge was not used in this particular case.

8.4.3 Frames

We can use a data visualizer system to convert each of the entries in *Table 16: CAPSO's Output for Logistical Camp Structure Type* into a geographical heatmap corresponding to each entry. Said heatmaps can be found in Figure 133 through Figure 136.



Musthave Rules:

Dist to Water > 0

Dist to Food Source < 5164.96m

Standard Rules:

Dist to Training Set H. Blind < 1785.97m

Dist to Cache < 228.16m

Vegetation Percentage > 12.114%

Timestep Threshold: 3 Prediction Threshold: 2

Figure 133: Logistical Camps (Locations, Structures) = (2, 1)

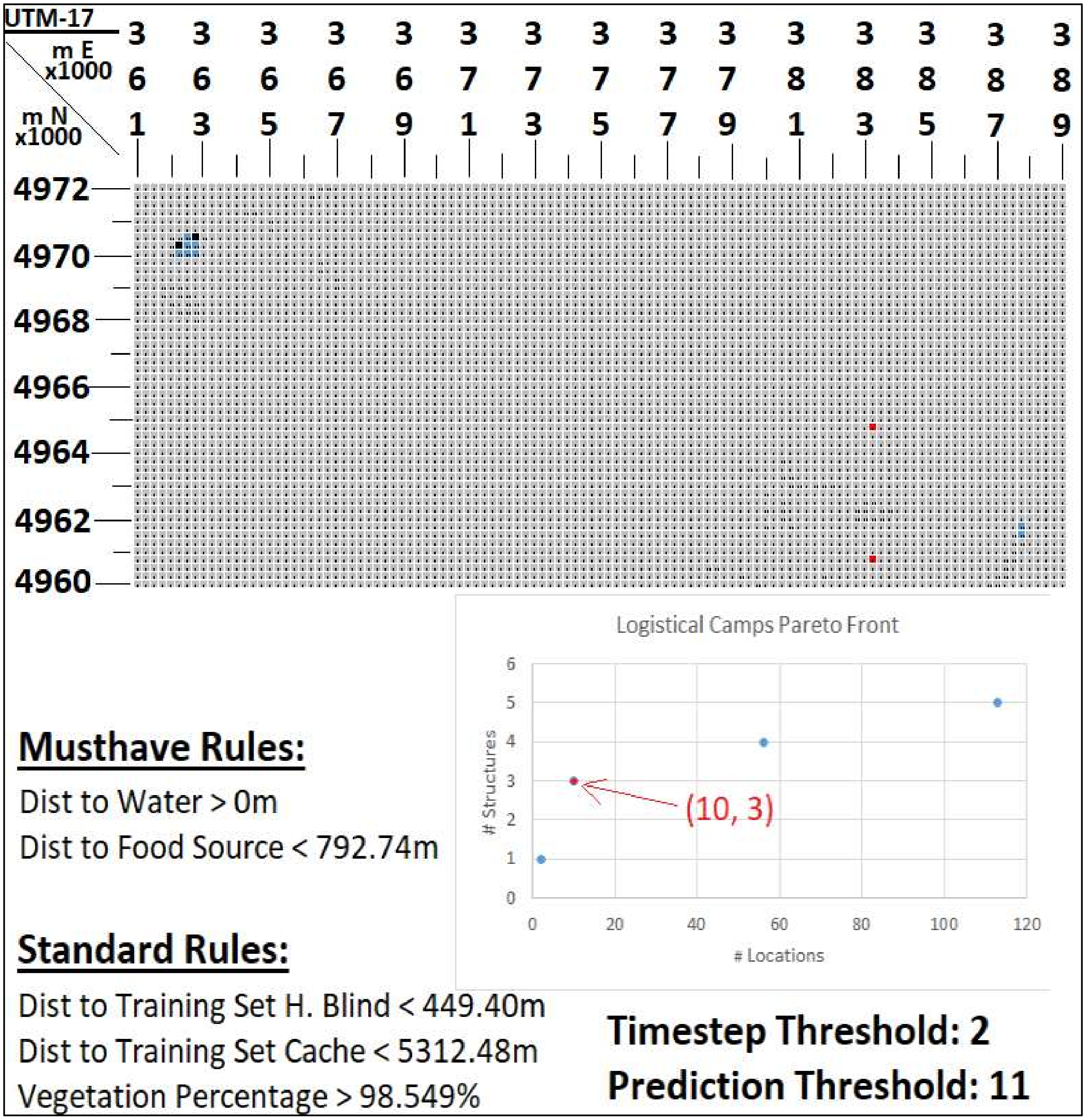


Figure 134: Logistical Camps Frame (Locations, Structures) = (10, 3)

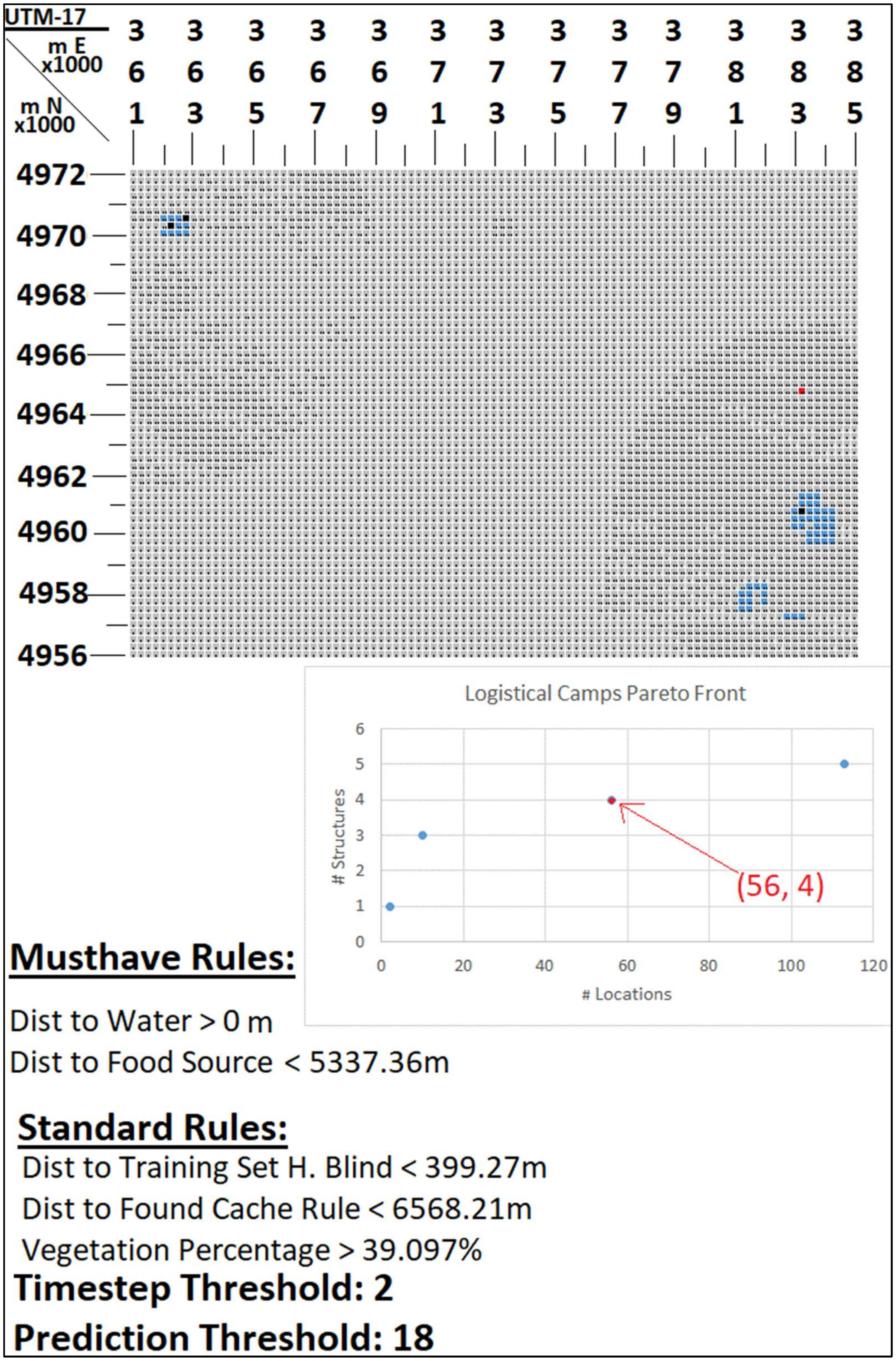


Figure 135: Logistical Camps Frame (Locations, Structures) = (56, 4)

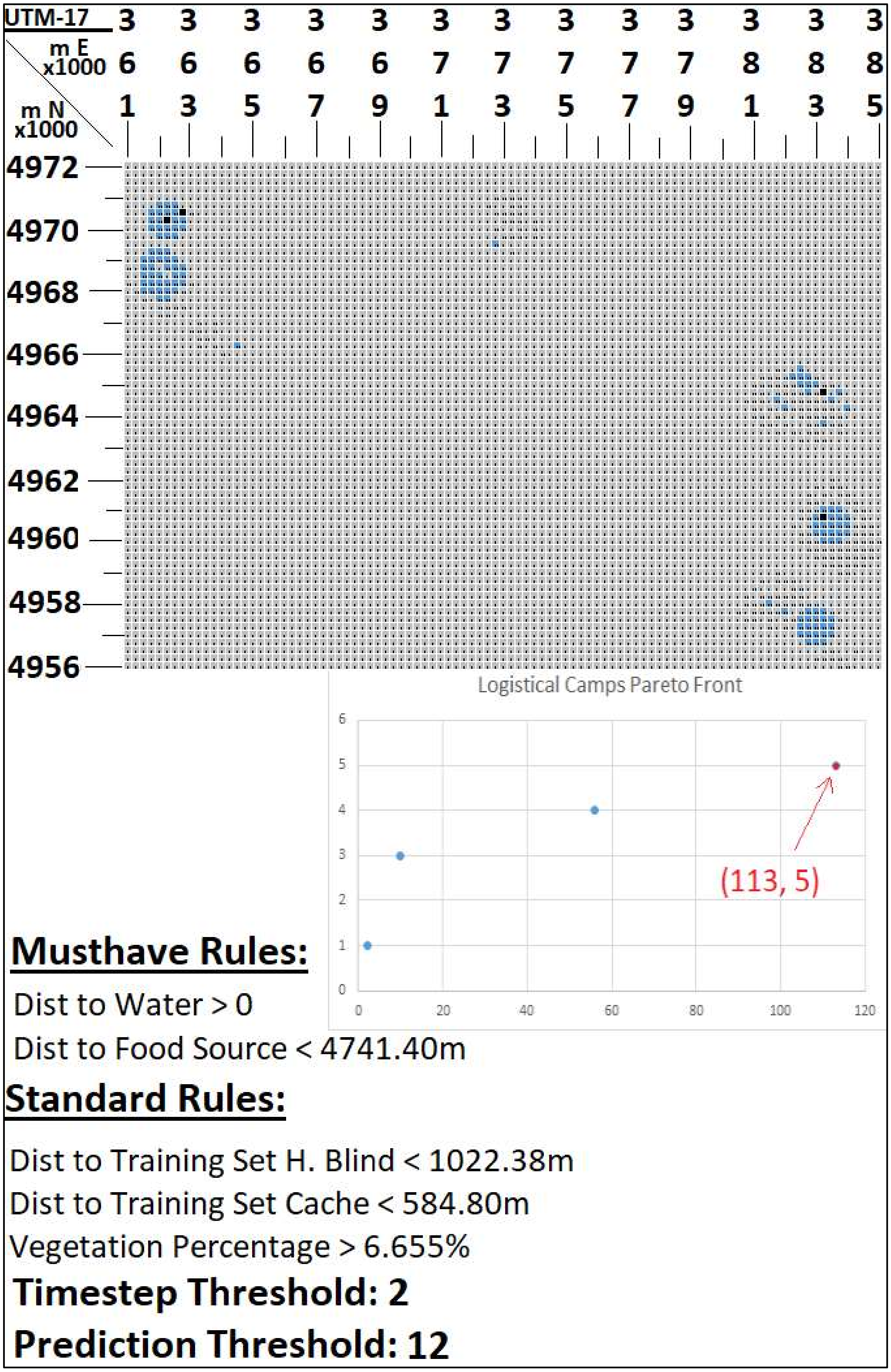


Figure 136: Logistical Camps Frame (Locations, Structures) = (113, 5)

8.5 Evaluating the Accelerating Cost Hypothesis

If the Accelerating Cost Hypothesis is true for a given structure type, then the Pareto Front for that structure type should follow a logarithmic pattern. A logarithmic pattern will signify that the cost (designated in terms of flagged locations that the model directs the archaeologist to search) of the benefit (designated in terms of training set Paleolithic structures found within those flagged locations) will increase at an increasing rate. If, on the other hand, the Accelerating Cost Hypothesis is false, then the cost of the benefit will increase at a constant rate. In other words, it should follow a linear pattern. We can thus test the Accelerating Cost Hypothesis by creating logarithmic regression models and linear regression models for each of the structure types, and then comparing the logarithmic regression model against the linear regression model for each of the structure types by means of F-tests.

8.5.1 Regression Curves

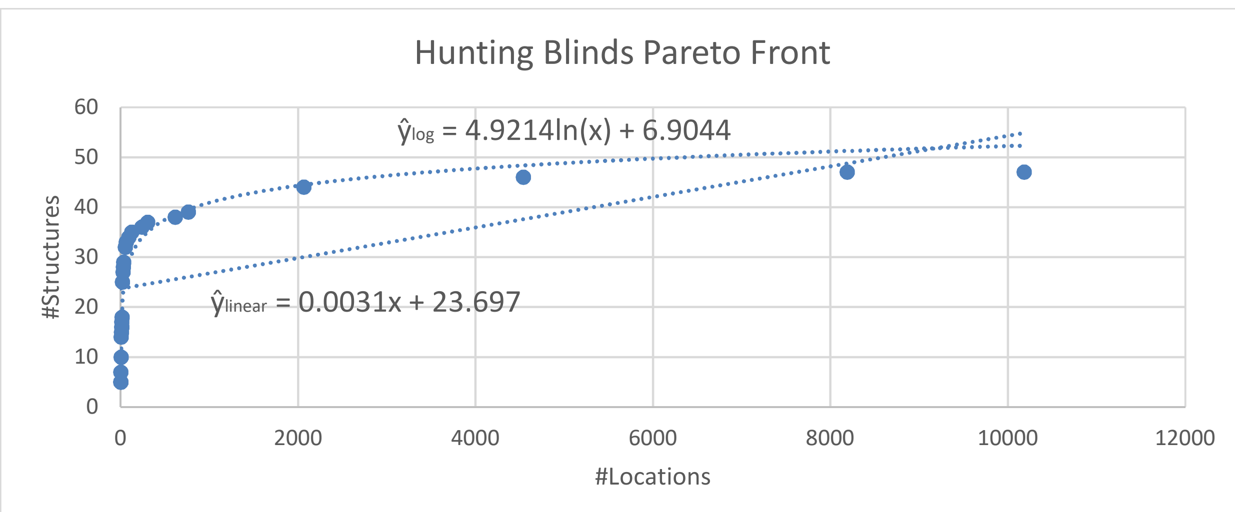


Figure 137: Hunting Blinds Pareto Front (Linear vs. Logarithmic Models)

Value \ Model	Logrithmic	Linear
MSM	27602273.98	27446863.58
MSE	22.09080679	116.1591822
F-stat	1249491.44	236286.6462
p	< 0.0007	< 0.002

Table 17: Logarithmic vs. Linear Regression Models – Hunting Blinds

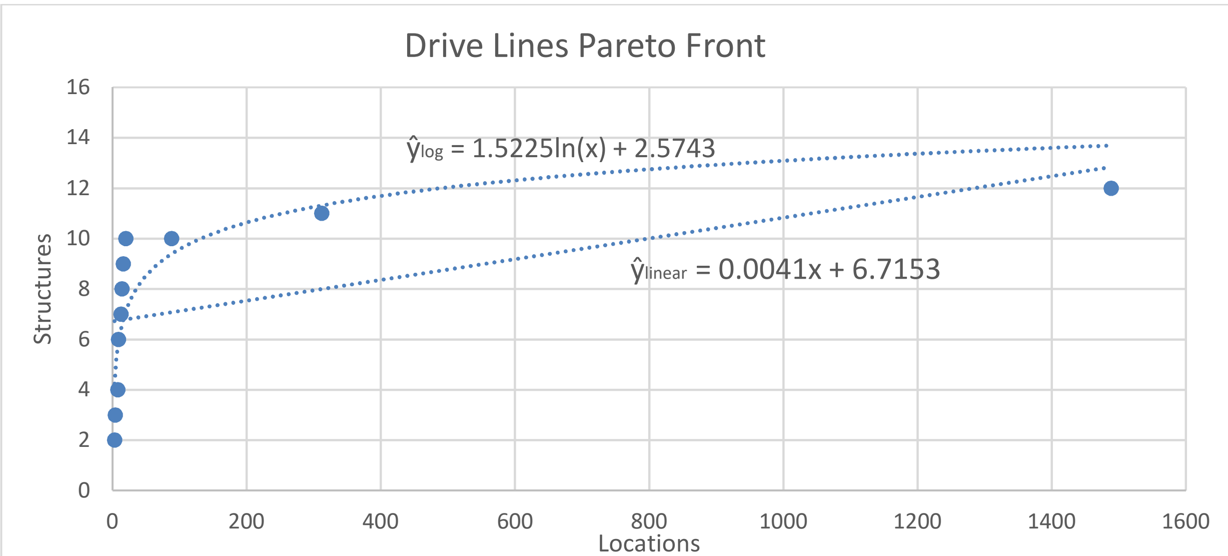


Figure 138: Drive Lines Pareto Front (Linear vs. Logarithmic Models)

Value \ Model	Logrithmic	Linear
MSM	326195.2227	326155.8185
MSE	2.959477	7.93901
F-stat	110220.5817	41082.68247
p	< 0.002	< 0.004

Table 18: Logarithmic vs. Linear Regression Models – Drive Lines

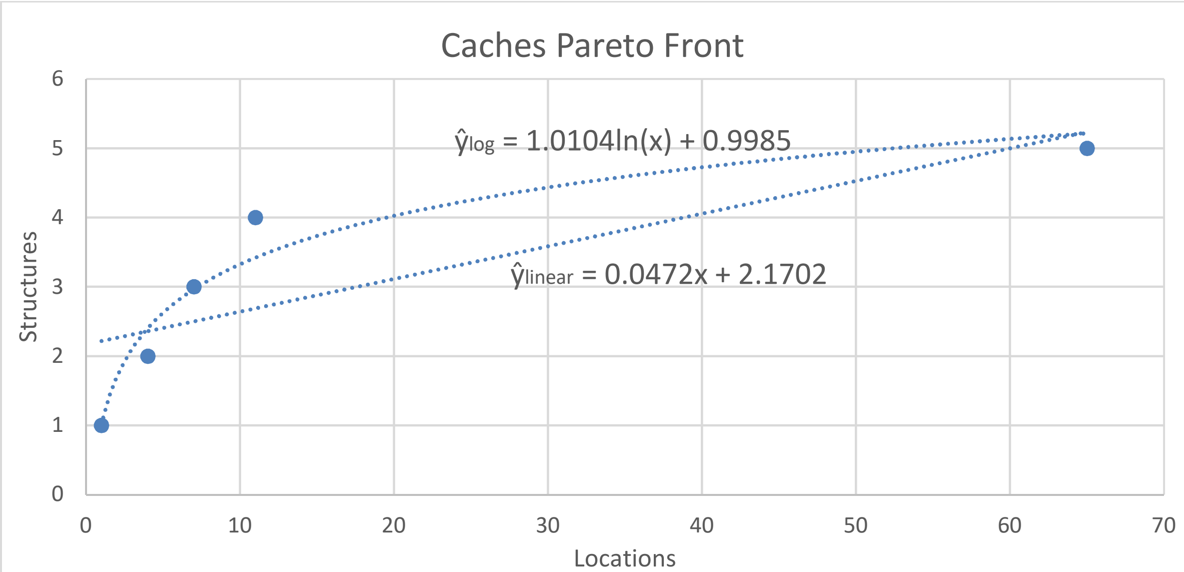


Figure 139: Caches Pareto Front (Linear vs. Logrithmic Models)

Value \ Model	Logrithmic	Linear
MSM	9.457709964	6.37875572
MSE	0.135565813	0.90868893
F-stat	69.76471253	7.019735257
p	< 0.09	< 0.23

Table 19: Logarithmic vs. Linear Regression Models – Caches

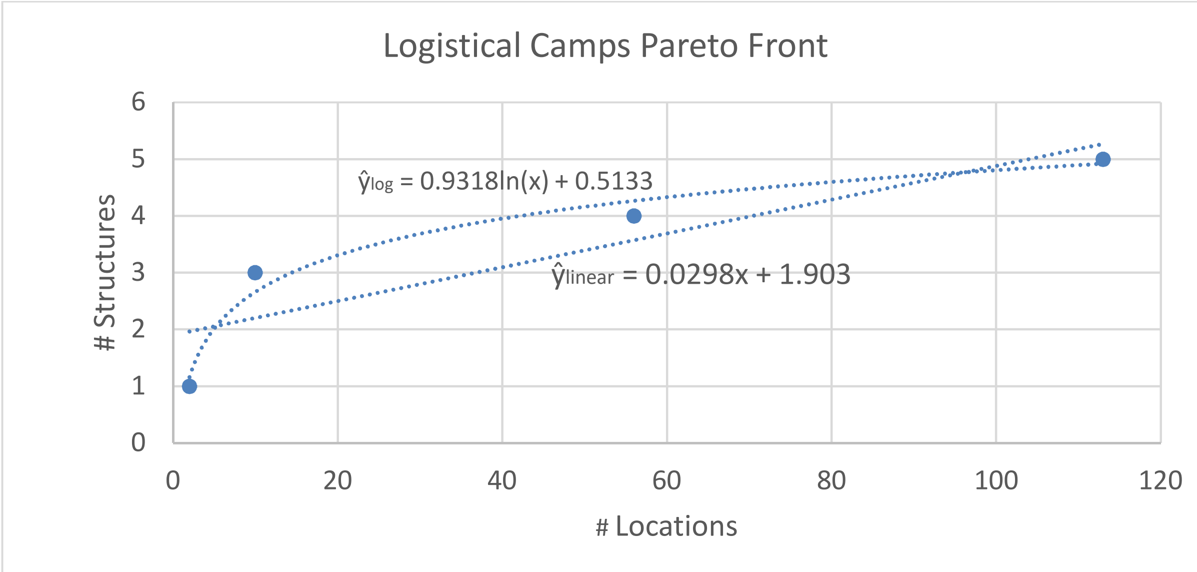


Figure 140: Logistical Camps Pareto Front (Linear vs. Logarithmic Model)

Value \ Model	Logrithmic	Linear
MSM	8.532613764	6.94337116
MSE	0.072719883	0.607157053
F-stat	69.76471253	7.019735257
p	< 0.07	< 0.21

Table 20: Logarithmic vs. Linear Regression Models – Logistical Camps

8.5.2 F-Tests Analysis

The F-tests for the Hunting Blind and Drive Line categories yielded a highly statistically significant correlation between the data and both the linear model ($p<0.002$ for Hunting Blinds and $p<0.004$ for Drive Lines) and the logarithmic model ($p<0.0007$ for Hunting Blinds and $p<0.002$ for Drive Lines). Although both model types yielded highly significant p-values, the p-values for the logarithmic model were substantially better in both cases. We therefore conclude that the analysis suggests that the Accelerating Cost Hypothesis is true for the Hunting Blind and Drive Line categories.

The Cache and Logistical Camp results were the most surprising, although the surprise was a very pleasant one. Before doing these F-tests, we thought that there might be a problem here due to the paucity of data for these two structure categories. However, paucity of data was unable to overcome how well the logarithmic model fit what data did exist for these two categories, and decent if not spectacular p-values ($p<0.09$ and $p<0.07$, respectively) were achieved in the F-tests of the logarithmic models for both of these categories, indicating that there is at least some correlation between the data and the logarithmic model for both of them. The linear models for the Cache and Logistical Camp categories, however, achieved p-values ($p<0.23$ and $p<0.21$, respectively) that were well worse than what would even arguably establish any decent correlation between the data and the model. Based on these results, we also affirm that the Accelerating Cost Hypothesis has been validated for the Cache and Logistical Camp categories as well.

8.6 Accelerating Cost Rates

We can do a comparative plot of the Pareto Fronts for the four structure types in order to compare the severity of the accelerating costs of each of them against each other (see Figure 141 and Figure 142). What we find out is that the less instances there are of a certain structure type, the greater the severity the Pareto curve for that structure type. More will be discussed on the implications of this fact in Chapter 8.

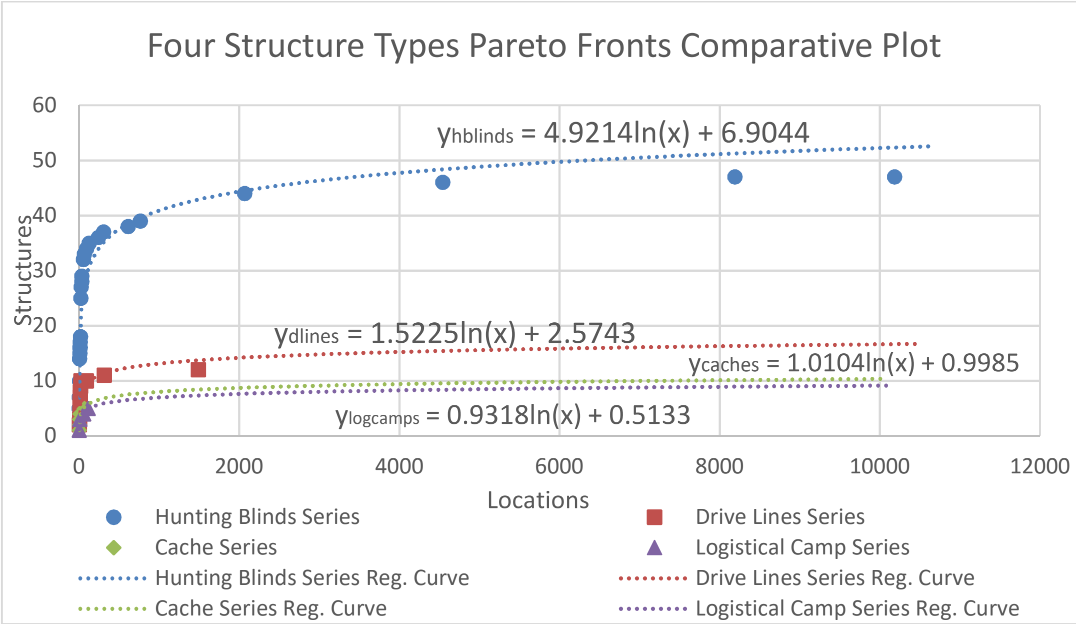


Figure 141: Four Structure Types Pareto Fronts Comparative Plot

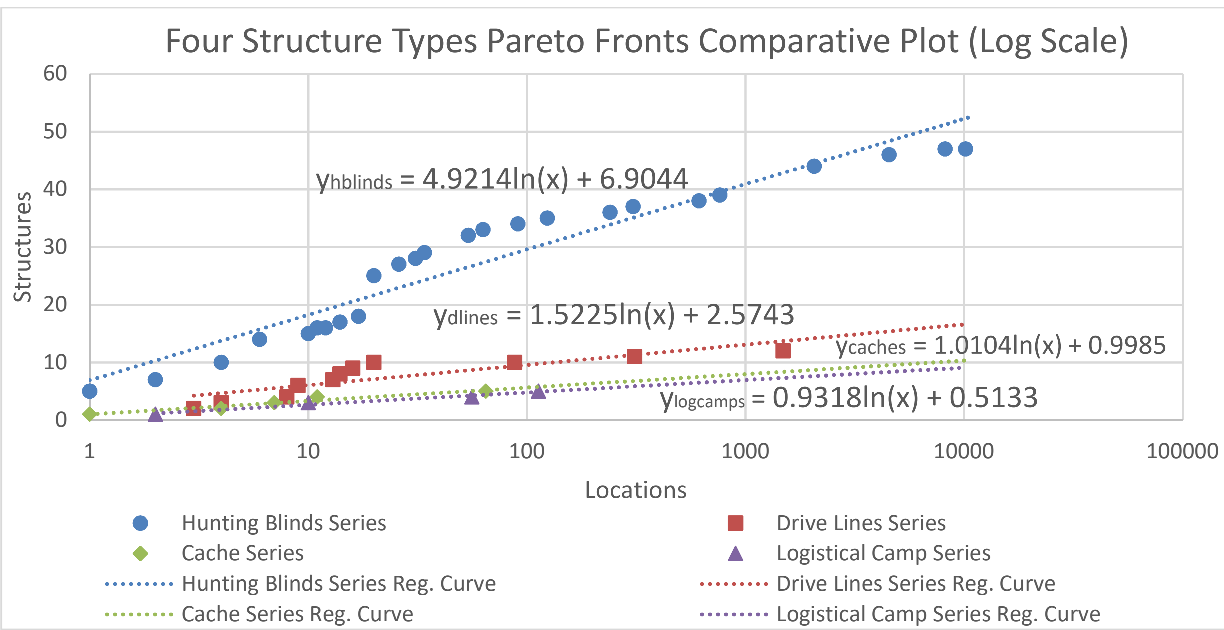


Figure 142: Four Structure Types Pareto Fronts Comparative Plot (Log Scale) with Regression Curves

8.7 Conclusions and Ruleset Size vs. Problem Complexity Hypothesis

The Hunting Blind structure type contained two Musthave rules, four Standard rules, and two thresholds. The Drive Line structure type contained two Musthave rules, two Standard rules, and two thresholds. The Logistical Camp structure type contained two Musthave rules, two Standard rules, and two thresholds. The Cache structure type contained two Musthave rules, three Standard rules, and two thresholds.

For the Drive Line structure type, there were 33 threads in the first run. Looking at the Learning Curves and Knowledge Source Dominance Graph, an exploitative source, Historical Knowledge, dominated up until the end. For both the Logistical Camp and Cache structure types, there was 1 thread in the first run, and Normative Knowledge, an explorative knowledge source, dominated the entire time.

The Hunting Blind structure type provided the most interesting behavior, probably due to the fact that there were far more Hunting Blinds than any other structure type in the training set. For the Hunting Blind structure type, there were 129 threads in the first run. Looking at the Learning Curves and Knowledge Source Dominance Graph, an explorative source, Normative Knowledge, dominated up until Generation 350 and then an exploitative source, Historical Knowledge, made a “giant leap” and dominated from then on out.

From these results, we submit that we have shown that complex behavior is possible even with a relatively small ruleset. We thus submit that we have demonstrated the veracity of the Ruleset Size vs. Problem Complexity Hypothesis.

CHAPTER 9: PLANNING AN EXPEDITION SEASON

We now arrive at the task of taking our results from the previous chapter and using them to create a consolidated heatmap that can be used to plan an entire expedition season. This is done by combining, from each of the four categories, one of the images located in the *Hunting Blinds Frames*, *Drive Lines Frames*, *Caches Frames*, and *Logistical Camps Frames* sections from the previous chapter respectively. Due to the nature of Pareto-optimality, there is no single image the previous chapter which is objectively "better" than any other single image from the same Pareto Front. When combined with the fact that the four different Pareto Fronts are of four different structure types that have differing degrees of value to different archaeologists and different expeditions, it means that there is no way to automate the final step of choosing the four-different category-images to consolidate into a single expedition heatmap: A human judgment call must be made in deciding which individual category-images to combine into a full consolidated image that can be used to plan an entire expedition season.

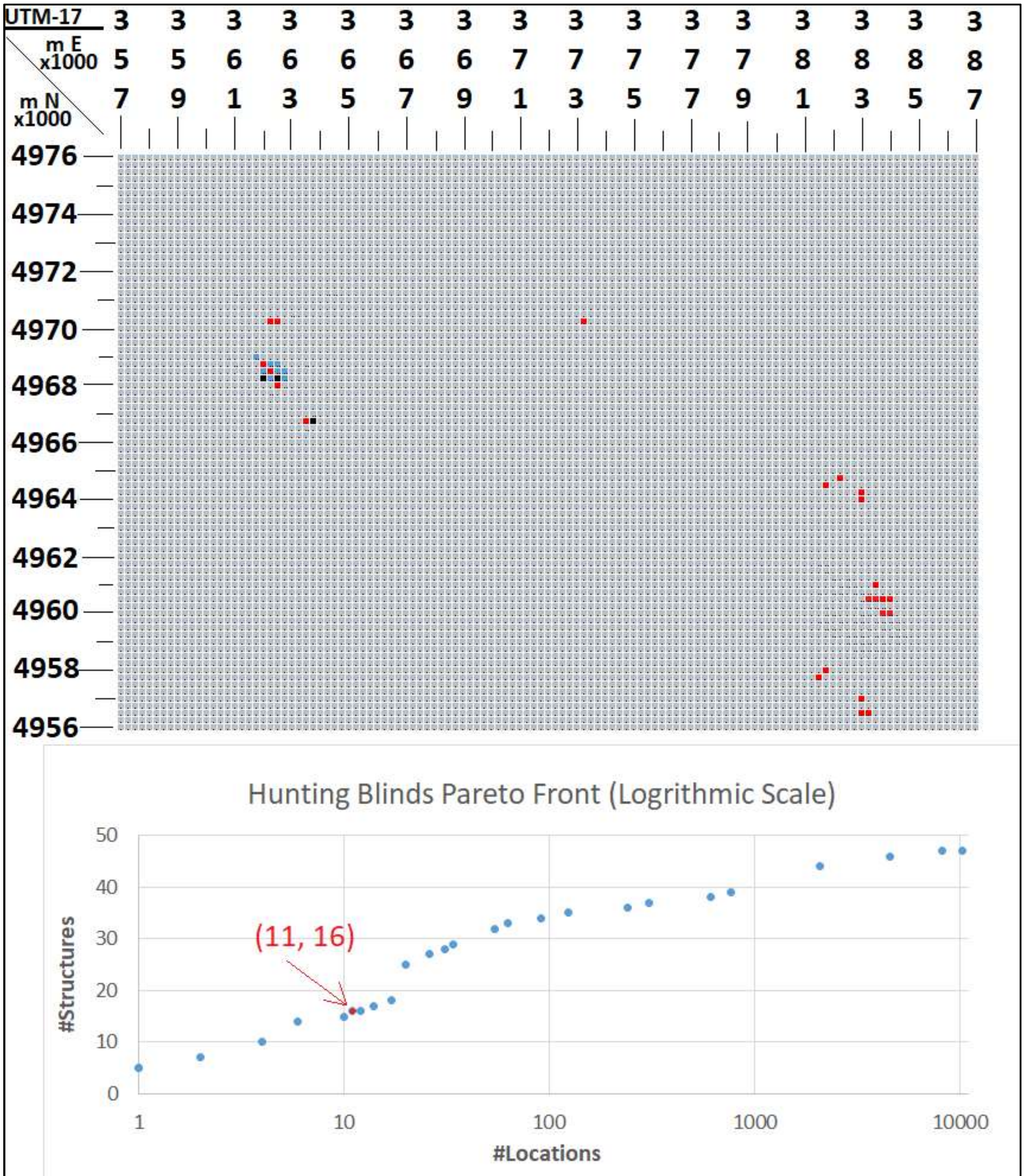
For demonstration purposes, we will dedicate the rest of this chapter to creating several "candidate heatmaps" by combining images corresponding to interesting Pareto points located in prominent places throughout their various distributions. We will create the first of these "candidate heatmaps" by imagining a scenario of a hypothetical archaeological expedition which desires most of all to find one or more logistical camps, and values other artifact types to a significantly lesser degree. For the fourth "candidate heatmap", we will imagine a scenario of an archaeological expedition which sees caches as the most valuable structure type, logistical camps as the second-most valuable structure type, drive lines are the third most valuable structure type, and hunting blinds are the fourth most valuable structure type. For the third scenario, we will imagine a hypothetical archaeological expedition which desires most of all to find drive lines, followed by caches, followed by logistical camps, then finally by hunting blinds.

9.1 Candidate Heatmap from Scenario 1

Scenario 1 involves a hypothetical team of archaeologists (we can call them "Team 1"), which prizes logistical camps above all else. They would still value finding a hunting blind, drive line, or cache, but far above all they want to find a logistical camp.

9.1.1 Team 1's Selection

With their goals and priorities in mind, Team 1 chooses the following frames (Figure 143-Figure 146) with which to create a composite. These frames can be found on the following pages.



Musthave Rules:

Dist to Water > 0m   Dist to Caribou < 1749.16m

Standard Rules:

|Height Above/Below Caribou| < 19.09m Vegetation Percentage > 22.74%

Dist to Training Set Drive Line < 364.50m Dist to T. Set Log Camp < 2107.43m

Timestep Threshold: 3   Prediction Threshold: 5

Figure 143: Hunting Blinds Frame (Locations, Structures) = (11, 16)

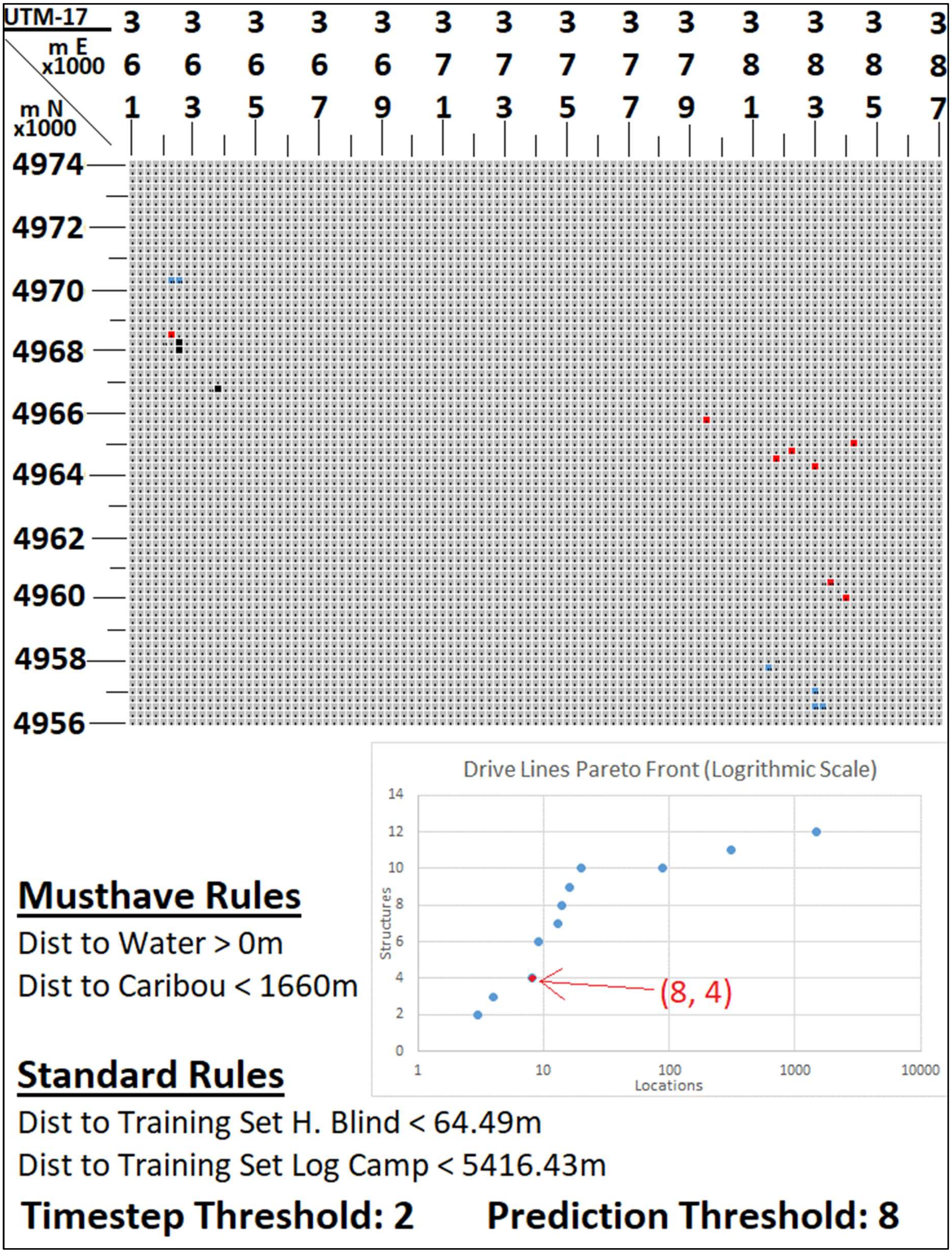


Figure 144: Drive Lines Frame (Locations, Structures) = (8, 4)

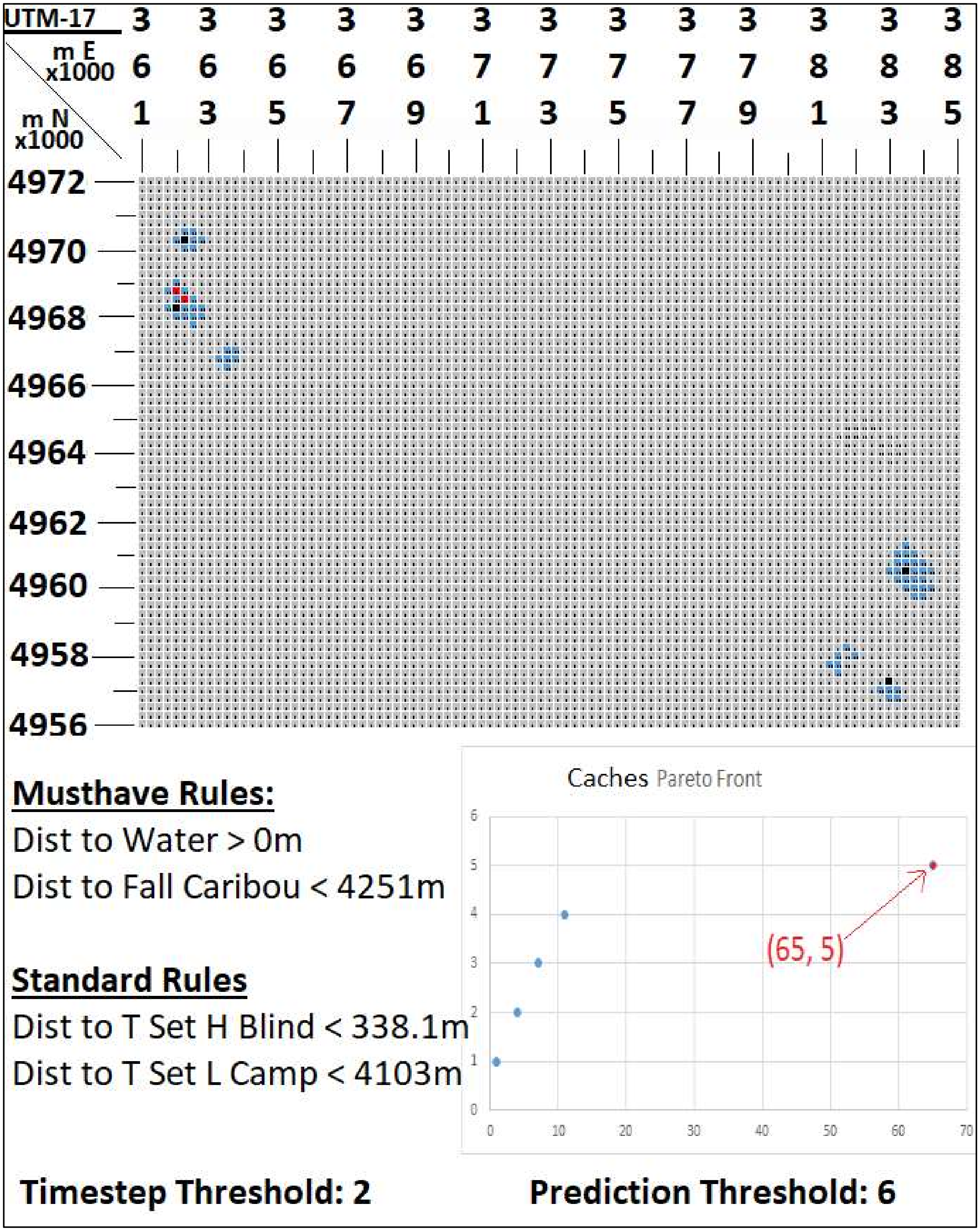
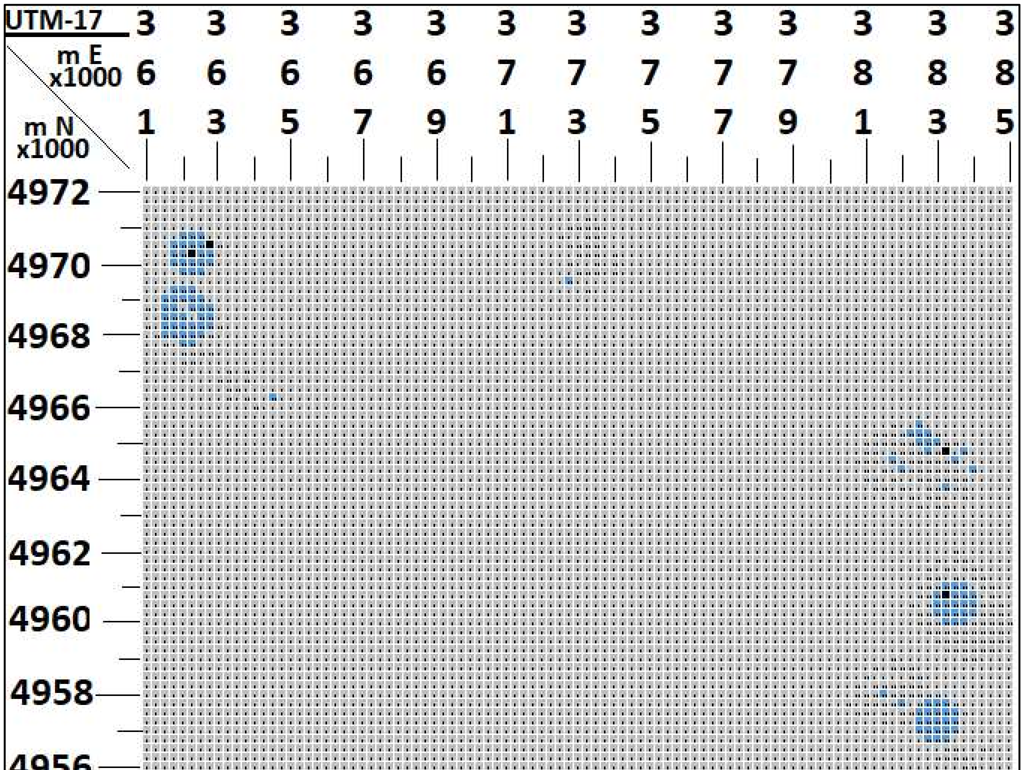
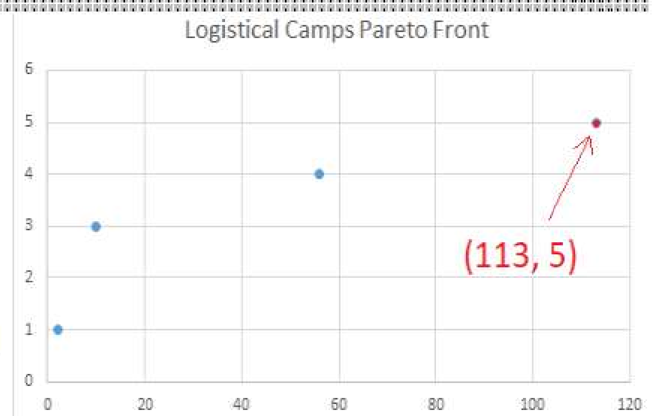


Figure 145: Caches Frame (Locations, Structures) = (4, 2)





Musthave Rules:

Dist to Water > 0

Dist to Food Source < 4741.40m

Standard Rules:

Dist to Training Set H. Blind < 1022.38m

Dist to Training Set Cache < 584.80m

Vegetation Percentage > 6.655%

Timestep Threshold: 2

Prediction Threshold: 12

Figure 146: Logistical Camps Frame (Locations, Structures) = (113, 5)

9.1.2 Scenario 1 Composite

Team 1 then composites the images in Figure 143-Figure 146, producing Figure 147, which they then use to help plan their expedition season:

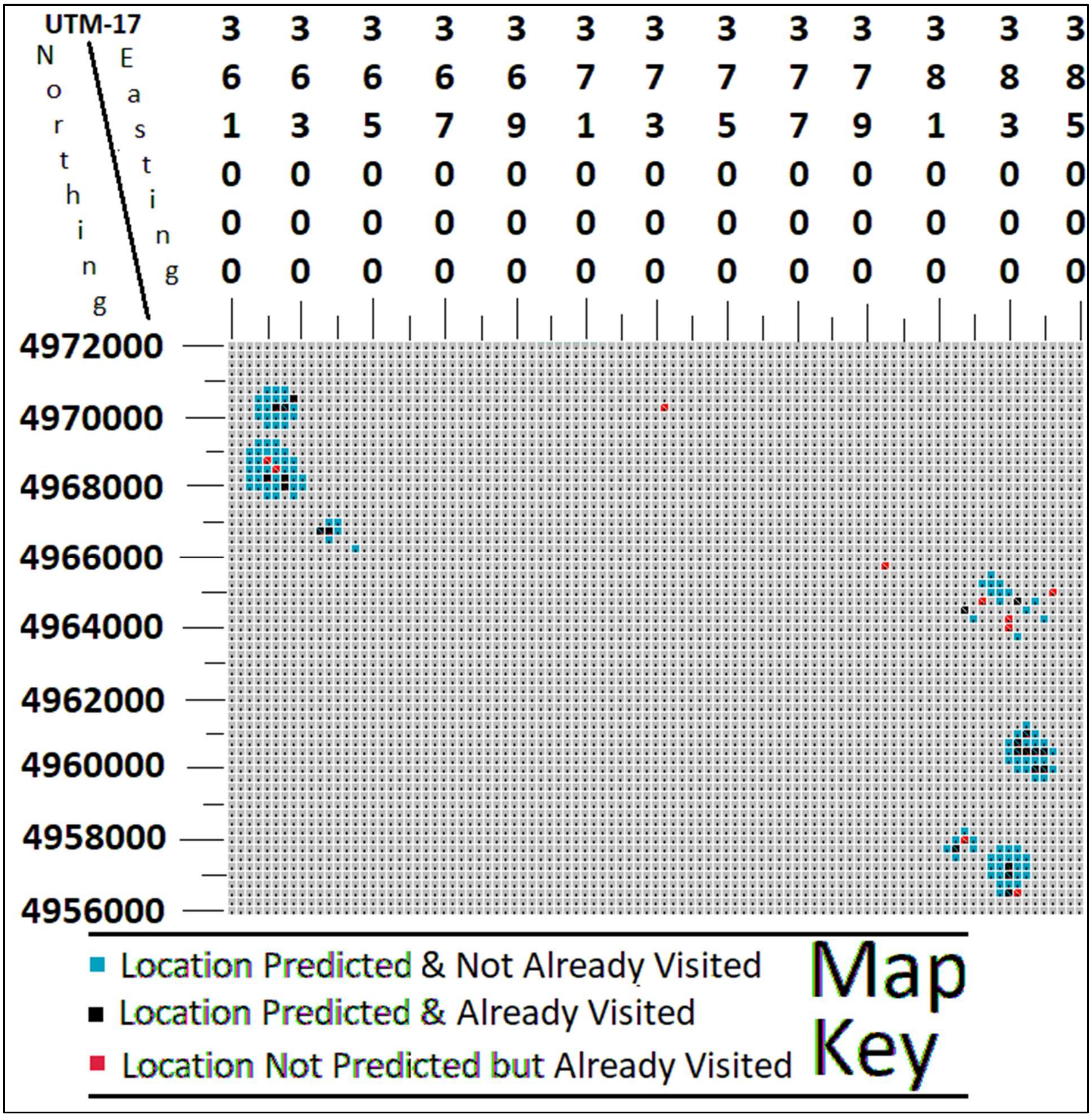


Figure 147: Scenario 1 Composite

9.2 Scenario 2

Scenario 2 involves a hypothetical archaeological expeditionary team which sees caches as the most valuable structure type, logistical camps as the second-most valuable structure type, drive lines are the third most valuable structure type, and hunting blinds are the fourth most valuable structure type. We will call this team “Team 2”.

9.2.1 Team 2’s Selections

With their goals and priorities in mind, Team 2 selects Figure 148-Figure 151 as their constituent images from which to form a whole-season composite. These can be found on the following pages.

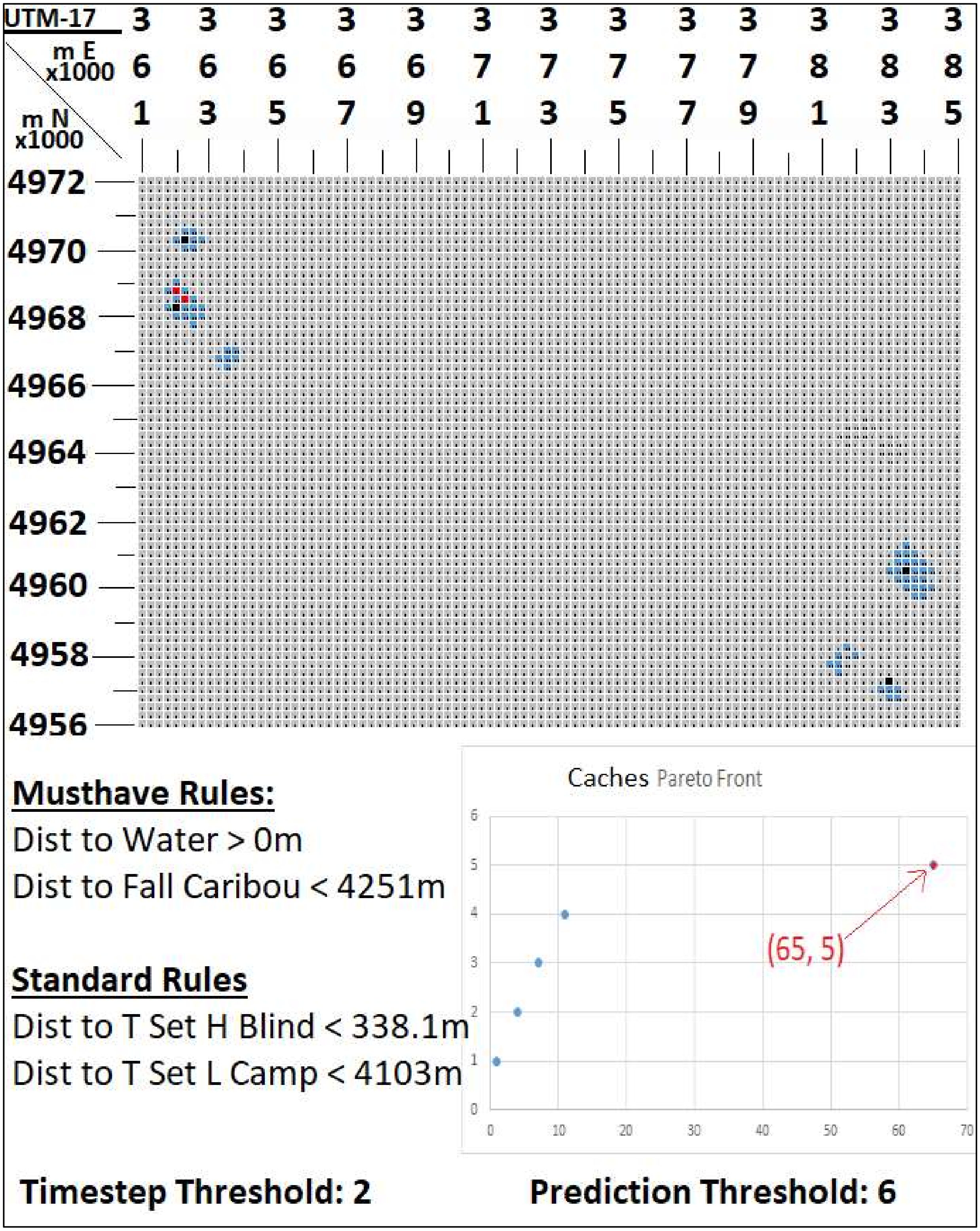
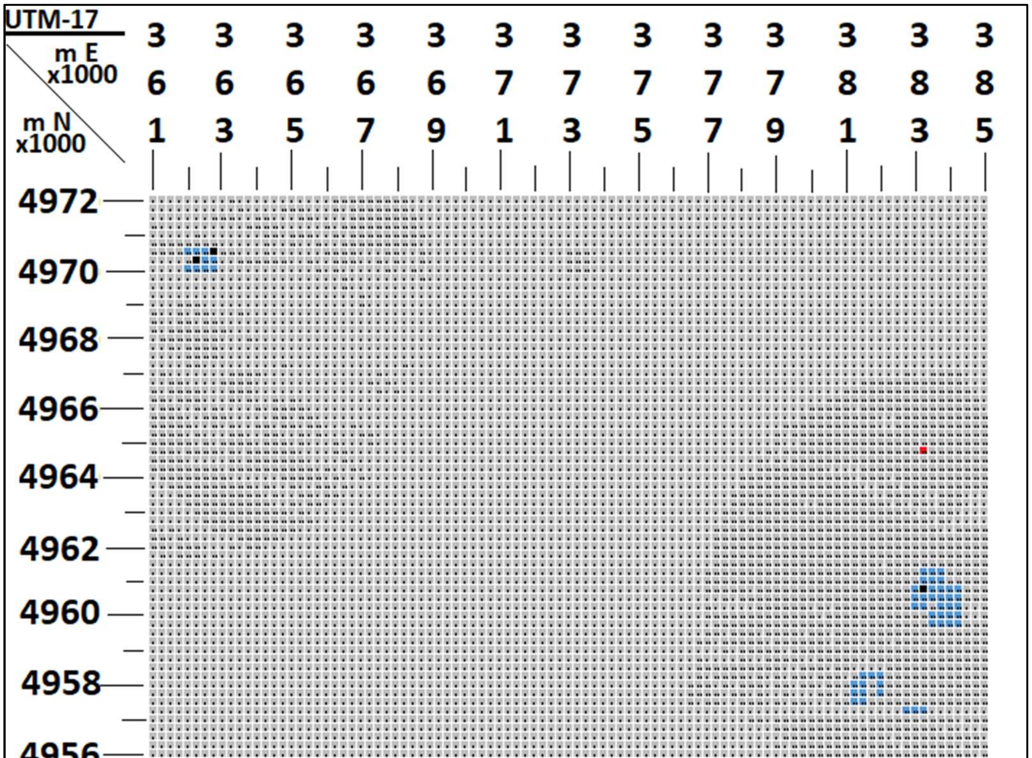
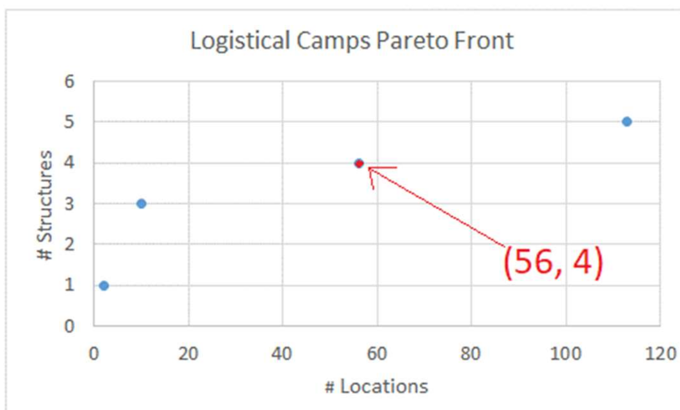


Figure 148: Caches Frame (Locations, Structures) = (65, 5)





Musthave Rules:

Dist to Water > 0 m

Dist to Food Source < 5337.36m

Standard Rules:

Dist to Training Set H. Blind < 399.27m

Dist to Found Cache Rule < 6568.21m

Vegetation Percentage > 39.097%

Timestep Threshold: 2

Prediction Threshold: 18

Figure 149: Logistical Camps Frame (Locations, Structures) = (56, 4)

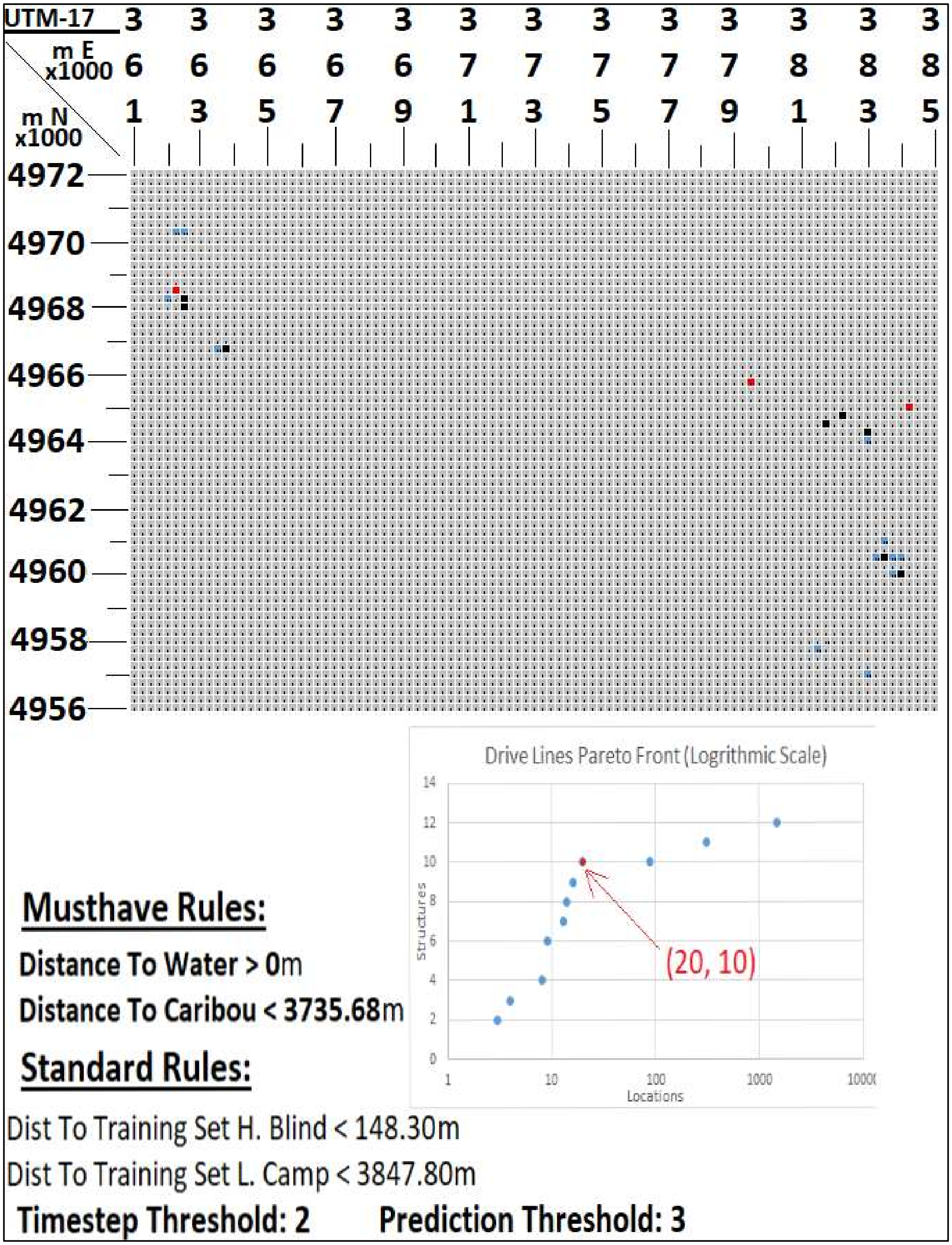
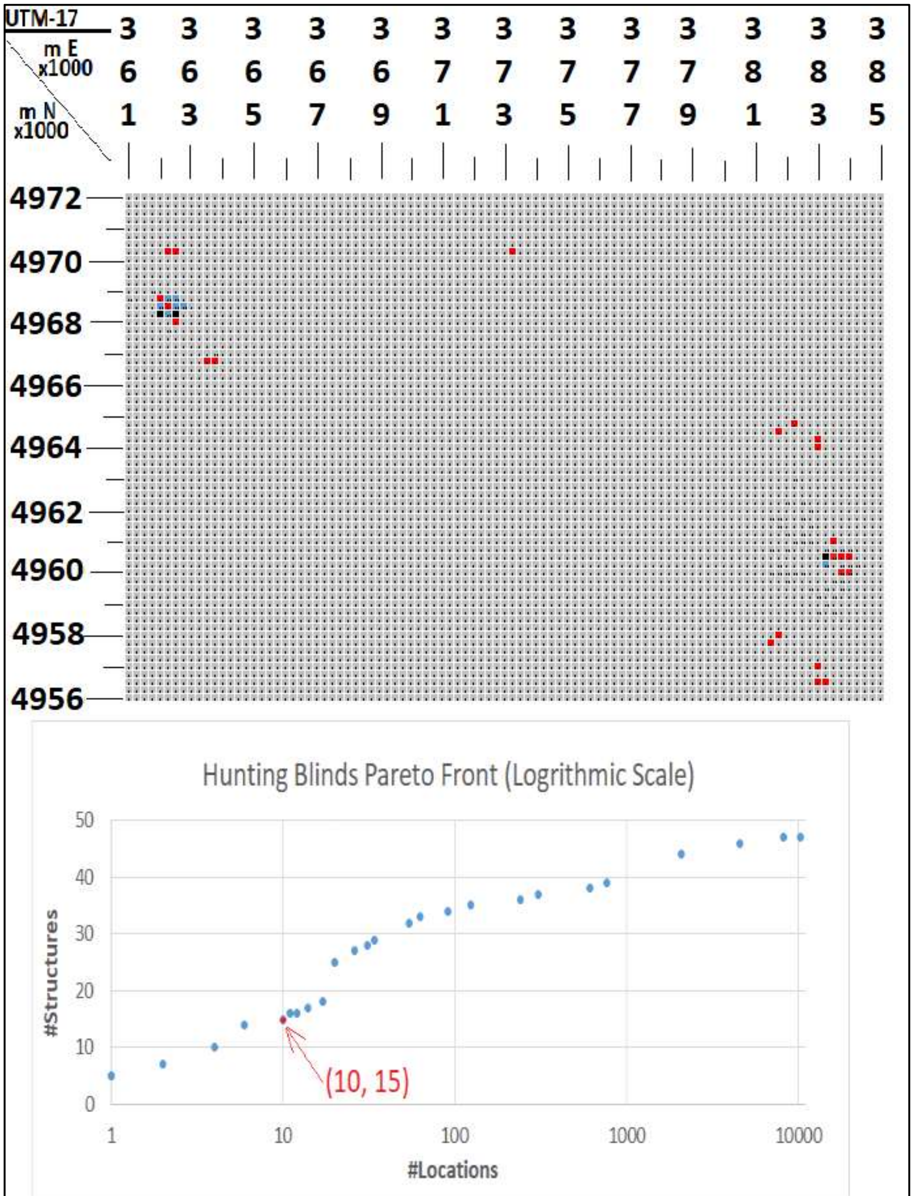


Figure 150: Drive Lines Frame (Locations, Structures) = (20, 10)



Musthave Rules:

Dist to Water > 0m   Dist to Caribou < 2546.11m

Standard Rules:

|Height Above/Below Caribou| < 2.18m   Vegetation Percentage > 64.07%

Dist to Training Set Drive Line < 474.92m   Dist to T. Set Log Camp < 2057.45m

Timestep Threshold: 3   Prediction Threshold: 6

Figure 151: Hunting Blinds Frame (Locations, Structures) = (10, 15)

9.2.2 Scenario 2 Composite

Team 2 then composites the images in Figure 148-Figure 151, producing Figure 152, which they then use to help plan their expedition season.

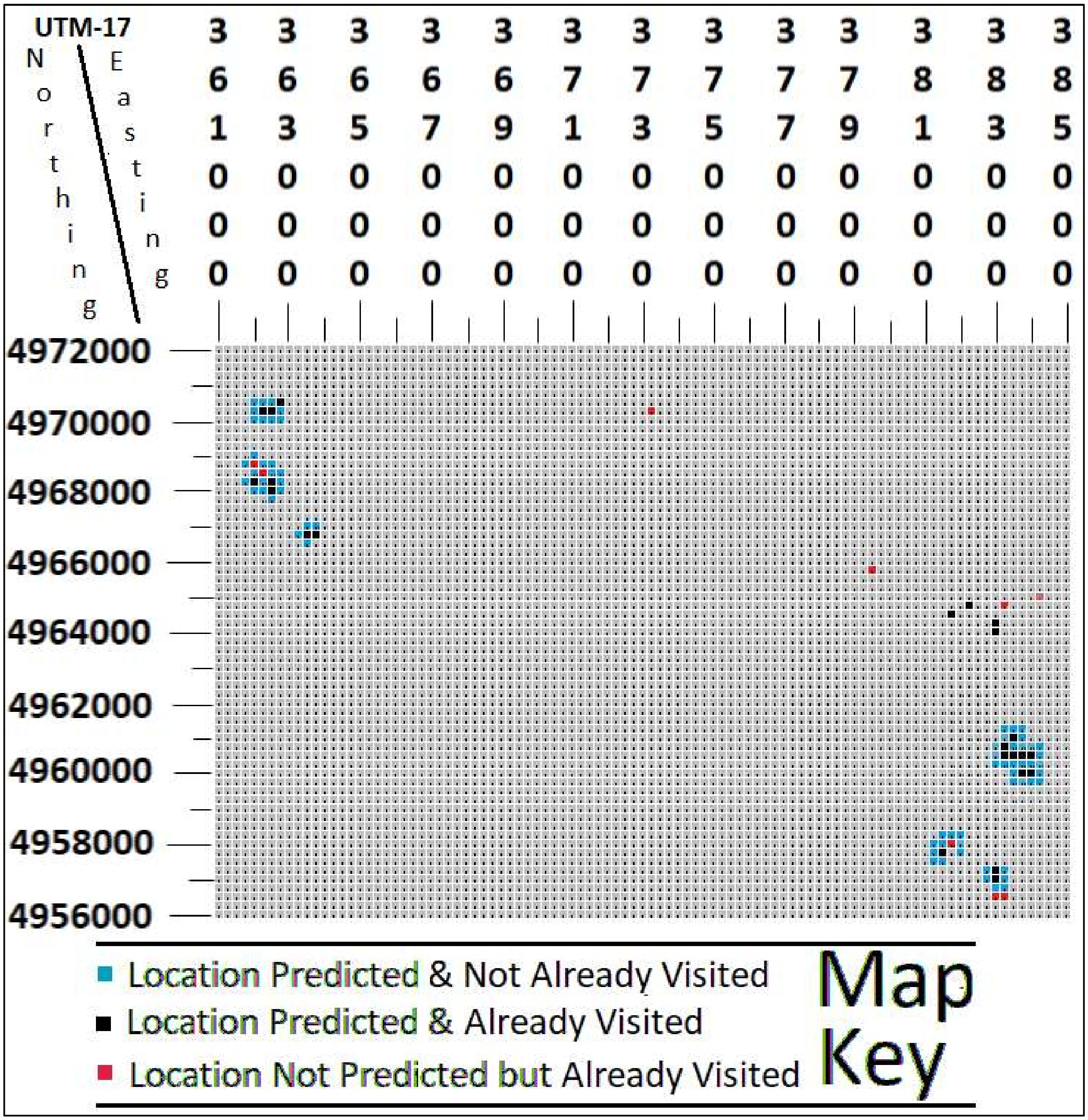


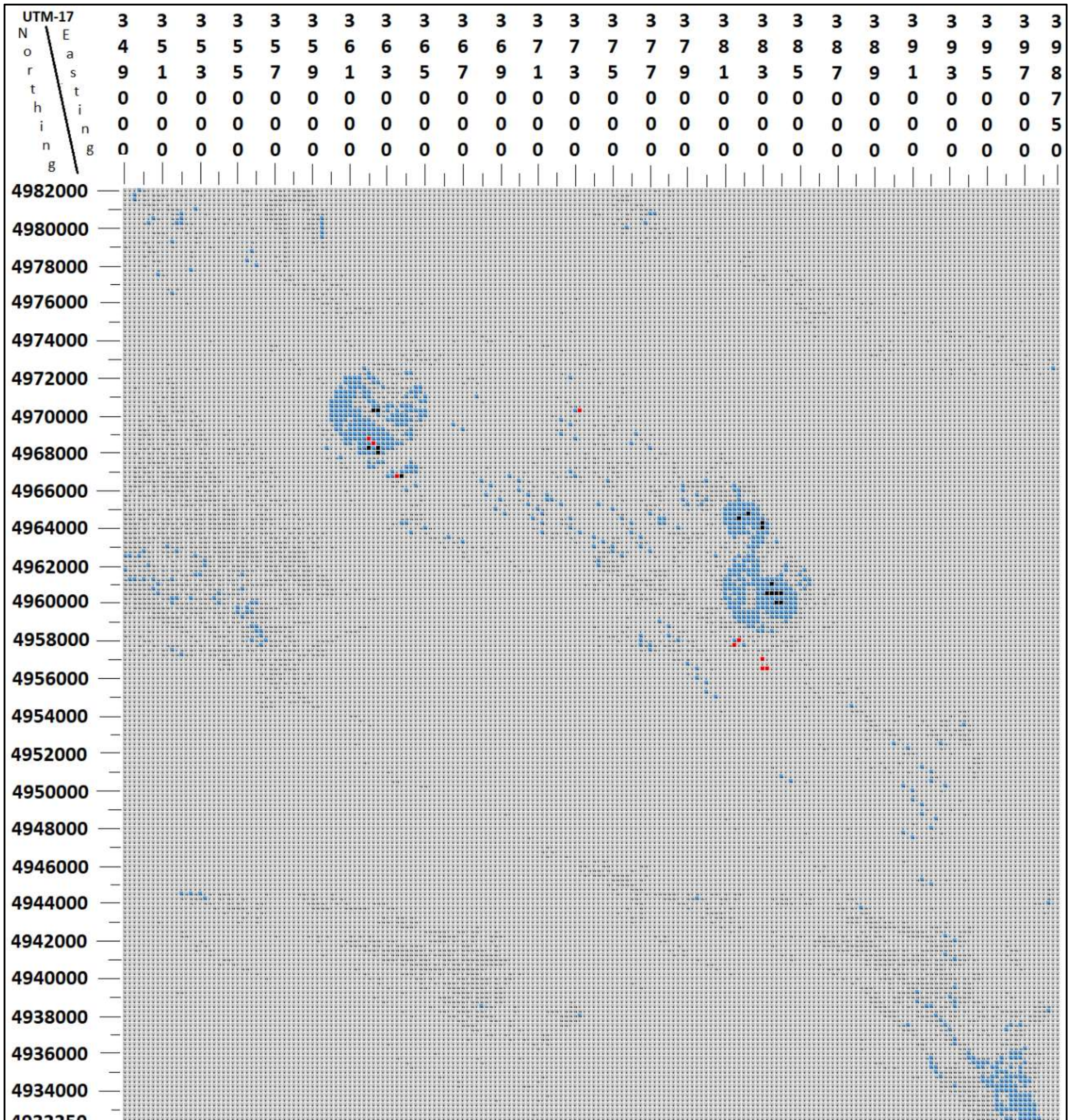
Figure 152: Scenario 2 Composite

9.3 Scenario 3

The third scenario involves a hypothetical archaeological expedition which desires most of all to find logistical camps, followed by hunting blinds, followed by caches, then finally by drive lines. We will also assume that this hypothetical expeditionary team is a group of archaeologists with a significantly larger time, money, and manpower budget than the hypothetical teams in Scenario 1 and Scenario 2. We can call them "Team 3".

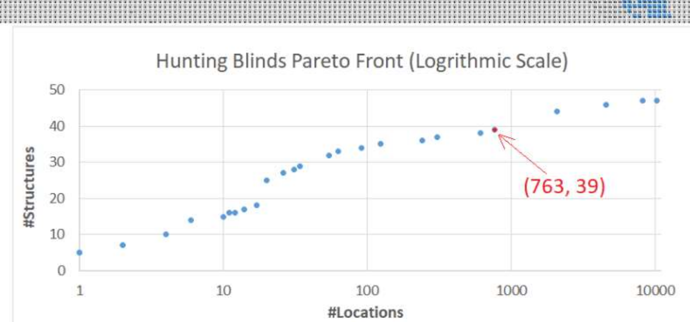
9.3.1 Team 3's Selections

With their goals and priorities in mind, Team 3 selects Figure 153-Figure 156 as constituent frames in order to produce an eventual composite. Their selections can be seen on the following pages.



Musthave Rules:
Dist to Water > 0m
Dist to Caribou < 3914.95m

Standard Rules:
|Height Above/Below Caribou| < 1.41m
Vegatation Percentage > 45.30%
Dist to Training Set Drive Line < 963.12m
Dist to Training Set Log Camp < 2310.22m



Timestep Threshold: 2 **Prediction Threshold: 3**

Figure 153: Hunting Blinds Frame (Locations, Structures) = (763, 39)

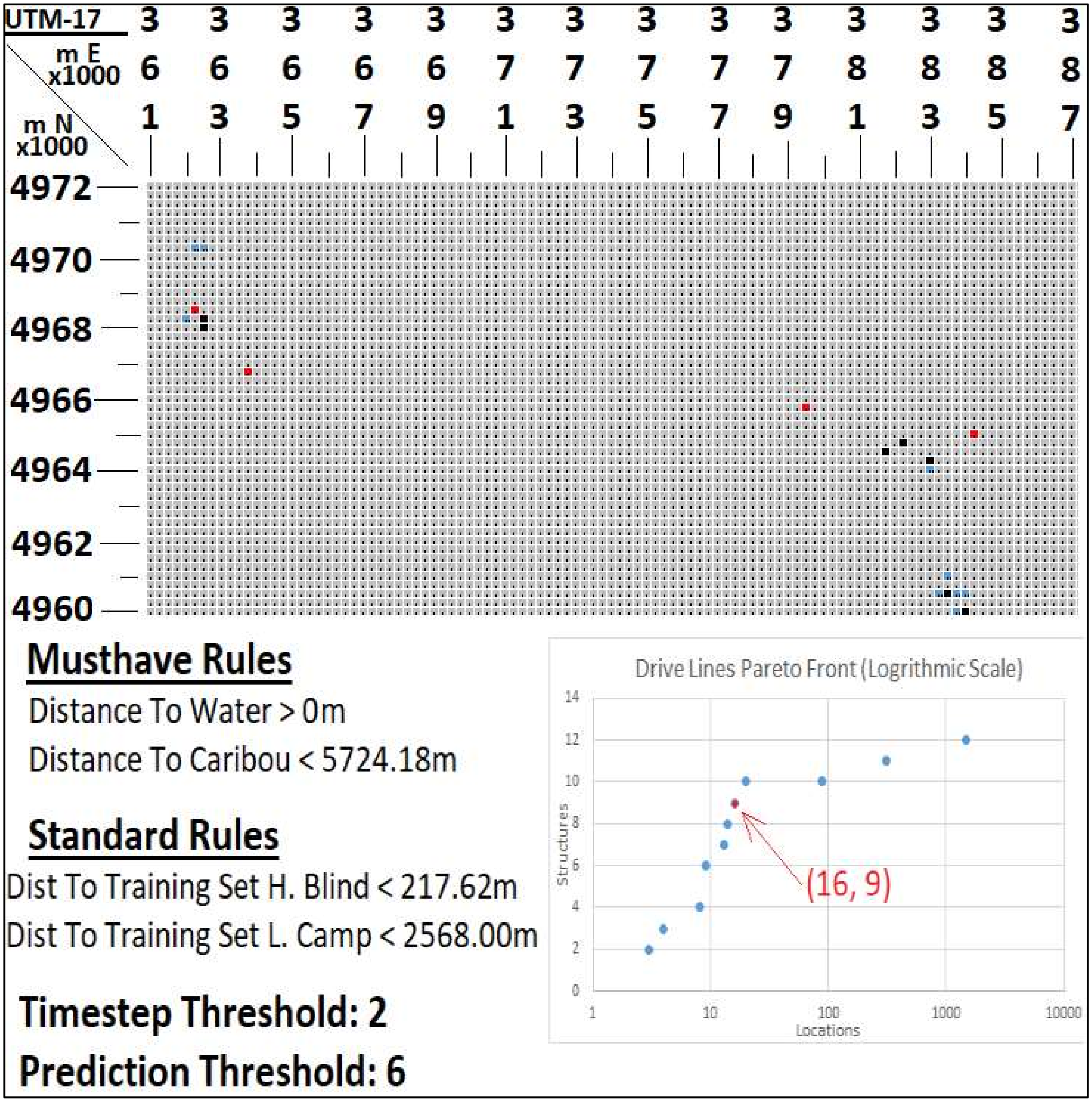


Figure 154: Drive Lines (Structures, Locations) = (16, 9)

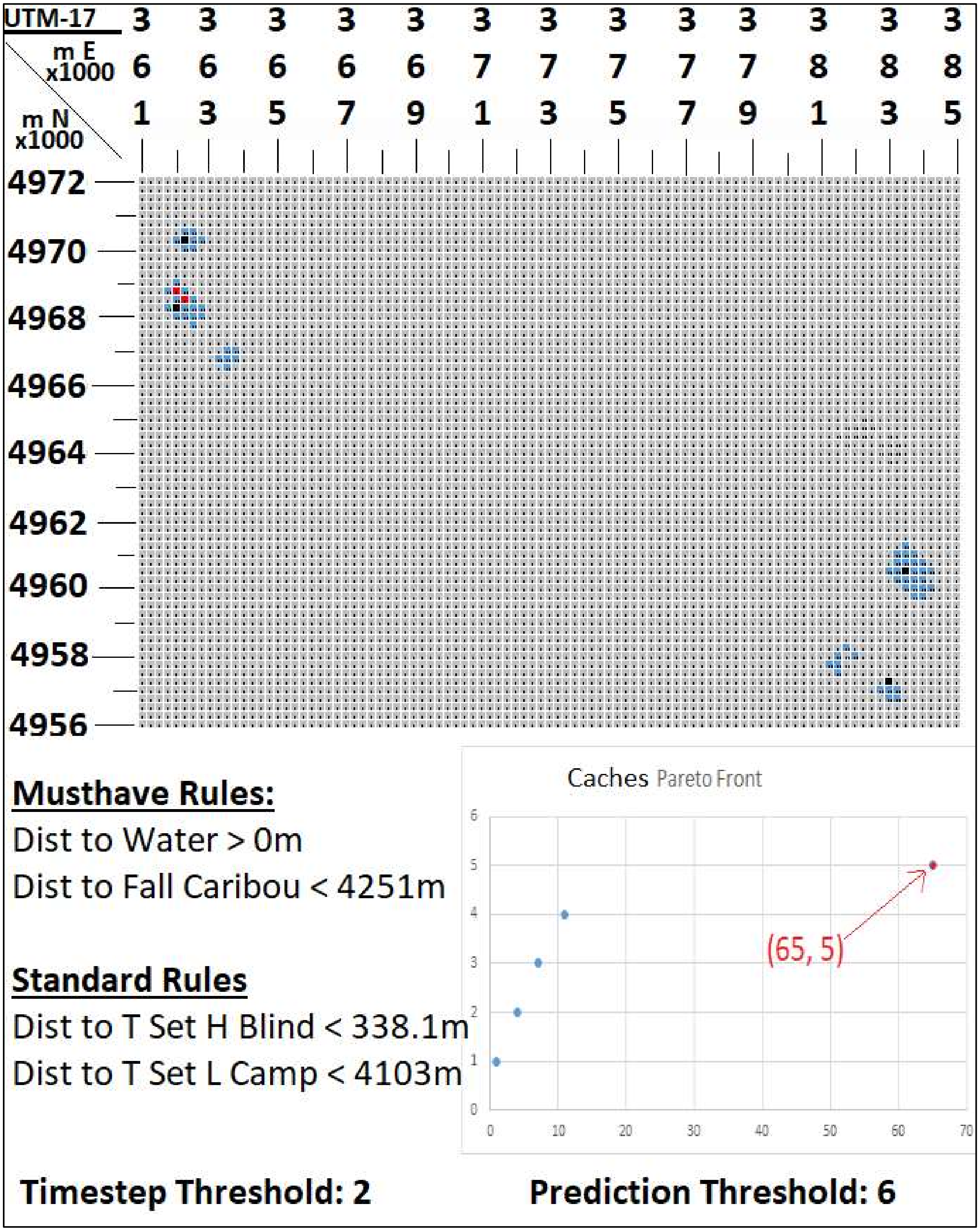
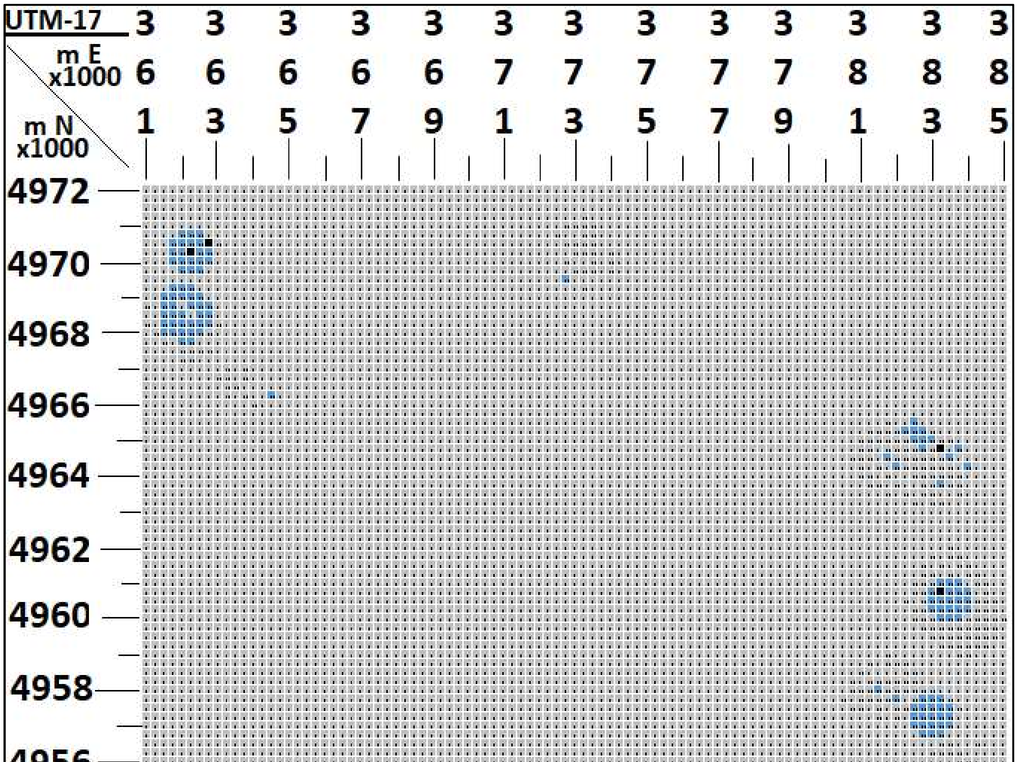
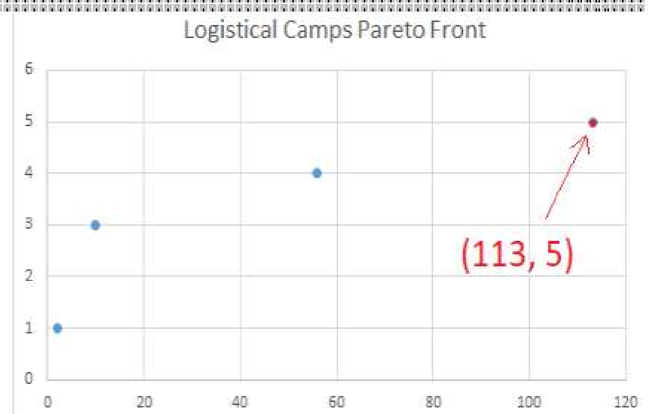


Figure 155: Caches Frame (Locations, Structures) = (65, 5)





Musthave Rules:

Dist to Water > 0

Dist to Food Source < 4741.40m

Standard Rules:

Dist to Training Set H. Blind < 1022.38m

Dist to Training Set Cache < 584.80m

Vegetation Percentage > 6.655%

Timestep Threshold: 2

Prediction Threshold: 12

Figure 156: Logistical Camps Frame (Locations, Structures) = (113, 5)

9.3.2 Scenario 3 Composite

Team 3 then composites Figure 153-Figure 156, producing Figure 157 for their season.

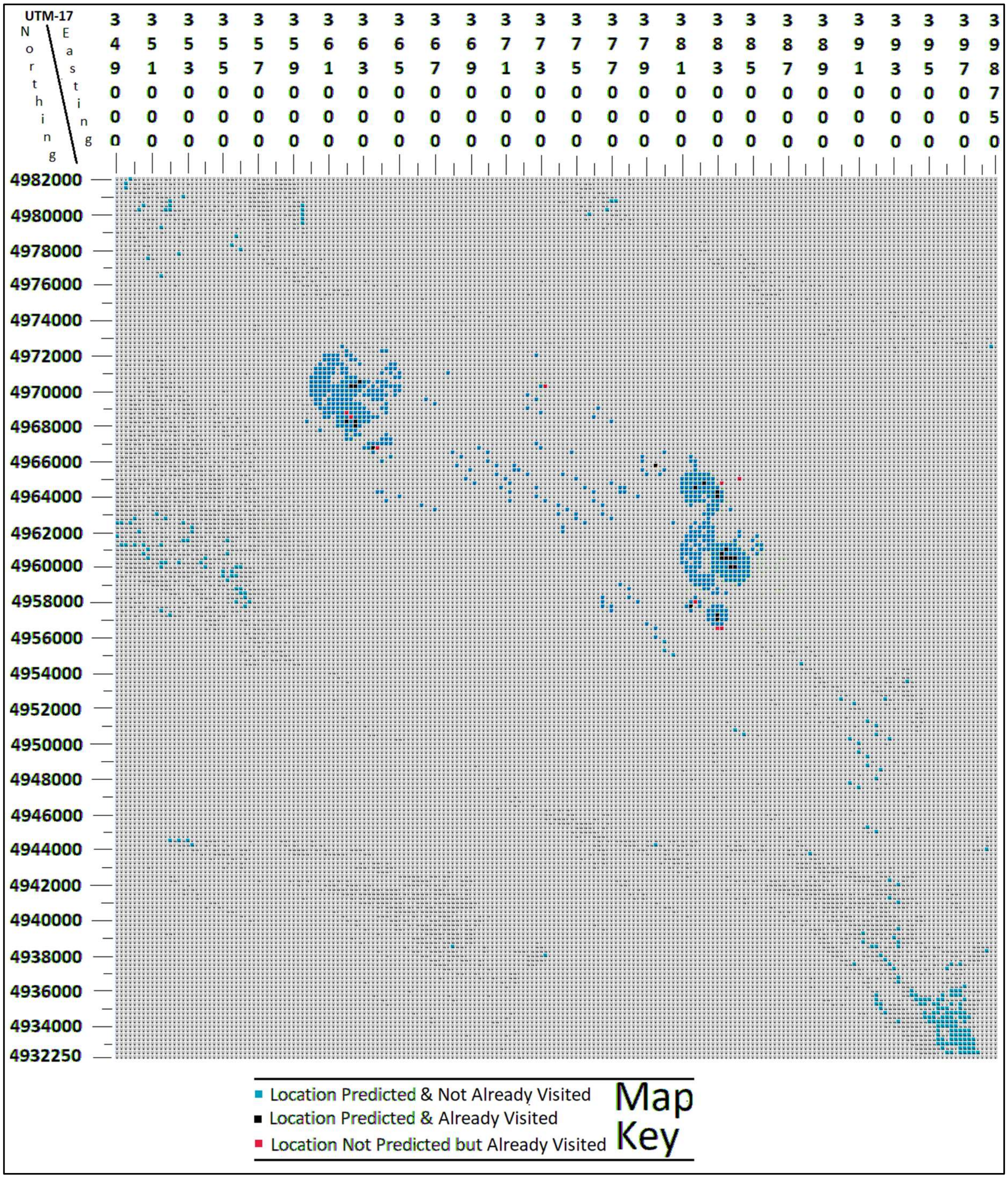


Figure 157: Scenario 3 Composite

9.4 Conclusions Concerning the Accelerating Cost Hypothesis

The fact that the Accelerating Cost Hypothesis is true is ultimately a result of the fact that we are dealing with incomplete information, which itself results from the fact that the objects of study within the Land Bridge Project are Paleolithic structures that were built thousands of years before recorded history. No matter how brilliant the archaeologists' work is, and no matter how brilliant our computer models are, perfect information regarding every Paleoindian structure that ever existed upon what once was the Alpena-Amberley Land Bridge is never going to be achieved.

That being said, it can be reasonably postulated that for any given structure type, the more examples of that structure type that have been discovered, the more information that can be added to the overall body of knowledge about that structure type as a whole, since each individual structure provides us with information such as its elevation, relation to the environment, relation to other structures, etc., which can be added to the consolidated body of knowledge about its structure type as a whole. In Figure 141 - Figure 142, the consistent pattern is that the more examples that exist of any given structure type, i.e. the more information that exists about any given structure type, the less severe is the accelerating cost curve for that structure type.

9.5 Connecting the Accelerating Cost Hypothesis and the Composite Results

The predicted locations in composites consisting of lower numbers of predicted locations (i.e., in composites consisting of lower-cost frames) are almost always in geographical association with previously discovered structures. This is markedly the case in Scenario 1 and Scenario 2 in the previous chapter. In other words, in composite solutions with lower numbers of predicted locations, the locations that *are* predicted almost always "piggyback" off of the locations of structures discovered in previous archaeological expeditions. This is because the least cost-intensive way to set about discovering a new structure is to search compelling unsearched locations around existing structures in hopes of finding a previously undiscovered structure that is associated with the existing structures in some way. Hence, when asked to produce maps with lower number of predicted locations (i.e., lower *cost*), the system will produce maps that are mostly filled with predicted locations that are in close geographic association with already found structures.

The term "accelerating cost" can sound like paying the higher end of it is always a bad decision. However, this is not always the case. Frames on the higher end of the accelerating cost curve often contain significant numbers of predicted locations that are not in association with any previously discovered structure, but have been predicted for other reasons (e.g., they are in areas with good vegetation, they are very near to caribou paths that don't change very often, etc.). One of these locations might turn up a totally unexpected structure, seemingly isolated but which could in and of itself be a bridge to finding a bevy of future structures that are in association with *it*.

However, neither is paying the higher end of the accelerating cost always a *good* decision, especially if one's desire is simply to find "low hanging fruit" around previously-successful areas. In the end, the decision on what cost to pay is, as it must be, left up to the individual team using the system based on their own circumstances, priorities, and desires.

9.6 Evaluating the Low Initial Cost Hypothesis

We designed Archaeological Teams 1 and 2 under the assumption that these were smaller teams with more limited means. By picking mostly from the low end of the cost curves, Expedition Team 1 created a season plan containing 135 250m x 250m locations. The total area covered by this season plan is 8.44 sq km, well within the reach of a smaller team with more limited means for an expedition season. Also by picking mostly from the low end of the cost curves, Archaeological Team 2 created a season plan containing 76 250m x 250m locations. The total area covered by this season plan is 4.75 sq km, again well within the reach of a smaller team with more limited means for an expedition season. We consider this an adequate demonstration of the Low Initial Cost Hypothesis.

CHAPTER 10: CONCLUSION

We began this dissertation by introducing the Alpena-Amberely Land Bridge Project and summarizing previous work on the Project. We then discussed the paleogeology of the Alpena-Amberley Ridge Region during the relevant prehistoric period (11800 BP – 8400 BP), as well as the ways in which the regional environment changes during this 3,400 year period of time. We then described all the relevant types of potential occupational structures. We then detailed how the prehistoric environment, along with all relevant environmental parameters such as prehistoric water levels, terrain elevations, and vegetation levels, are modeled.

We then posited that the essential problem facing archaeological expeditions could be stated in terms of a payout vs. cost tradeoff, and we proposed that this could be stated specifically in terms of "occupational structures predicted" vs. "locations predicted". We then devised a rule-based mathematical formula for each of these quantities and demonstrated how they could be set against each other in the form of a biobjective optimization problem with the former quantity taking the role of "payout" and the latter quantity taking the role of "cost". We demonstrated how solving this biobjective optimization problem for each occupational structure type simultaneously creates a "payout vs. cost" Pareto Front for that structure type along with a corresponding ruleset for each individual Pareto point.

We then introduced Cultural Algorithms (CA's) along with the different knowledge source types that CA's use. We then gave a brief overview of Pareto-based multi-objective optimization and provided a description, algorithm, and pseudocode for CAPSO, which is the optimizer system that we would use to solve the biobjective optimization problems posited earlier. We then tested CAPSO's performance using several famous multiobjective benchmark problems.

After determining that the benchmark tests had been successful, we then entered the biobjective problems posited earlier with respect to each relevant Paleolithic occupational structure type into the CAPSO system. It successfully produced a Pareto Front plus all relevant metrics (learning curves, etc.) for each individual structure type. Then, for each individual point in each Pareto front, we produced a "frame" containing the point, its corresponding ruleset, and prediction map.

We then proposed the Accelerating Cost Hypothesis (ACH), which can be stated as *"If predicting a certain number of structures is at the cost of flagging a certain number of locations, then predicting a slightly greater number of structures will be at the cost of flagging a much greater number of locations."* We evaluated each Pareto Front from each of the four structure types (Hunting Blinds, Drive Lines, Caches, and Logistical Camps) using statistical methodology and found the ACH to be statistically valid for all four structure types.

We then proposed the Ruleset Size vs. Problem Complexity Hypothesis, which states that a comparatively smaller ruleset size in an expert system does not necessarily proscribe complex behavior in the Cultural Algorithm that is providing the optimization services. When we evaluated learning curves, dominance graphs, and parallelization behavior, we came to the conclusion that the Ruleset Size vs. Problem Complexity Hypothesis is true.

Then, to explore how our system might be used in practice, we created three hypothetical archaeological teams each with different hypothetical research goals, degrees of expertise, and resources available. We explored how each of these teams might use our aforementioned results in order to plan their respective expedition seasons. We posited how each individual team might respond to the accelerating costs considering its research goals and available resources. We then evaluated our Low Initial Cost Hypothesis, which states that the lower end of the accelerating cost curve is still affordable even for expeditionary teams of more limited means, in light of the two smaller hypothetical archaeological teams. We found that by choosing from the lower end of the cost curve, Team 1 was able to assemble a plan covering 7.25 sq km, and Team 2 was able to assemble a season plan covering 4.25 sq km. We hold both of these quantities to be well within the reach of smaller teams of more limited means.

Finally, we discussed the underlying reason behind the Accelerating Cost Hypothesis (i.e., incomplete information) and also discussed some of the ACH's implications. We revisited the expedition planning decisions made by our hypothetical archaeological teams and noted that paying the higher end of the accelerating cost curve was not always a bad choice. Indeed, certain interesting predictions only become available when the higher end is paid. However, paying the higher end of the cost curve of course still remains a bad or perhaps even impossible choice for teams with more limited budgets.

BIBLIOGRAPHY

[Ali, 2011] M. Z. Ali, A. Salhieh, R. T. A. Snanieh and R. G. Reynolds, "Boosting Cultural Algorithms with an Incongruous Layered Social Fabric Influence Function," in *2011 IEEE Congress on Evolutionary Computation*, June 5-8, New Orleans, LA, pp. 1225-1232, 2011.

[Barth, 2010] Barth, Amy. “#95: Hidden Caribou-Hunting Civilization Found Under Lake Huron,” *Discover*, Jan-Feb 2010 Issue, 2010.

[Best, 2009] C. Best, *Multi-Objective Cultural Algorithms*, Wayne State University M.S. Thesis, Detroit, MI, 2009.

[Best, 2010] C. Best, C. X, R. Reynolds and D. Liu, "Multi-Objective Cultural Algorithms," in *IEEE Congress on Evolutionary Computation*, pp. 3330-3338, 2010.

[Binford, 1978a] L. R. Binford, "Dimensional Analysis of Behavior and Site Structure: Learning from an Eskimo Hunting Stand," *American Antiquity,* Vol. 43, No. 3, pp. 330-361, 1978.

[Binford, 1978b] L. Binford, *Nunamiut Ethnoarchaeology*, Academic Press, New York, 1978.

[Binford, 1980] L. Binford, "Willow Smoke and Dogs' Tails: Hunter-Gatherer Settlement Systems and Archaeological Site Formation," American Antiquity, Vol. 45, No. 1, pp. 4-20, 1980.

[Binford, 1982] L. Binford, "The Archaeology of Place," Journal of Anthropological Archaeology, Vol. 1, No. 1, pp. 5-31, 1982.

[Binford, 1991] L. Binford, "A Corporate Caribou Hunt," Expedition, Vol. 33, No. 1, pp. 33-43, 1991.

[Brink, 2005] J. W. Brink, "Inukshuk: Caribou Drive Lanes on Southern Victoria Island," Arctic Anthropology, Vol. 42, No. 1, pp. 1-28, 2005.

[Deb, 2001] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*, New York: John Wiley and Sons, Inc., 2001.

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

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

[Eberhart, 1995] R. Eberhart and J. Kennedy, “A New Optimizer Using Particle Swarm Theory” in *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, Nagoya, Japan, Oct 4-6 1995.

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

[Gleason, 1915] E. Gleason, S. Guernsey, and C. Willoughby, “Model of Alaska Eskimo House Group”. Peabody Museum of Archaeology and Ethnology: Cambridge, MA, 1915.

[Hall, 1865] C. Hall, *Arctic Researches and Life Among the Esquimaux*. Harper & Brothers Publishers: New York, 1865.

[Herdendorf, 1990] C. Herdendorf, “Great Lakes Estuaries” in *Estuaries*, Vol. 13, No. 4, pp. 493-503, 1990.

[Hoobler, 2004] N. Hoobler, G. Humphreys and M. Agrawala, "Visualizing Competitive Behaviors in Multi-User Virtual Environments," in *IEEE Visualization*, Austin, Texas, pp. 163-170, 2004.

[Jayyousi, 2012] T. Jayyousi, *Bringing to Life an Ancient Urban Center at Monte Alban, Mexico: Exploiting the Synergy Between the Micro, Meso, and Macro Levels in a Complex System*, Wayne State University PhD Dissertation, Detroit, MI, 2012.

[Jin, 2011] J. Jin, *Path Planning in Reality Games Using Cultural Algorithm: The Land Bridge Example*, Wayne State University M.S. Thesis, Detroit, MI, 2011.

[Kennedy, 1995] J. Kennedy and R. Eberhart, “Particle Swarm Optimization,” *Proceedings of ICNN’95 – International Conference on Neural Networks*, Perth, WA, Australia, 1995, Vol. 4, pp. 1942-1948.

[Kita, 1996] H. Kita, Y. Yabumoto, N. Mori and Y. Nishikawa, "Multi-objective Optimization by Means of the Thermodynamical Genetic Algorithm," in *International Conference on Parallel Problem Solving from Nature*, Berlin, Germany, Sept 22-26, 1996.

[Lemke, 2016] Lemke, Ashley. Anthropological Archaeology Underwater: Hunting Architecture and Foraging Lifeways beneath the Great Lakes. University of Michigan Dissertation, Ann Arbor, MI, 2016.)

[Lewis, 1989] 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.

[Lewis, 1994] C. F. M. Lewis, T. C. Moore, D. K. Rea, D. L. Dettman, A. M. Smith, and L. A. Mayer, “Lakes of the Huron Basin: Their Record of Runoff from the Laurentide Ice Sheet,” *Quatrenary Science Reviews*, Vol. 13, pp. 891-922, 1994.

[Lewis, 2005] 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.

[Lewis, 2007] C. F. M. Lewis, C. W. Heil, J. B. Hubeny, J. W. King, T. C. Moore and D. K. Rea, "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.

[Lewis, 2012] C. F. M. Lewis and T. W. Anderson, "The Sedimentary and Palynological Records of Serpent River Bog, and Revised Early Holocene Lake-Level Changes in the Lake Huron and Georgian Bay Region," *Journal of Paleolimnology*, Vol. 47, No. 3, pp. 391-410, 2012.

[Lewis, 2016] C. F. M. Lewis, “Understanding the Closed-Basin Phases (Lowstands) of the Laurentian Great Lakes and their Significance,” *Geoscience Canada*, Vol. 43, pp. 179-197, 2016.

[LAC, 1930] Library and Archives Canada, PA-101294, “Eskimo Meat Cache on Kazan River, Keewatin District, N. W. T.” Photographer: A. E. Porslid, 1930.

[Li, 2018] J. Li and Y. Tan. “The Bare Bones Fireworks Algorithm: A Minimalist Global Optimizer”, *Applied Soft Computing*, Vol. 62, pp. 454-462, 2018.

[Miles, 2006] C. Miles and S. Louis, "Towards the Co-Evolution of Influence Map Tree Based Strategy Game Players," in *IEEE Symposium on Computational Intelligence and Games*, Reno, Nevada, pp. 75-82, 2006.

[Millington, 2009] I. Millington and J. Funge, Artificial Intelligence for Games, CRC Press: , Boca Raton, Florida, 2009.

[Milton, 1964] 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.

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

[O'Shea, 2009] J. M. O'Shea and G. A. Meadows, "Evidence for Early Hunters Beneath the Great Lakes," *National Academy of Sciences Proceedings,* Vol. 106, No. 25, pp. 10120-10123, 2009.

[O'Shea, 2013] J. O'Shea, A. Lemke and R. G. Reynolds, "Nobody Knows the Way of the Caribou": Rangefier Hunting at 45° North Latitude," *Quatrenary International,* Vol. 297, pp. 36-44, 2013.

[Palazzolo, 2019] T. Palazzolo. *Exploring Virtual Worlds with Cultural Algorithms: Ancient Alpena-Amberley Land Bridge*. Wayne State University PhD Dissertation. Detroit, MI, 2019.

[Peng, 2005] B. Peng, *Knowledge and Population Swarms in Cultural Algorithms for Dynamic Environments*, Wayne State University PhD Dissertation, Detroit, MI, 2005.

[Raquel, 2005] C. Raquel and P. Naval, Jr., "An Effective Use of Crowding Distance in Multiobjective Particle Swarm Optimization," *Proceedings of the 7th Annual Conference on Genetic and Evolutionary Computation*, pp. 257-264, June 25-29, 2005.

[Reynolds, 1978] R. G. Reynolds, "On Modeling the Evolution of Hunter-Gatherer Decision Making Systems," *Geographical Analysis*, Vol. 10, No. 1, pp. 31-46, 1978.

[Reynolds, 1979] R. G. Reynolds, *An Adaptive Computer Model of the Evolution of Agriculture for Hunter-Gatherers in the Valley of Oaxaca, Mexico,* Ann Arbor: PhD Dissertation, University of Michigan, 1979.

[Reynolds, 1994] R. G. Reynolds, "An Introduction to Cultural Algorithms," in *Proceedings of the 3rd Annual Conference on Evolutionary Programming*, pp. 131-139, 1994.

[Reynolds, 1997] R. G. Reynolds and C. Chung, "Knowledge-Based Self-Adaptation in Evolutionary Programming Using Cultural Algorithms," in *IEEE International Conference on Evolutionary Computation*, Indianapolis, IN, pp. 71-76, 1997.

[Reynolds, 2006] R. G. Reynolds, M. Ali and R. S. Alomari, "Optimization Problem Solving Using Predator/Prey Games and Cultural Algorithms," in *IEEE Synopsium on Computational Intelligence and Games*, Reno, NV, pp. 119-125, 2006.

[Reynolds, 2013] R. G. Reynolds and L. Kinnard-Heather, "Optimization Problem Solving with Auctions in Cultural Algorithms", *Memetic Computing*, Vol. 5, No. 2, June 2013.

[Reynolds, 2018] R. G. Reynolds, *Culture on the Edge of Chaos: Cultural Algorithms and the Foundations of Social Intelligence*. Springer, Cham, Switzerland, 2018.

[Smith, 1991] E. A. Smith, *Inujjuamiut Foraging Strategies*, Aldine De Gruyter: New York, 1991.

[Schaffer, 1985] J. D. Schaffer, "Multi Objective Optimization with Vector Evaluated Genetic Algorithms," in *Proceedings of an International Conference on Genetic Algorithms and their Applications*, Pittsburgh, PA, July 24-26, 1985.

[Sonnenburg, 2015] E. Sonnenburg, A. Lemke, and J. O'Shea, eds. *Caribou Hunting in the Upper Great Lakes: Archaeological, Ethnographic, and Paleoenvironmental Perspectives.* Memoirs of the Museum of Anthropology, University of Michigan, No. 57. Ann Arbor, MI, 2015.

[Srinivas, 1994] N. Srinivas and K. Deb, "Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms," *Journal of Evolutionary Computation*, Vol. 2, No. 3, pp. 221-248, 1994.

[Stanley, 2013] S. D. Stanley, *Analyzing Environmental Change And Prehistoric Hunter Behavior Through A 3D Time-Lapsed Model With Level Auto-Generation And Cultural Algorithms*, Wayne State University M.S. Thesis, Detroit, MI, 2013.

[Stanley, 2014] Stanley, S. D., Salaymeh, A., Palazzolo, T., and Warnke, D., "Analyzing Prehistoric Hunter Behavior with Cultural Algorithms", *Proceedings of the 2014 IEEE Congress on Evolutionary Computation (CEC)*, Beijing, China, pp. 2196-2205, July 6-11, 2014.

[Stopp, 2002] M. P. Stopp, "Reconsidering Inuit Presence in Southern Labrador," in *Populations et Migrations*. Vol. 26, No. 2, pp. 71-106, 2002.

[Sweester, 2005] P. Sweetser and J. Wiles, "Combining Influence Maps and Cellular Automata for Reactive Game Agents," in *Intelligent Data Engineering and Automated Learning (IDEAL), 6th International Conference*, Brisbane, Australia, pp. 524-531, 2005.

[Tanaka, 1995] M. Tanaka, H. Watanabe, Y. Furukawa and T. Tanino, "GA-based Decision Support System for Multicriteria Optimization," in *1995 IEEE International Conference on Systems, Man and Cybernetics. Intelligent Systems for the 21st Century*, Vancouver, Canada, 22-25 Oct, 1995.

[UWLSC] University of Washington Libraries, Special Collections. "Native Fish Drying Racks, Libby, McNeill & Libby Cannery, Ekuk". Guy V. Graham Libby McNeill & Libby Photograph Collection. PH Coll 1033.36. Order Number: AWC6362.

[Vitale, 2009] K. Vitale, *Learning Group Behavior in Games Using Cultural Algorithms and the Land Bridge Simulation Example*, Wayne State University M.S. Thesis, Detroit, MI, 2009.

[Wirth, 2008] N. Wirth and M. Gallagher, "An Influence Map Model for Playing Ms. Pac-Man," in *2008 IEEE Symposium on Computational Intelligence in Games*, Perth, WA, pp. 228-233, 2008.

[Yao, 2010] J. Yao, C. Lin, X. Xie, A. Wang, C. Hung. "Path Planning for Virtual Human Motion Using Improve A* Algorithm," in *2010 Seventh International Conference on Information Technology: New Generations.* Las Vegas, NV, pp. 1154-1158, 12-14 April 2010.

[Zhao, 2007] Y. Zhao, S. Xiong, and M. Li. "Constrained Single- and Multiple-Objective Optimization with Differential Evolution," in Third International Conference on Natural Computation, pp. 451-455, 2007.

[Zobrist, 1969] A. L. Zobrist, "A Model of Visual Organization for the Game of Go," in *AFIPS Conference Proceedings*, pp. 103-112, 1969.

APPENDIX

This appendix contains full-scale frames for all of the images found within Figure 71.

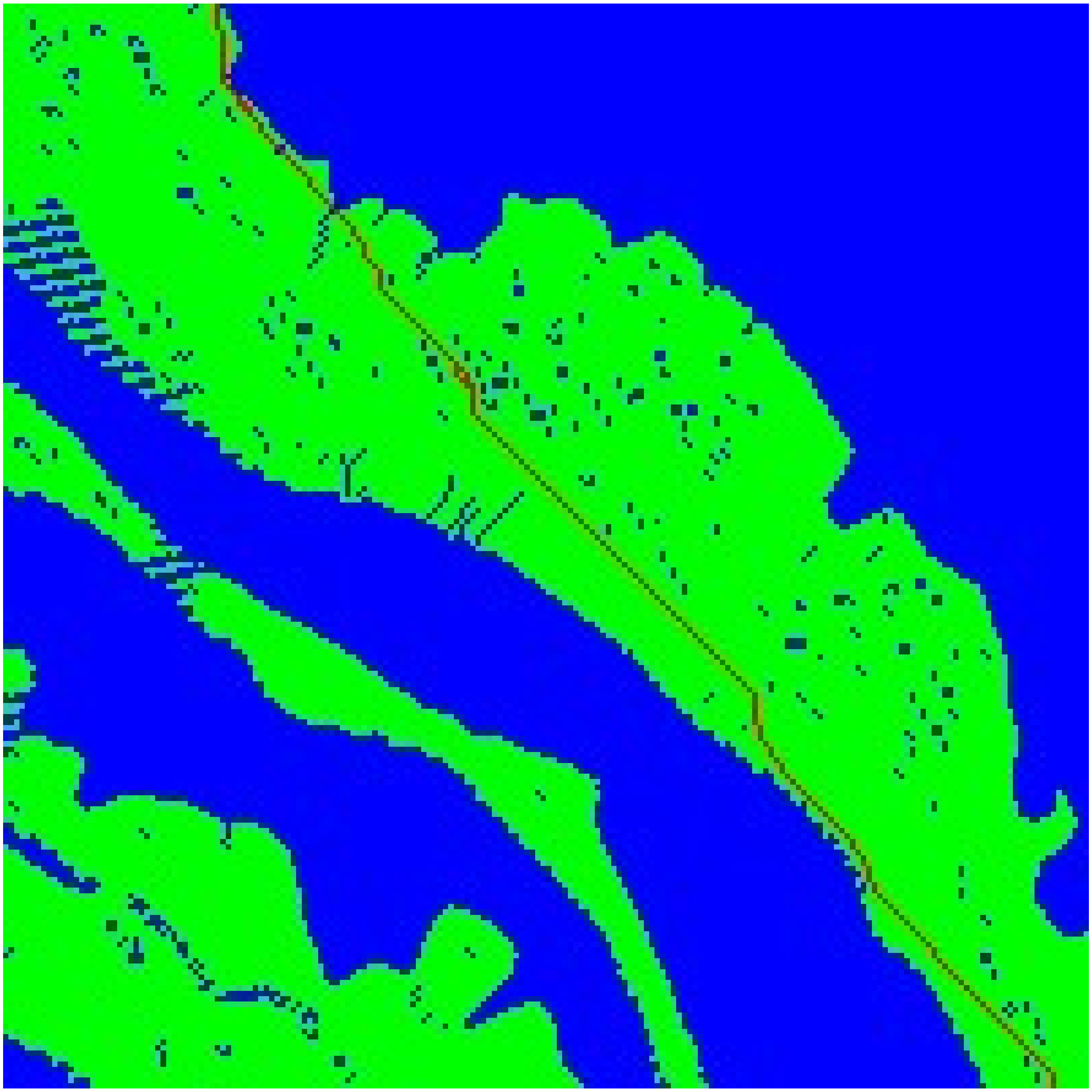


Figure 158: 11800BP Spring

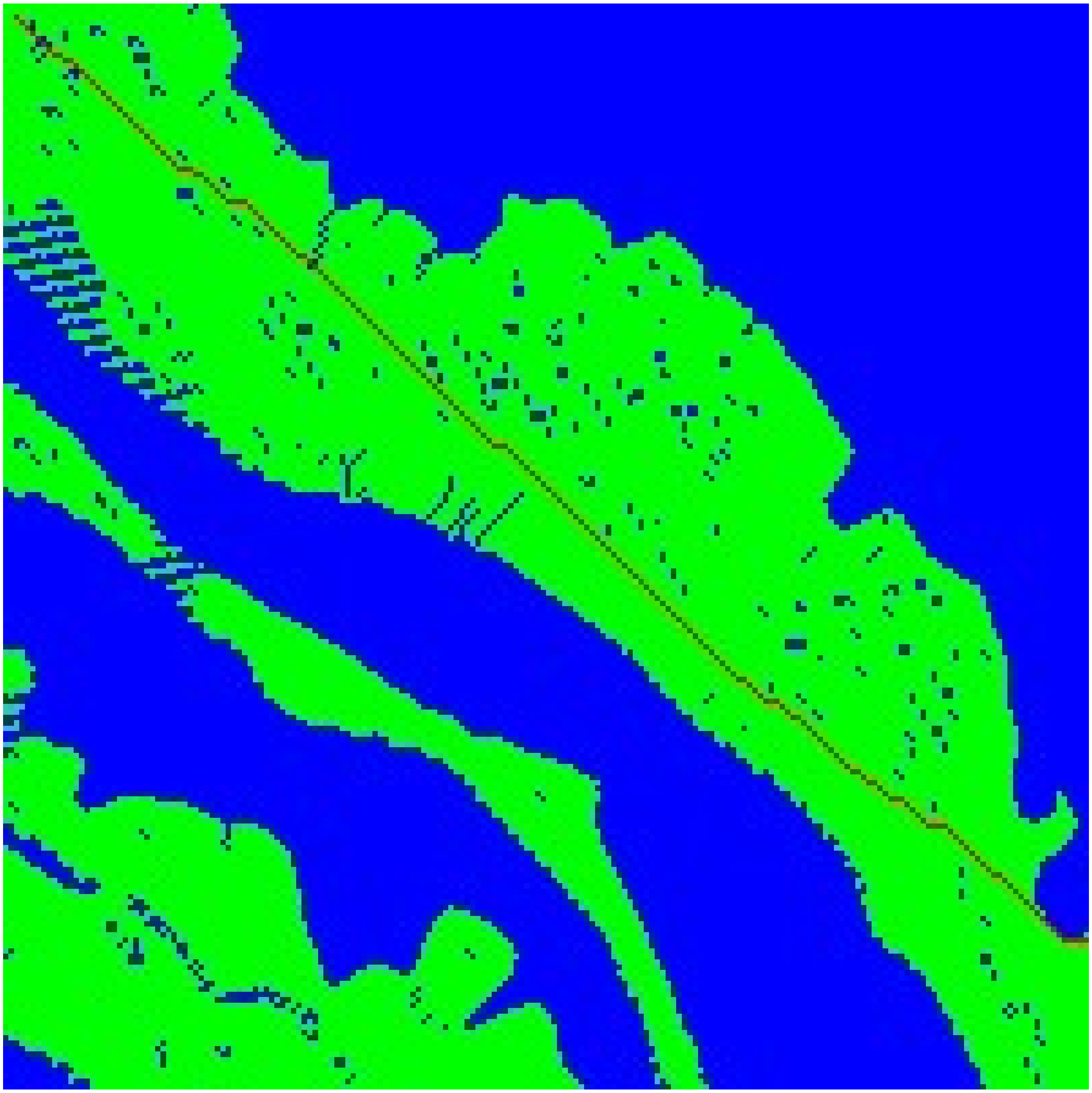


Figure 159: 11800BP Fall

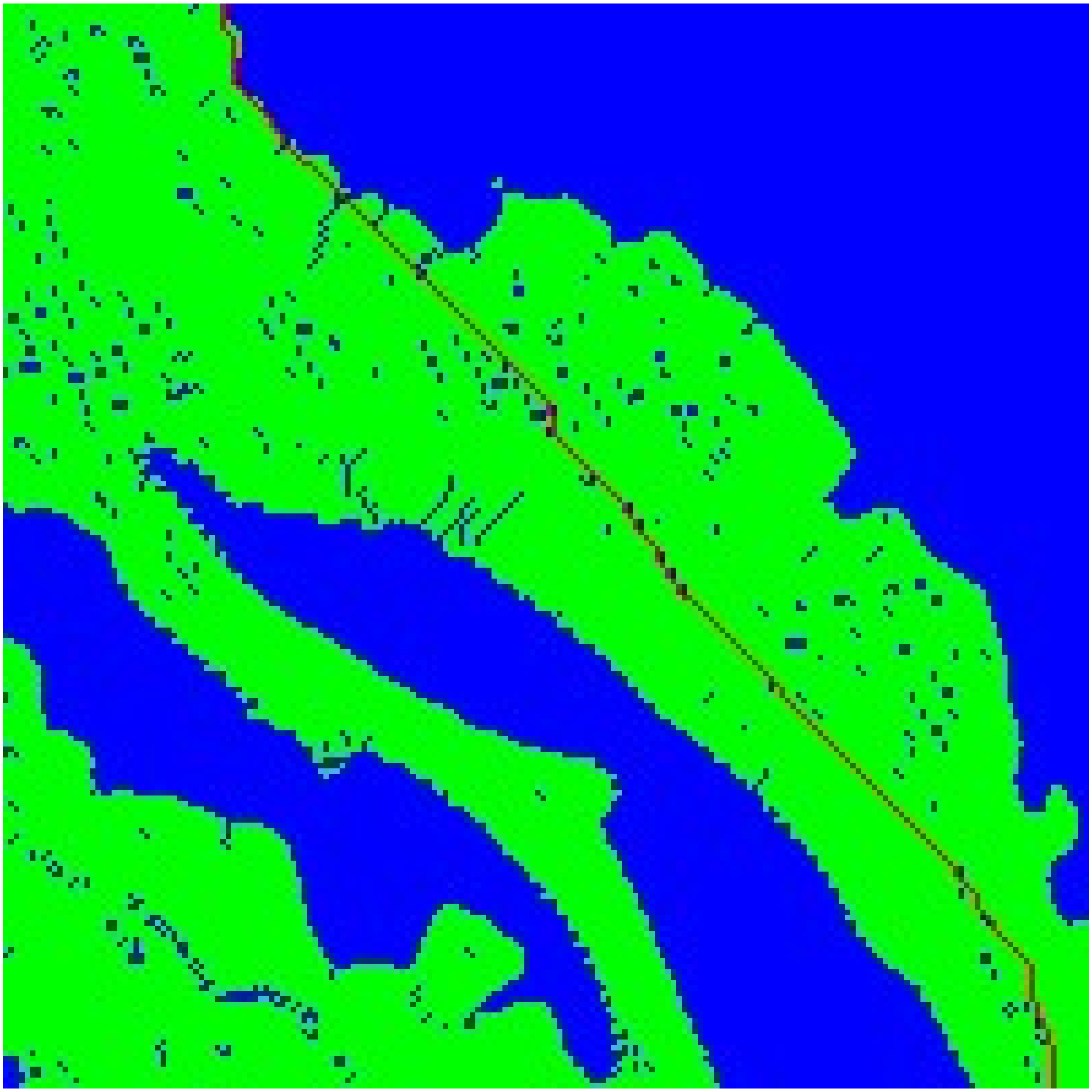


Figure 160: 11600 Spring

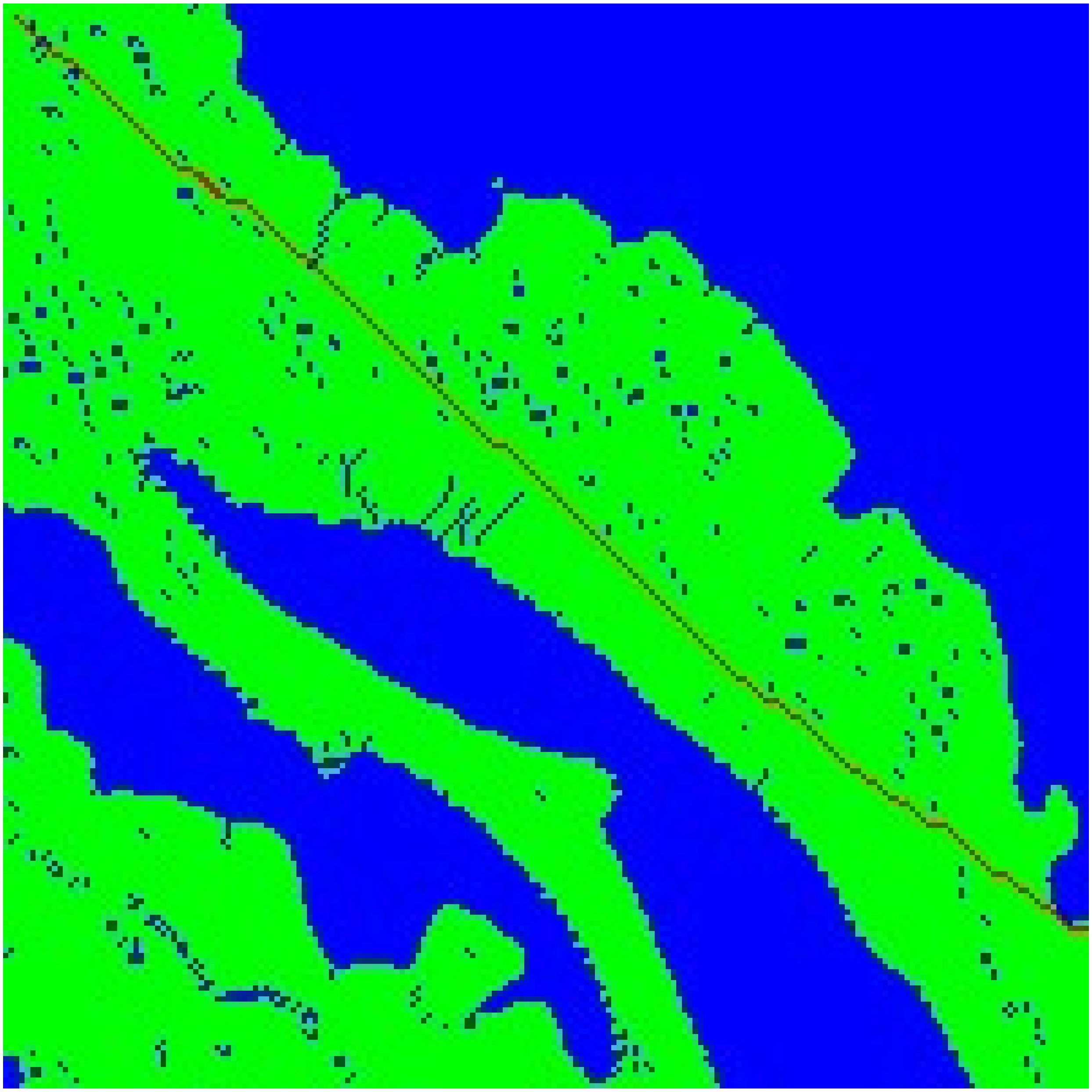


Figure 161: 11600 Fall

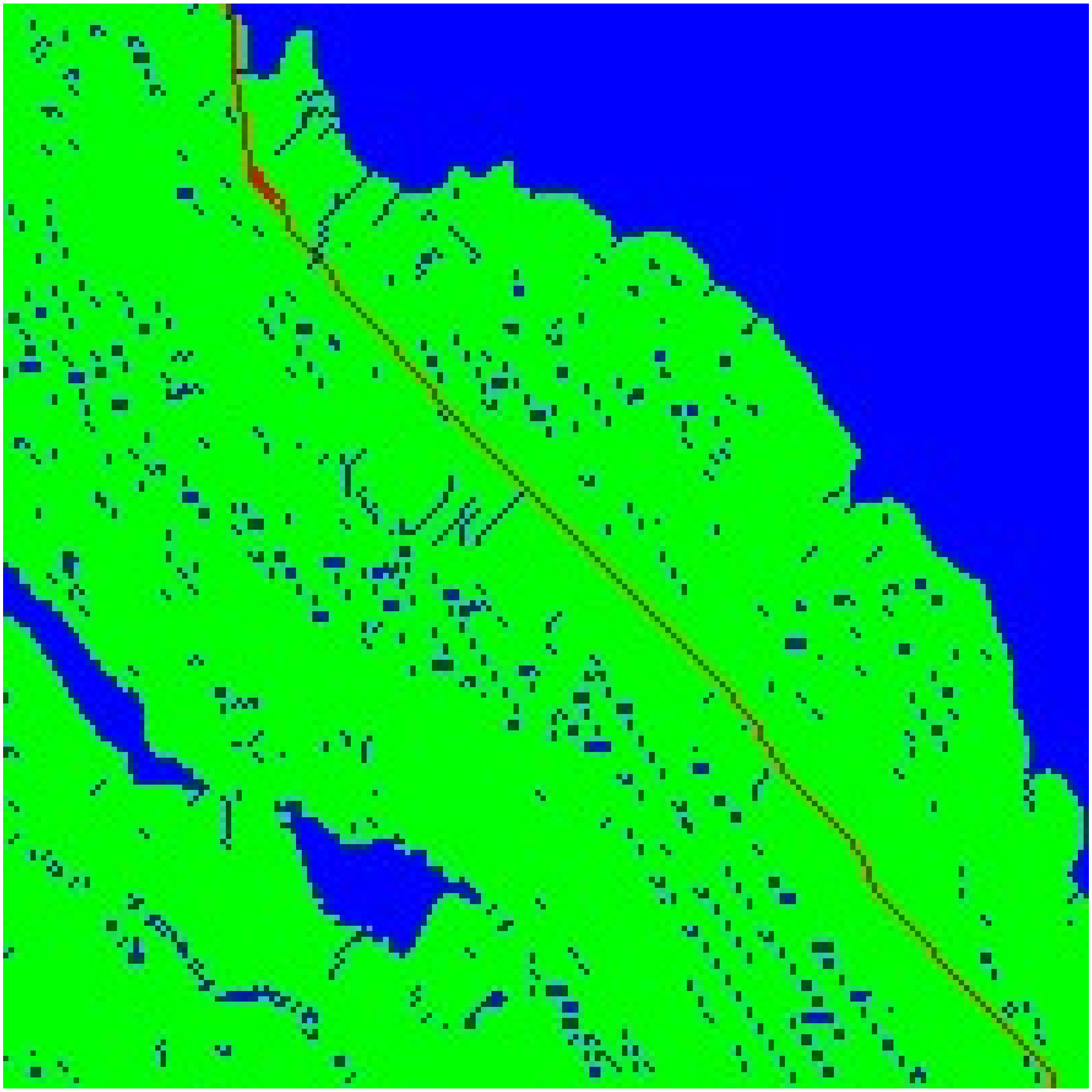


Figure 162: 11400 Spring

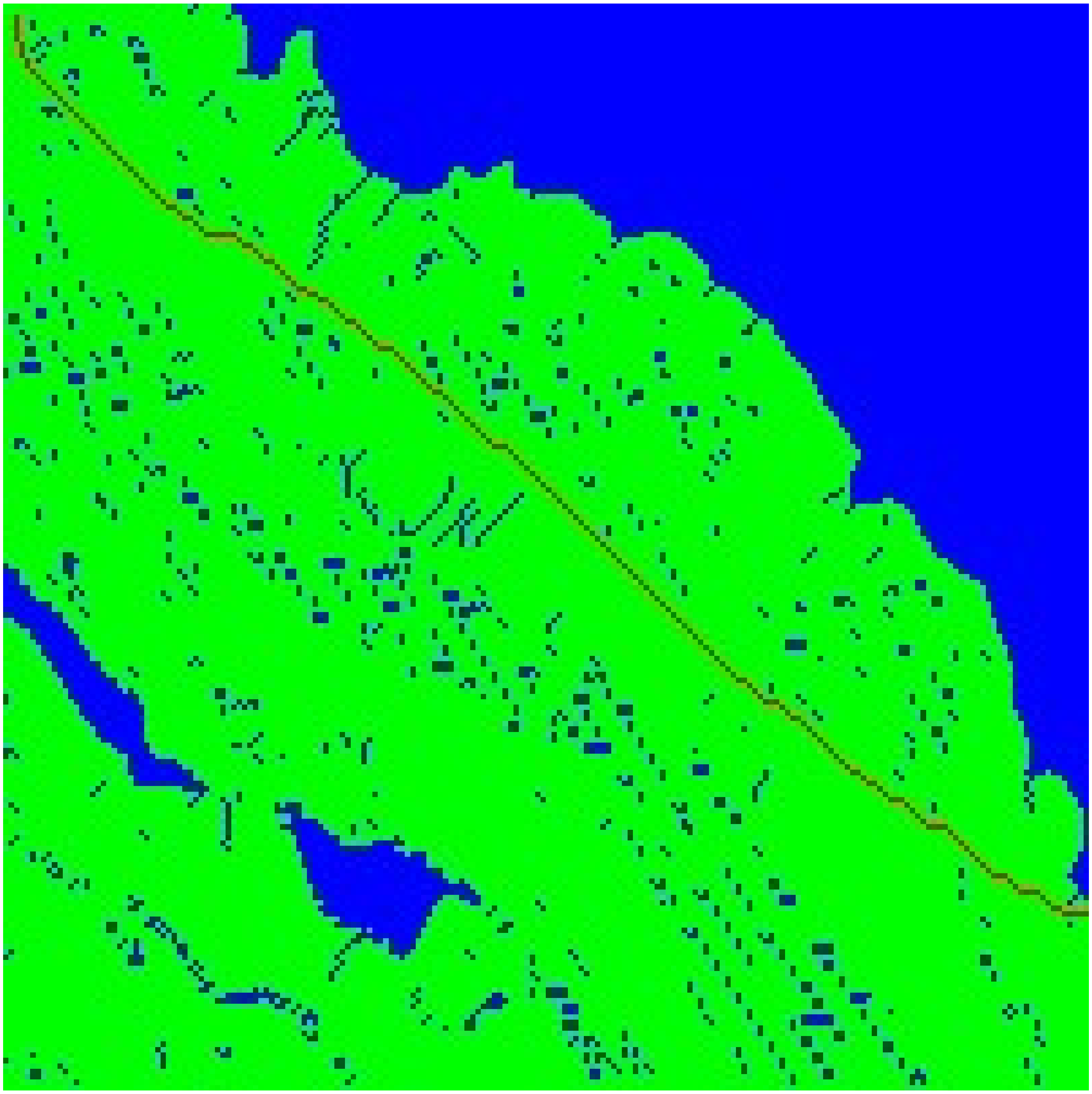


Figure 163: 11400 BP Fall

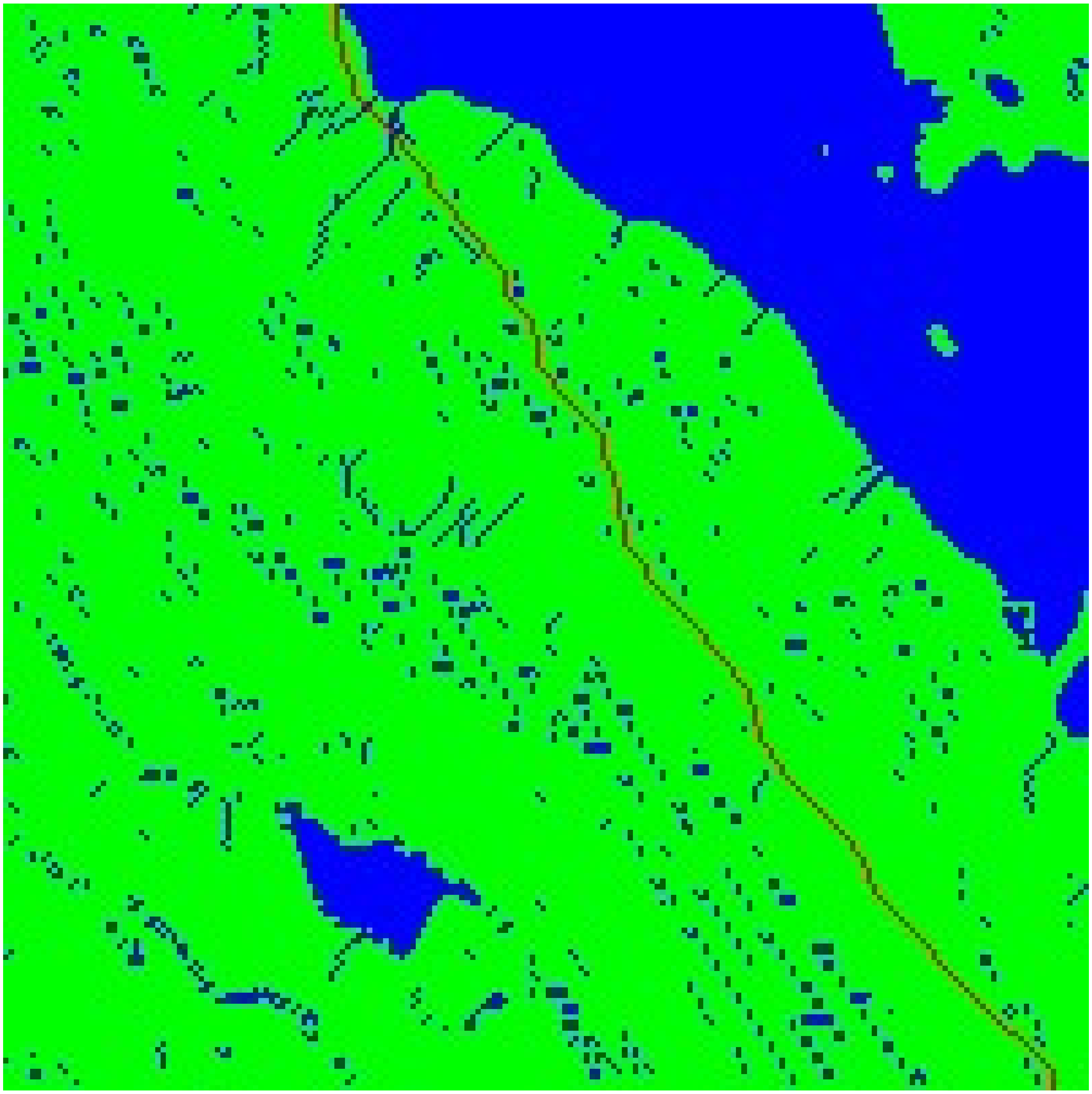


Figure 164: 11200BP Spring

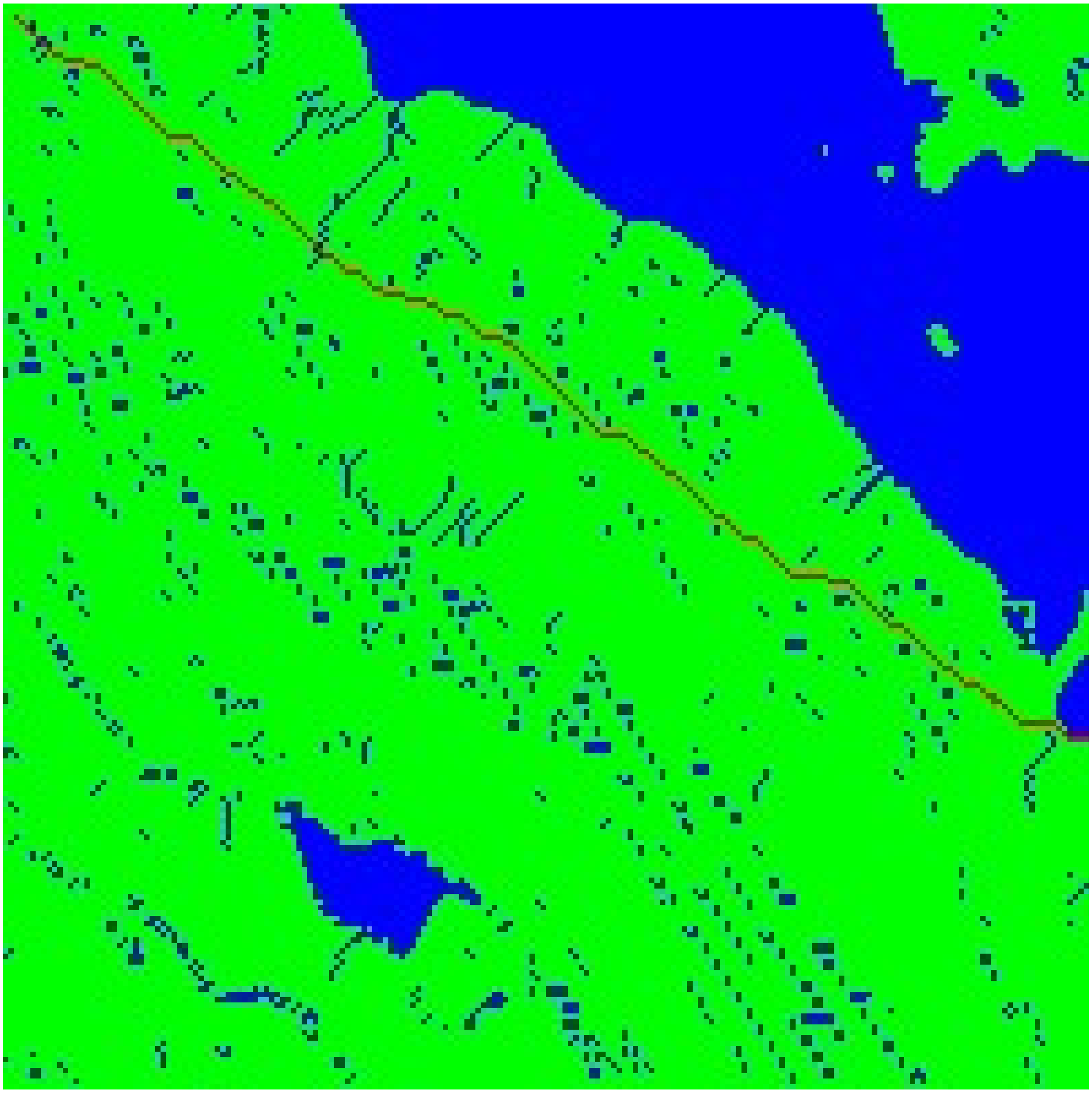


Figure 165: 11200BP Fall

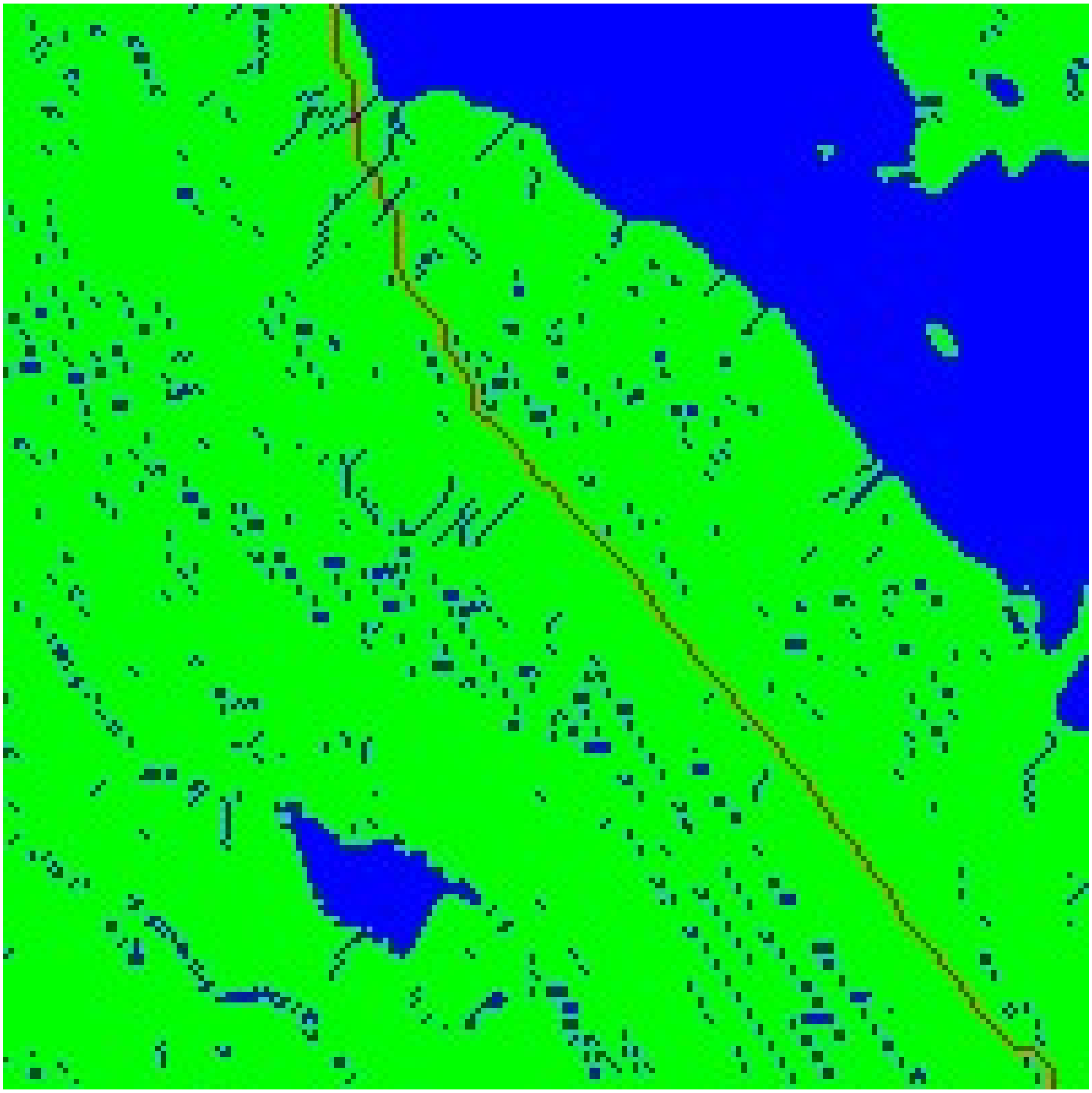


Figure 166: 11000 BP Spring

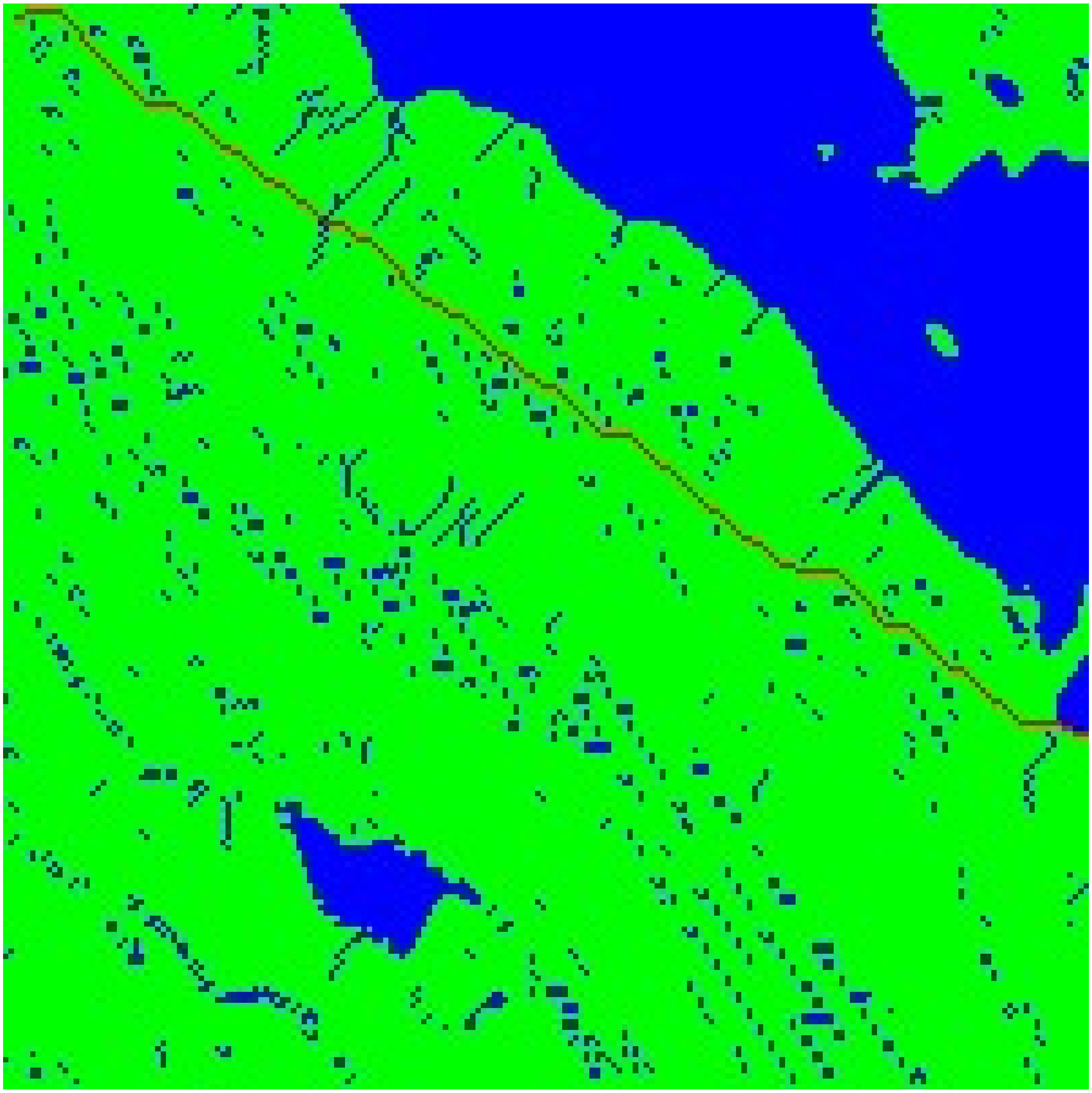


Figure 167: 11000BP Fall

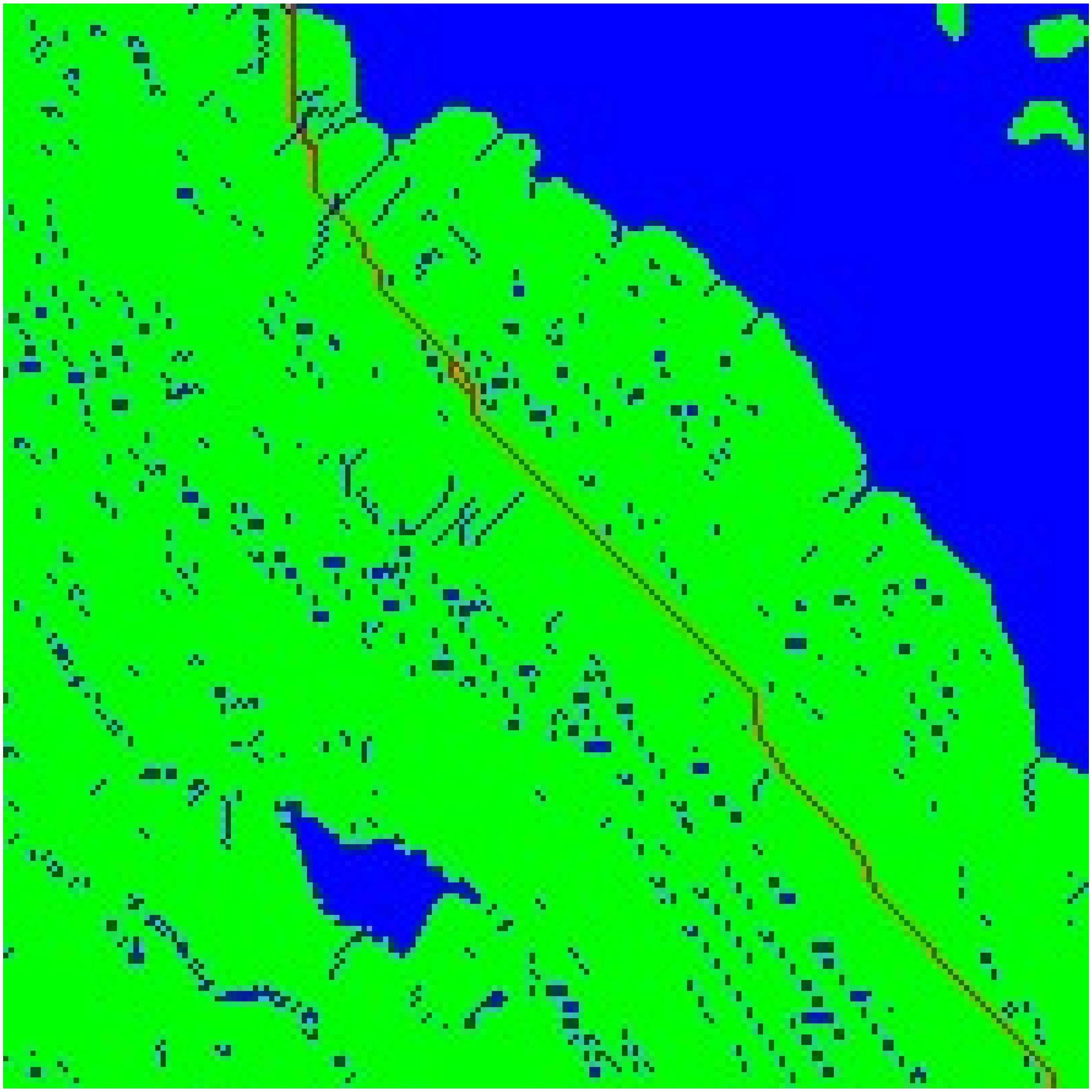


Figure 168: 10800BP Spring

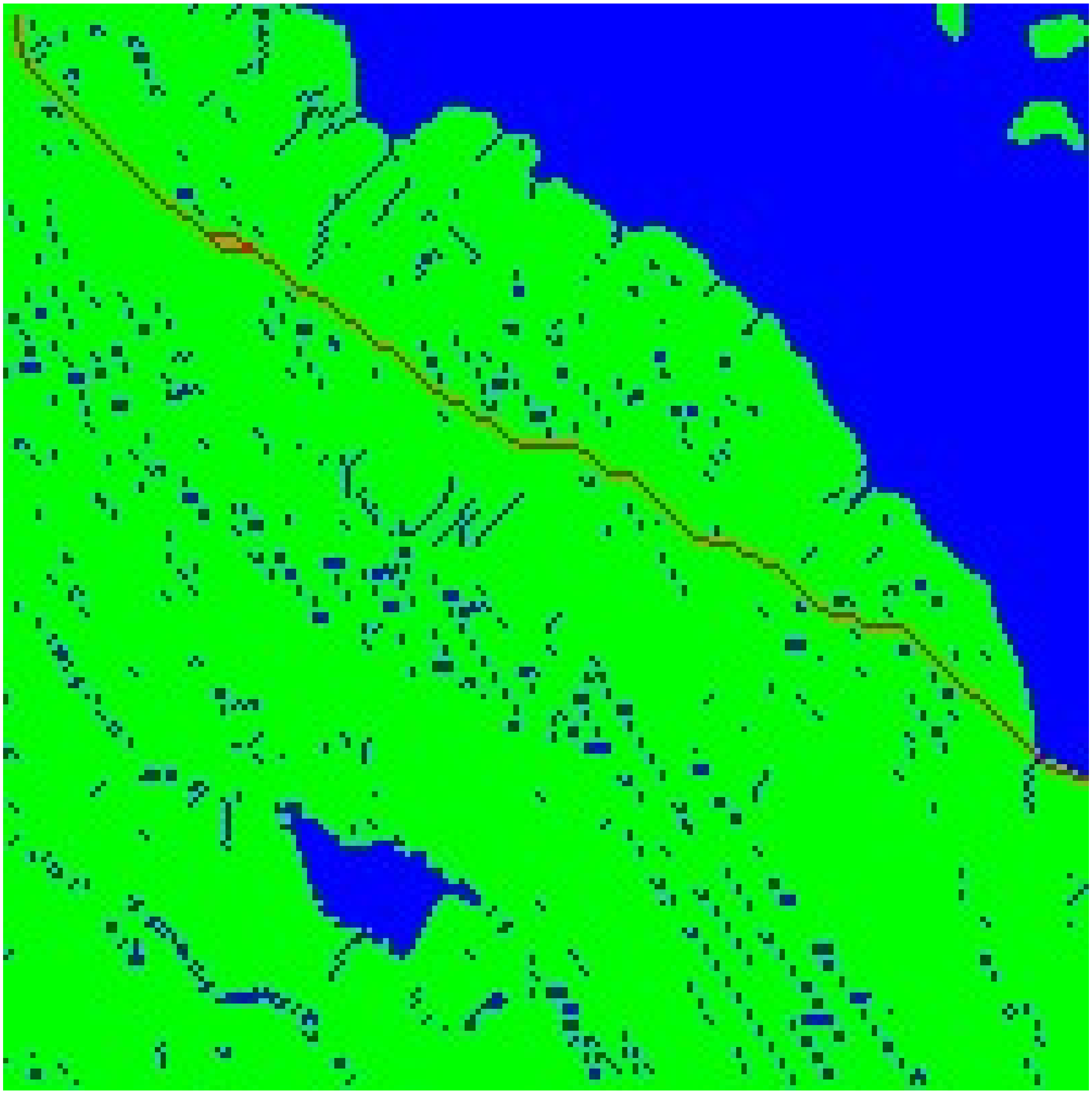


Figure 169: 10800BP Fall

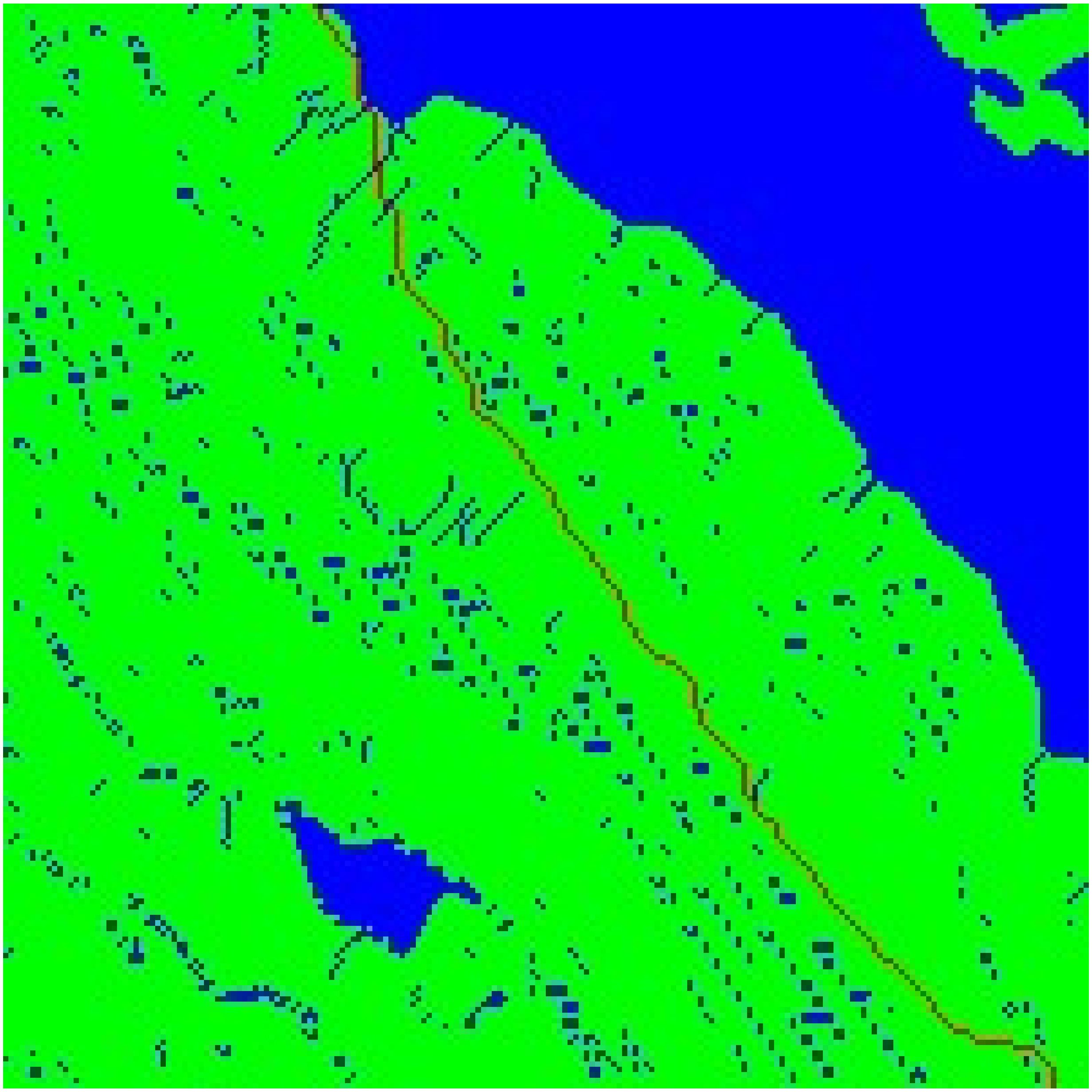


Figure 170: 10600BP Spring

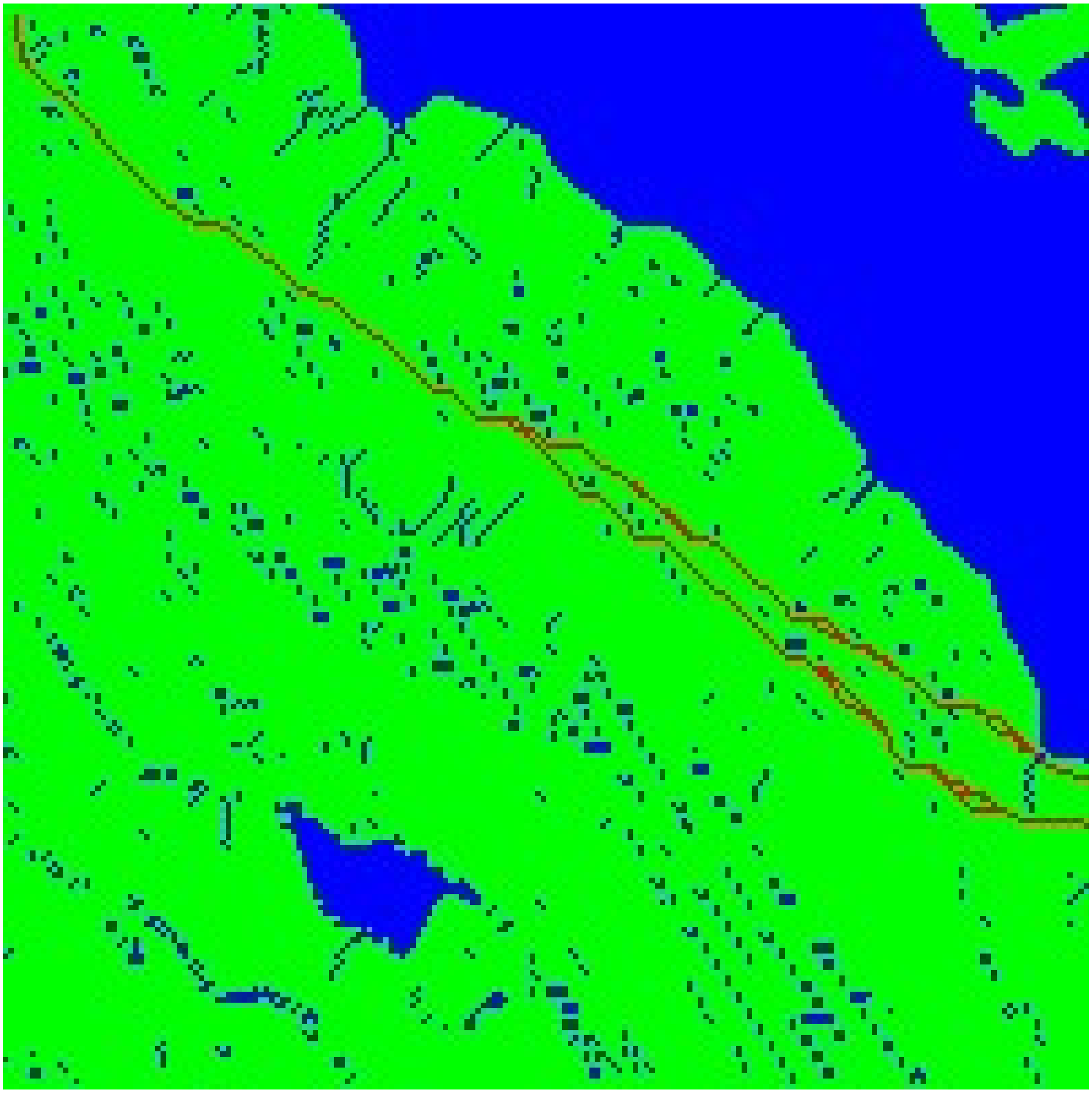


Figure 171: 10600BP Fall

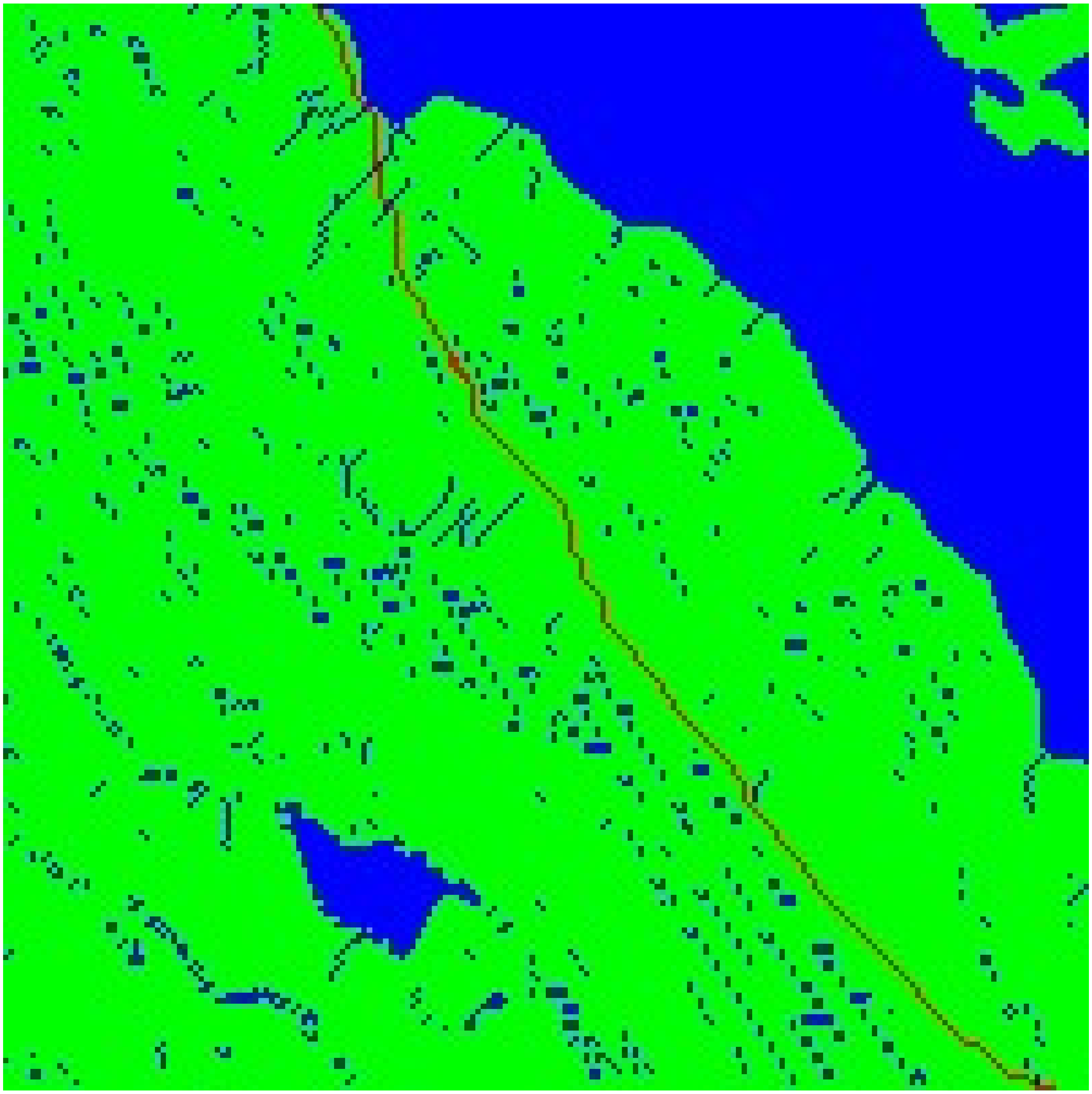


Figure 172: 10400 BP Spring

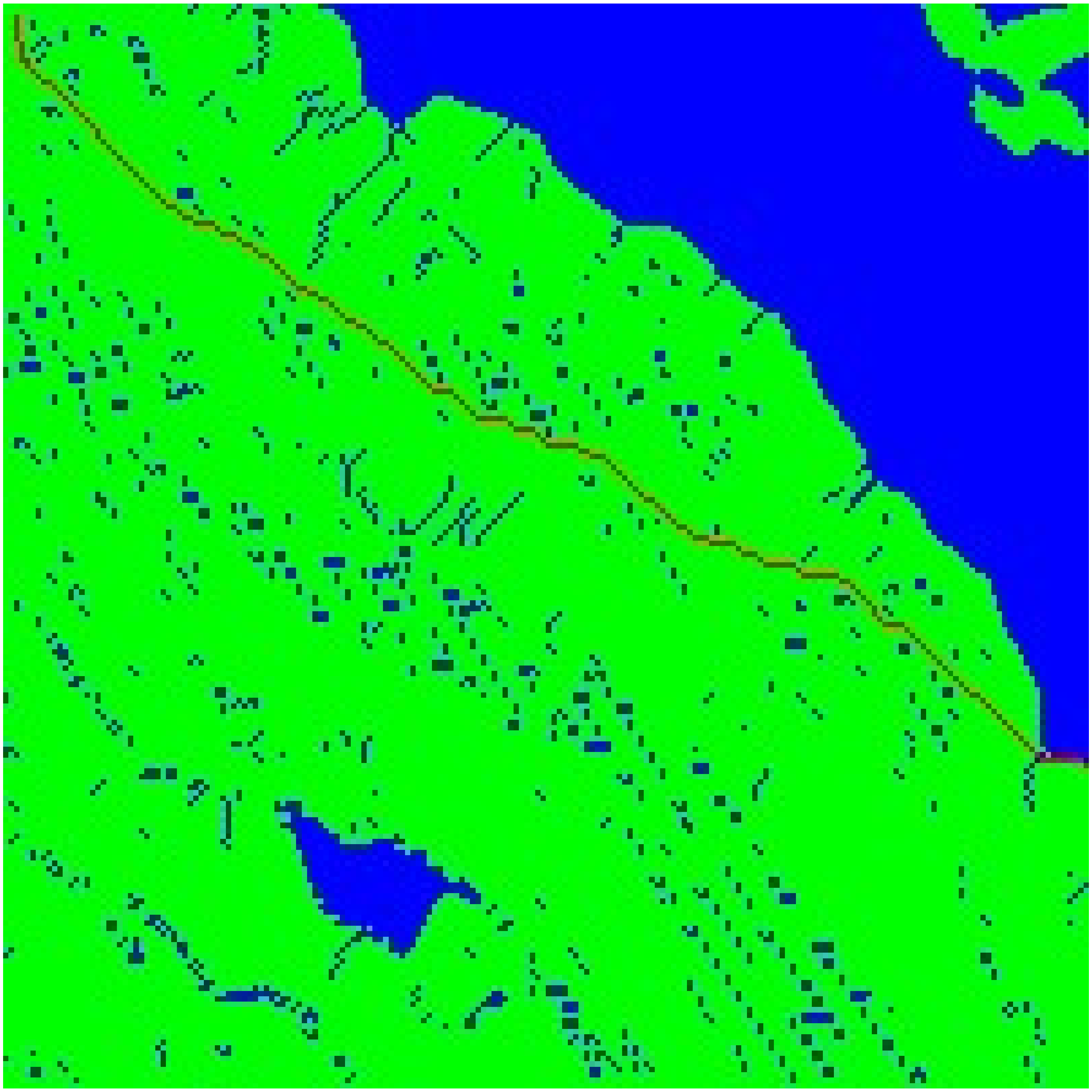


Figure 173: 10400BP Fall

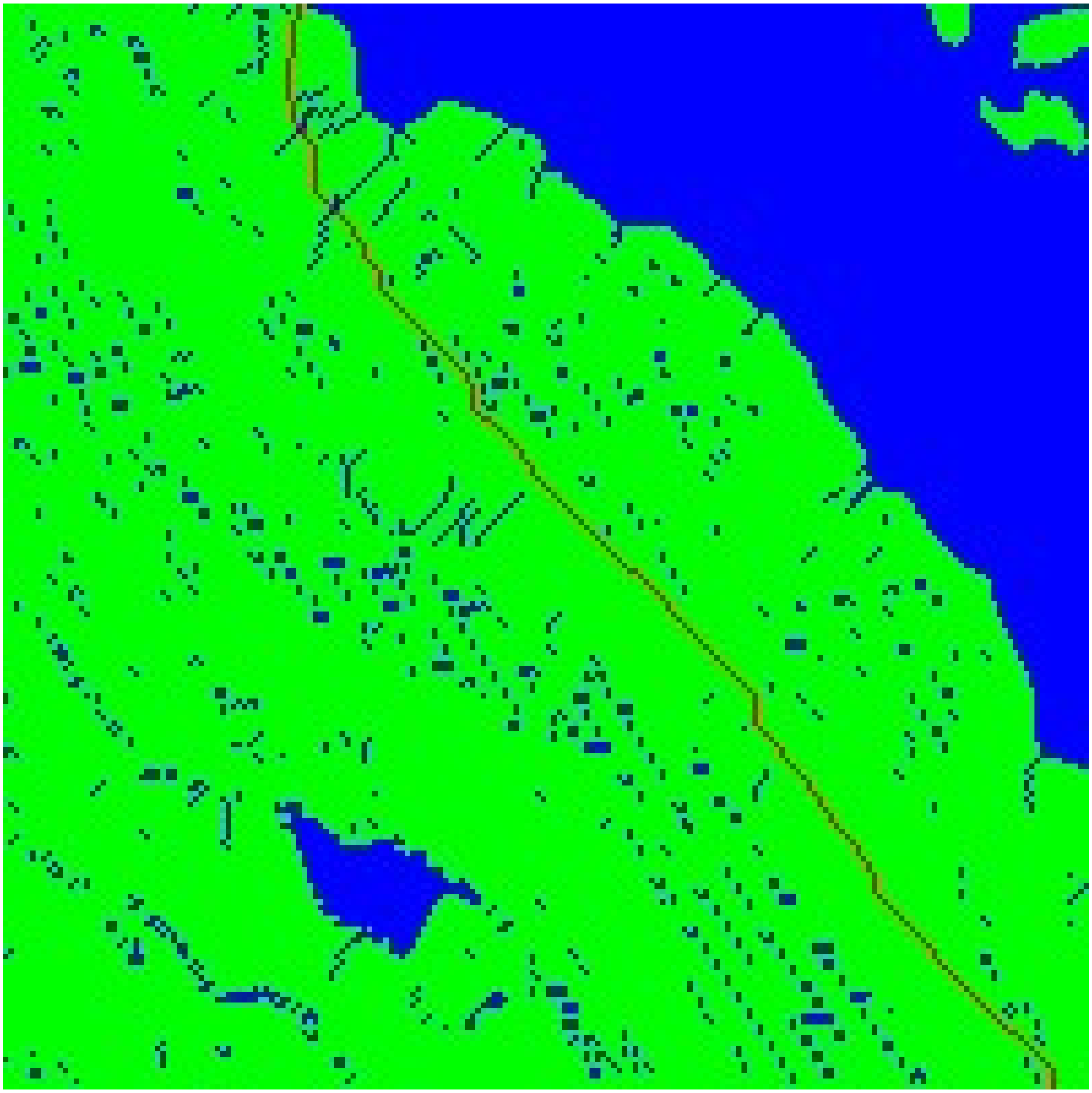


Figure 174: 10200 Spring

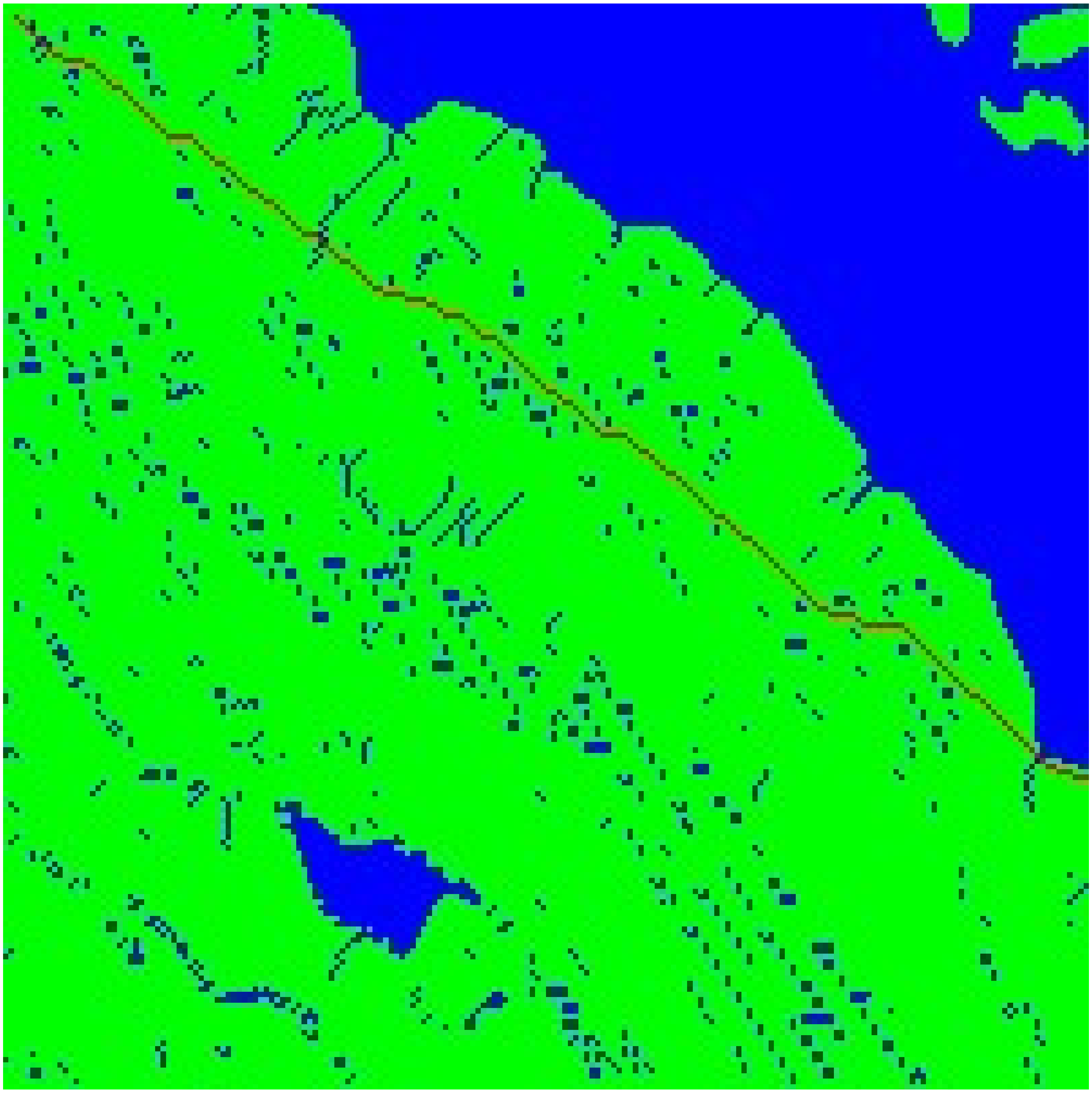


Figure 175: 10200BP Fall

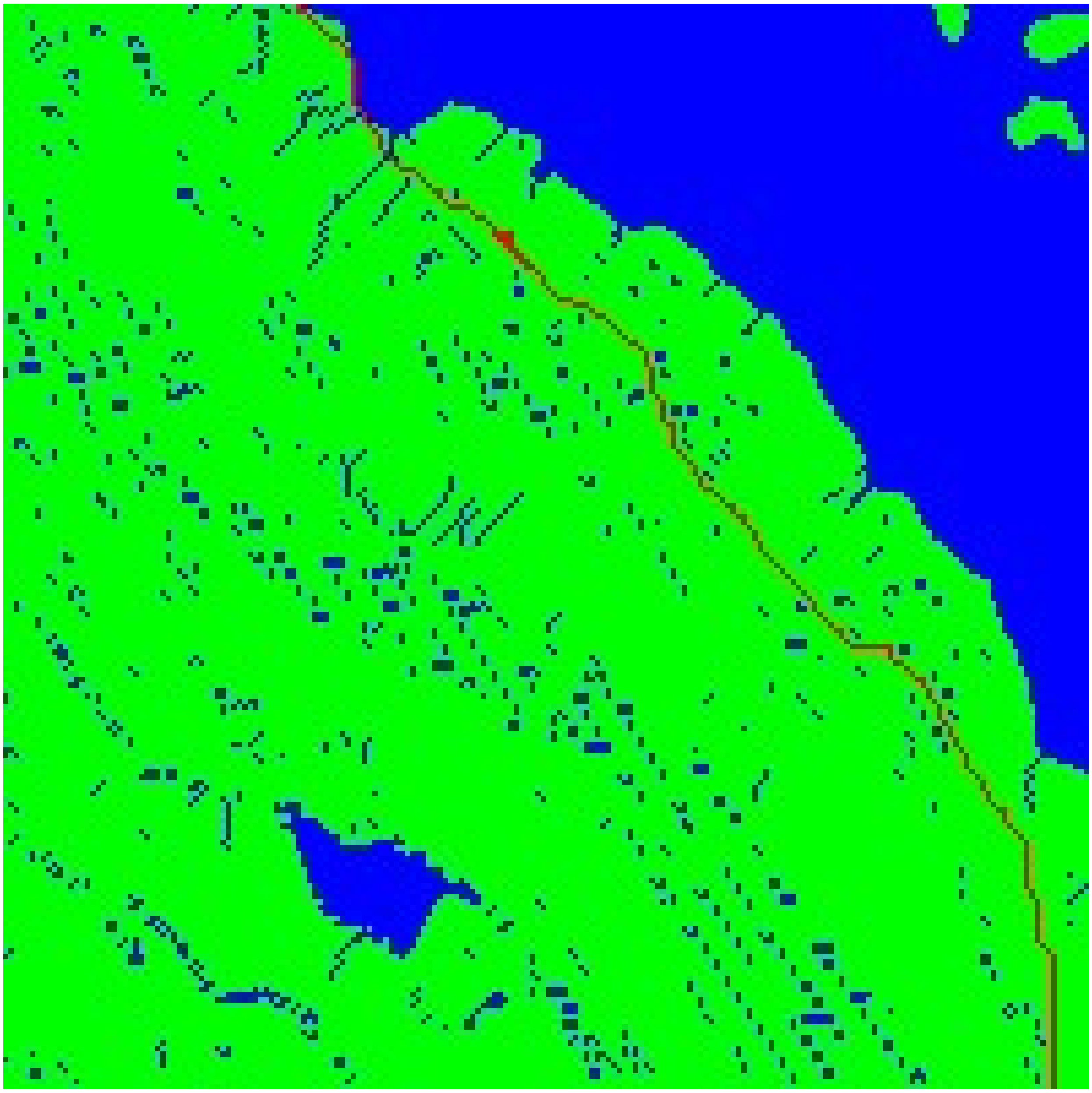


Figure 176: 10000BP Spring

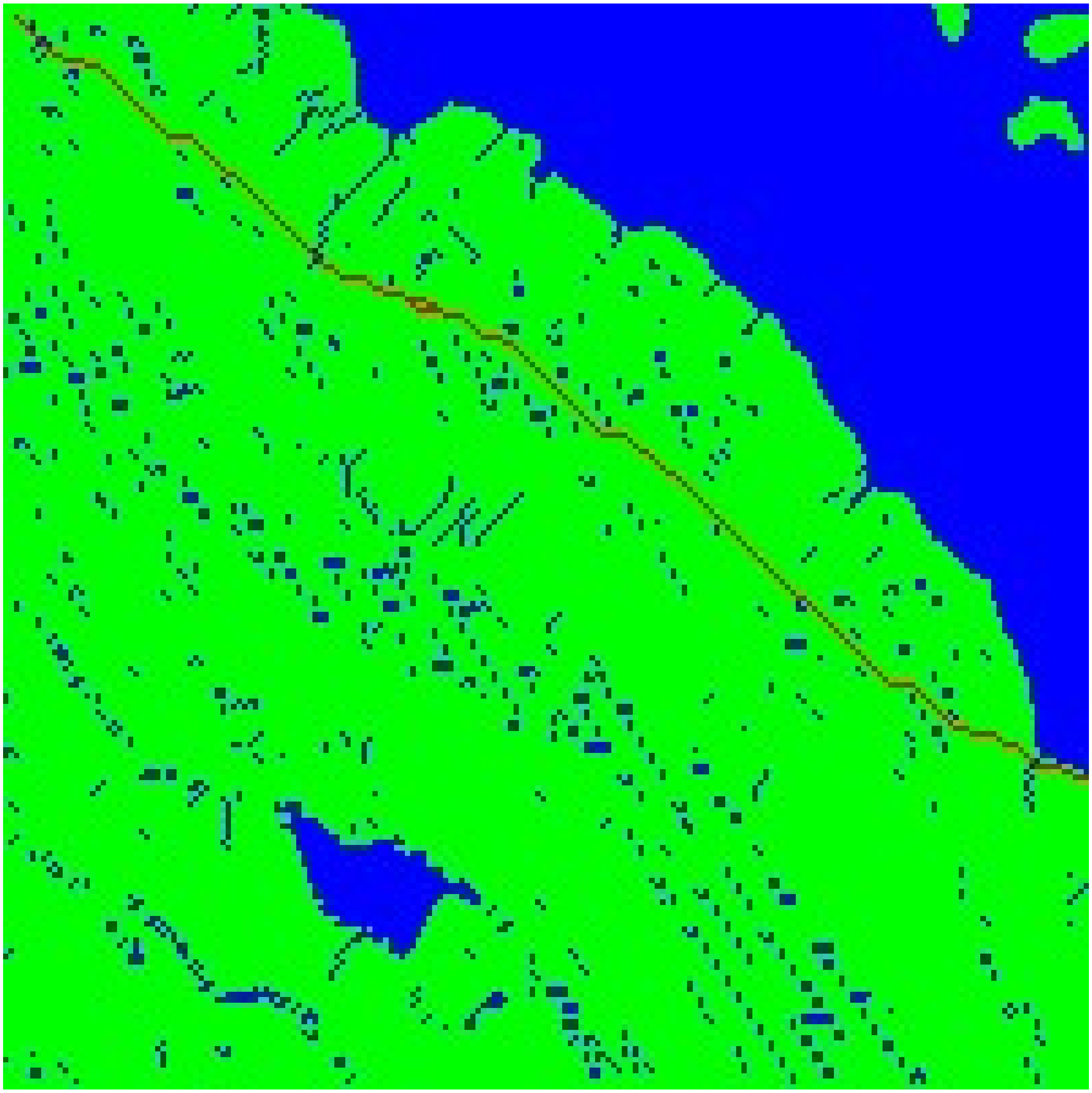


Figure 177: 10000BP Fall

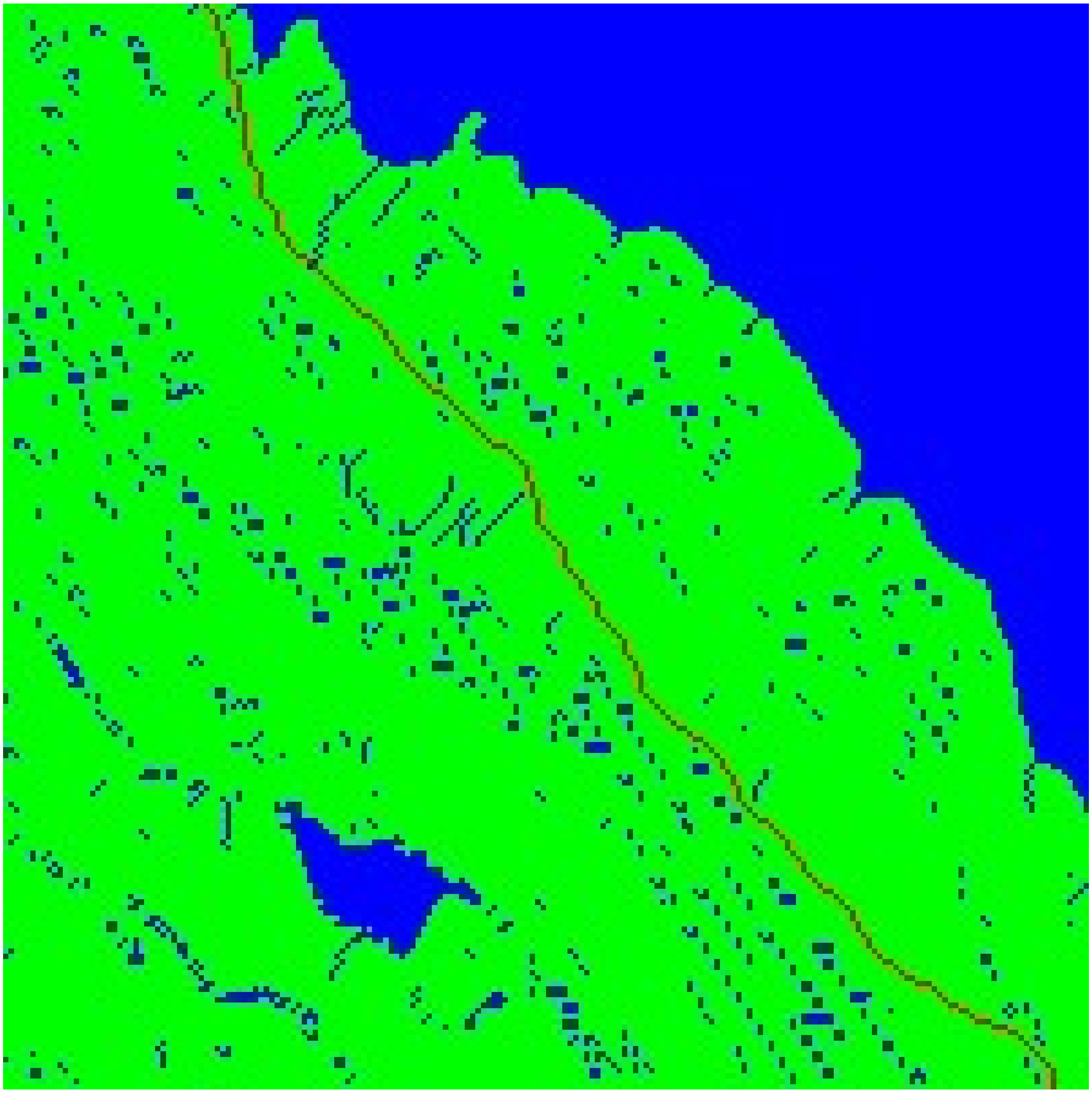


Figure 178: 9800BP Spring

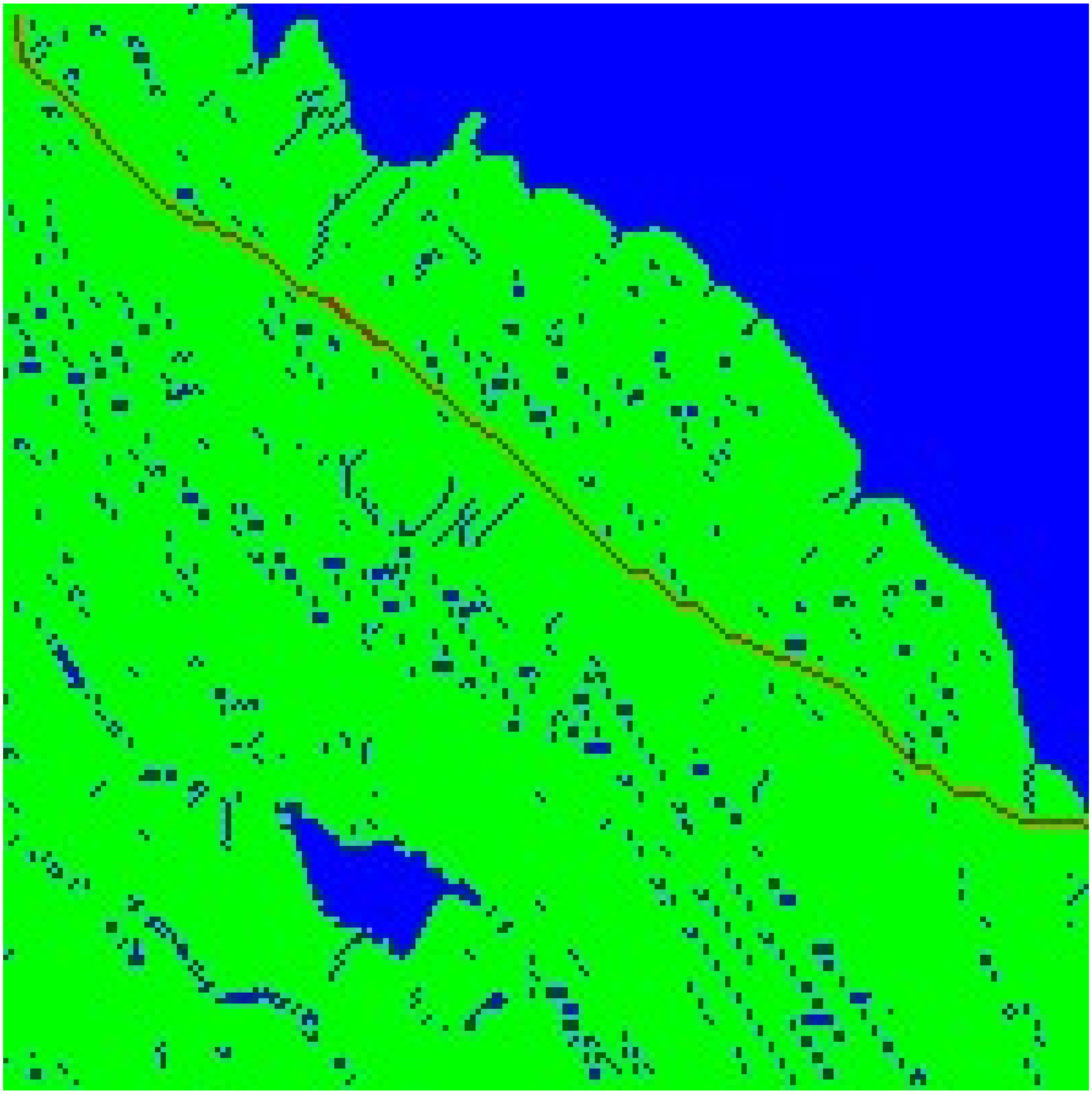


Figure 179: 9800BP Fall

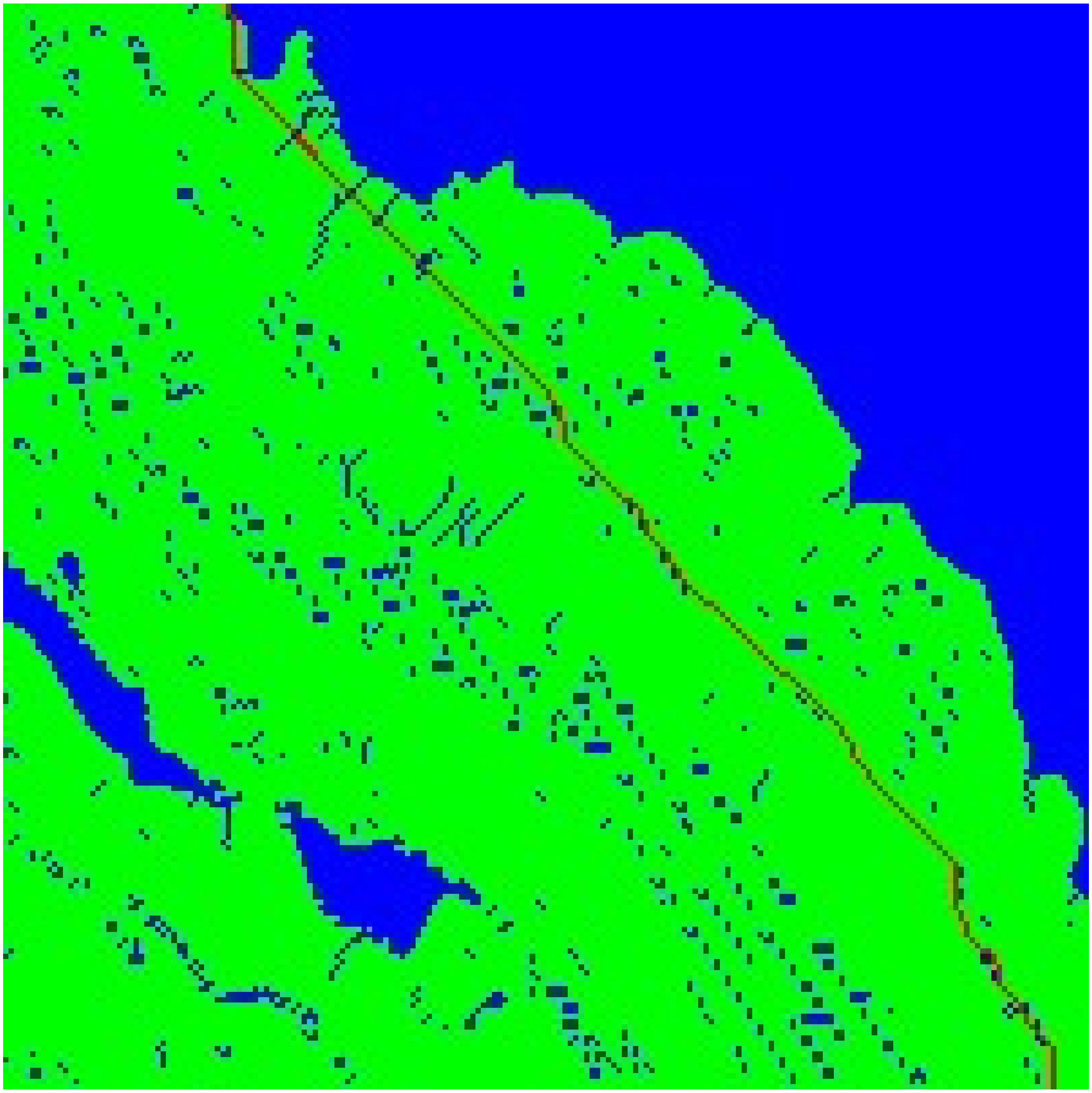


Figure 180: 9600BP Spring

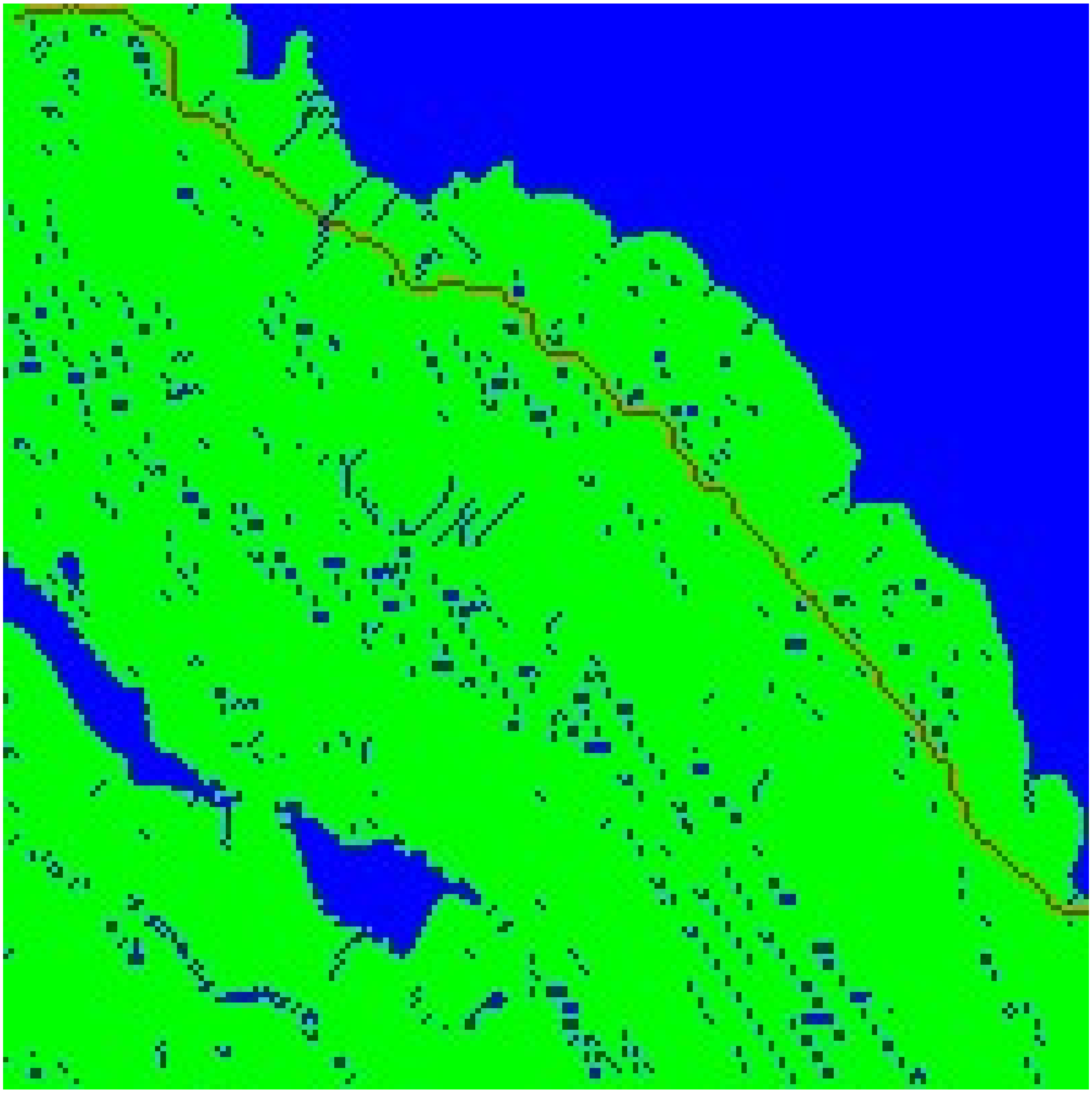


Figure 181: 9600BP Fall

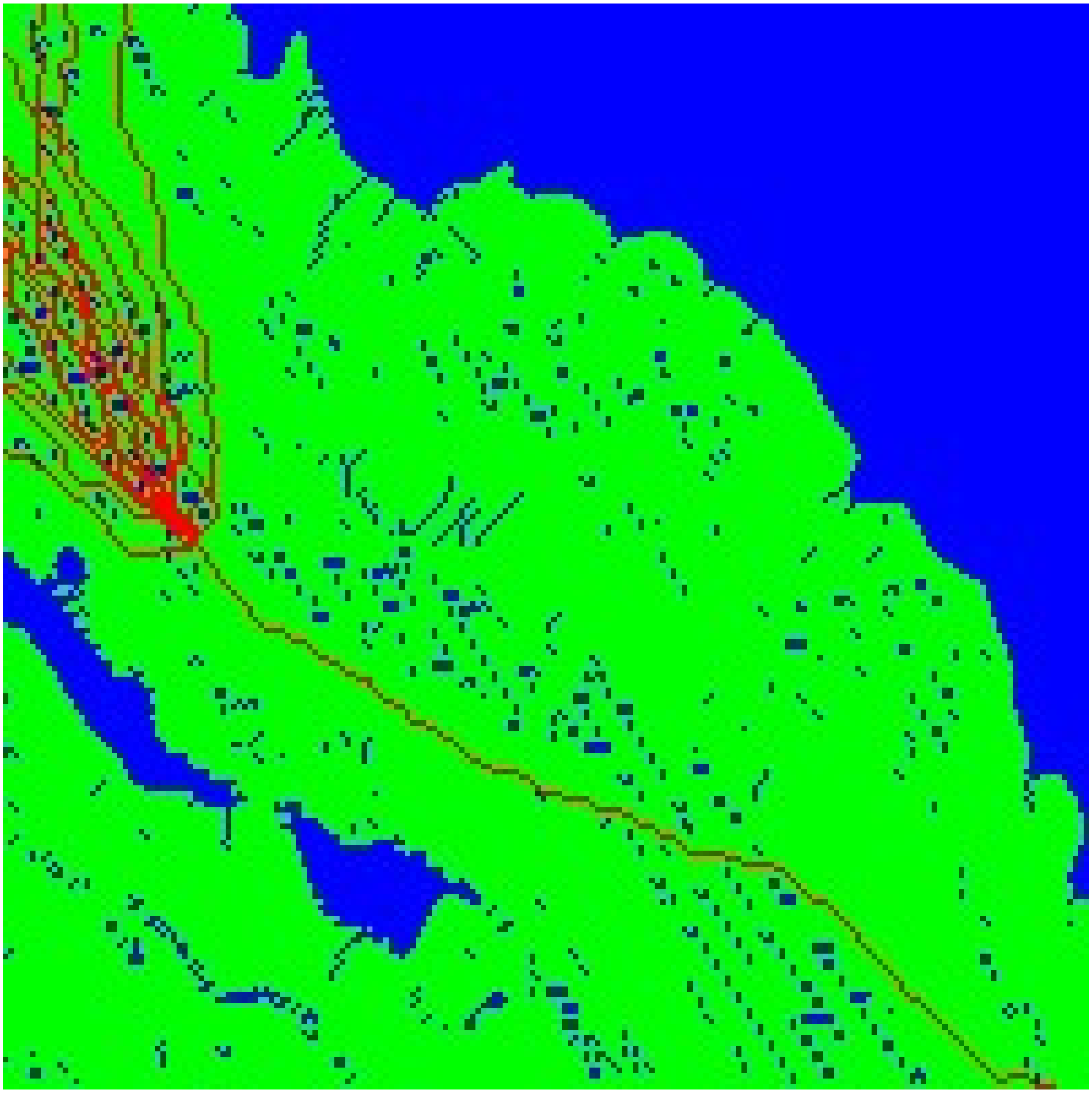


Figure 182: 9400BP Spring

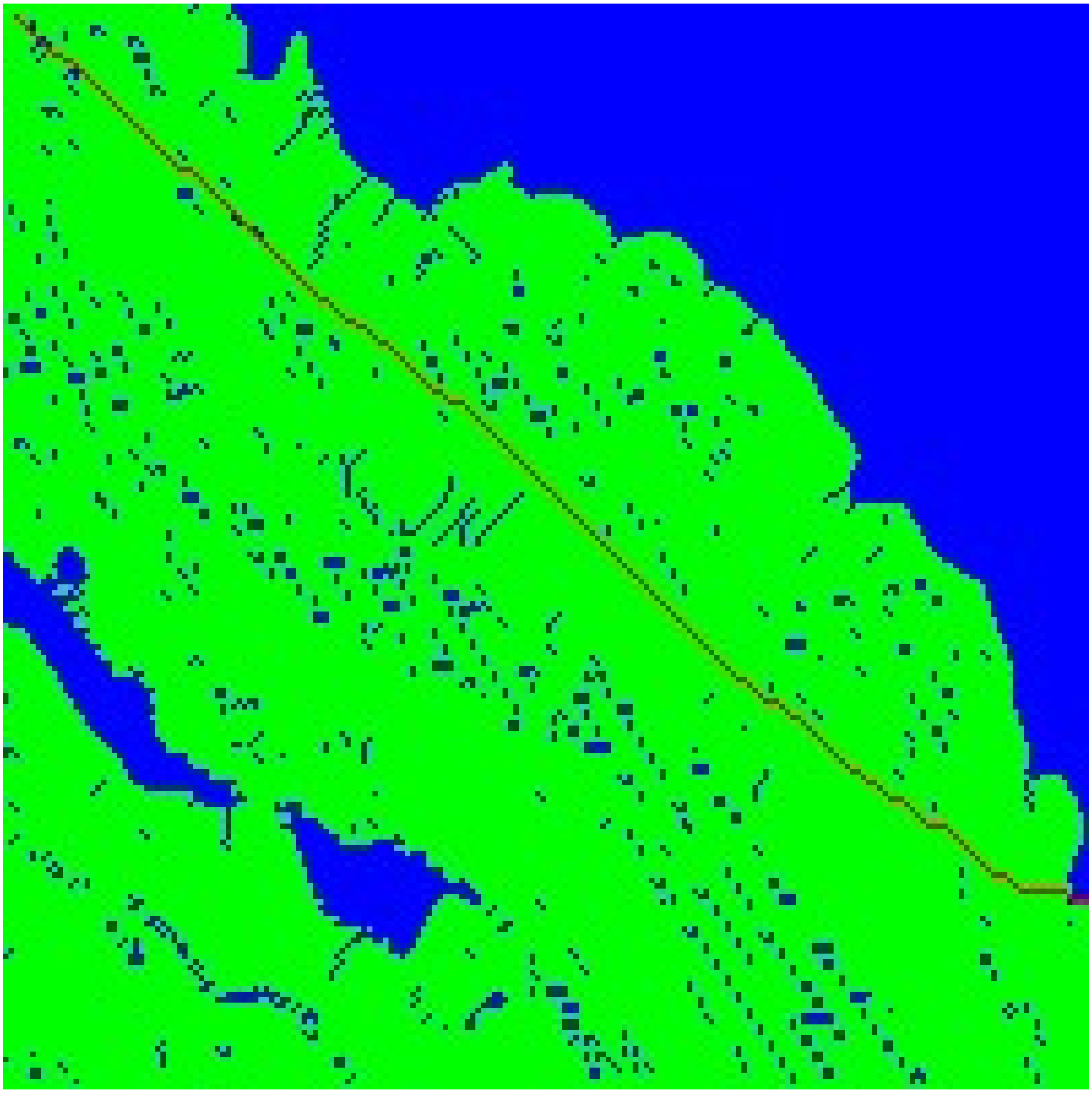


Figure 183: 9400BP Fall

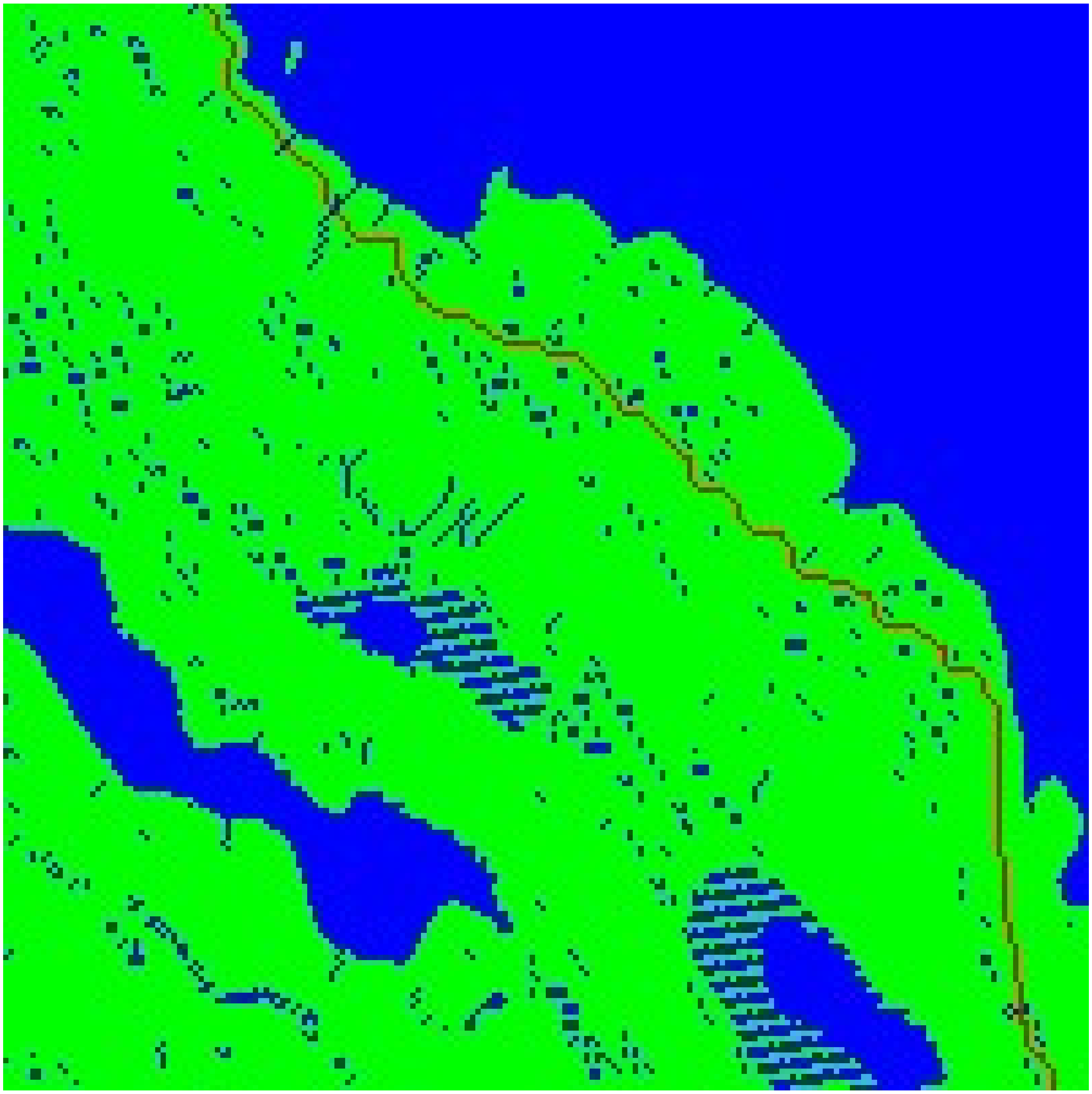


Figure 184: 9200BP Spring

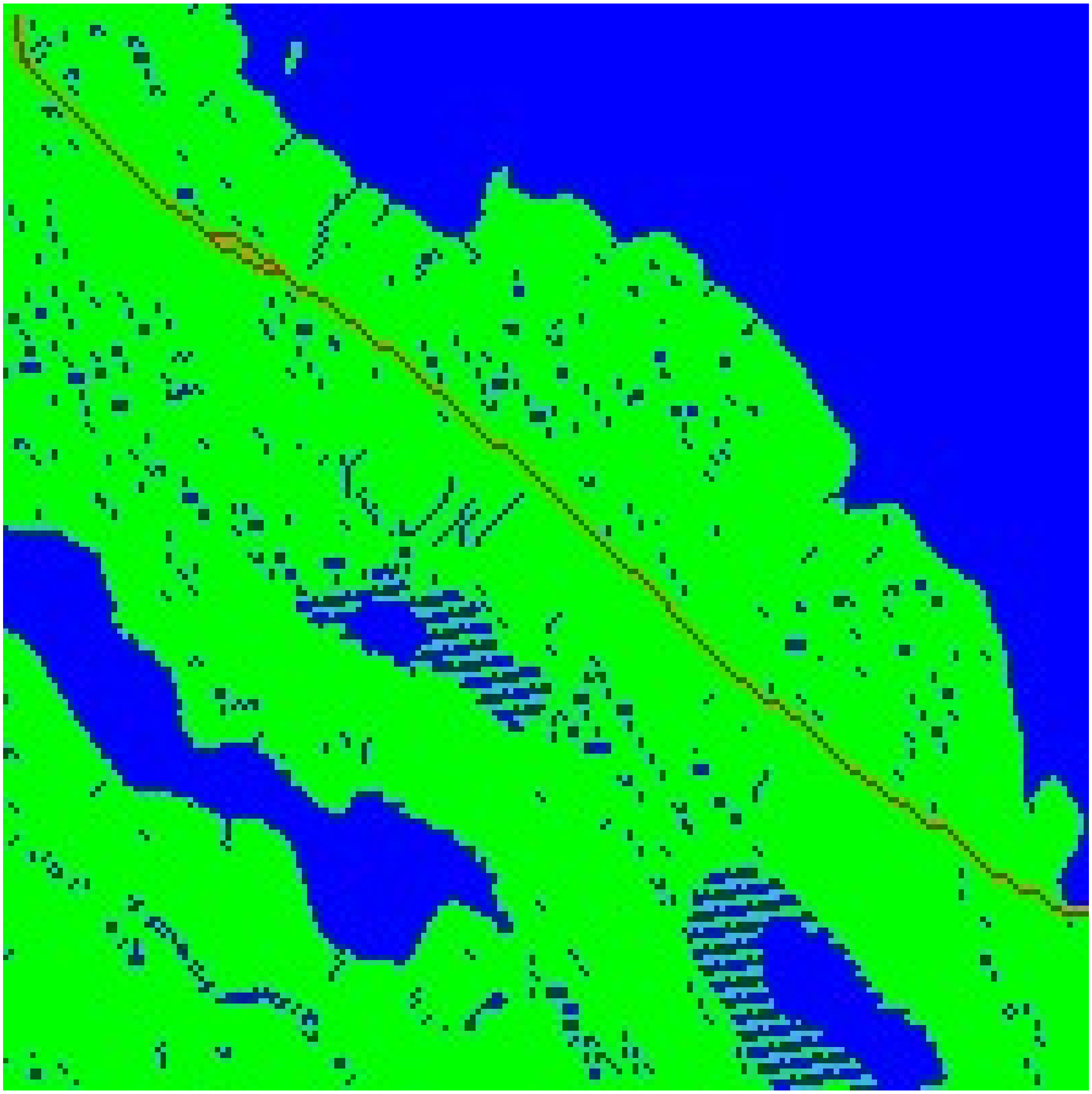


Figure 185: 9200 BP Fall

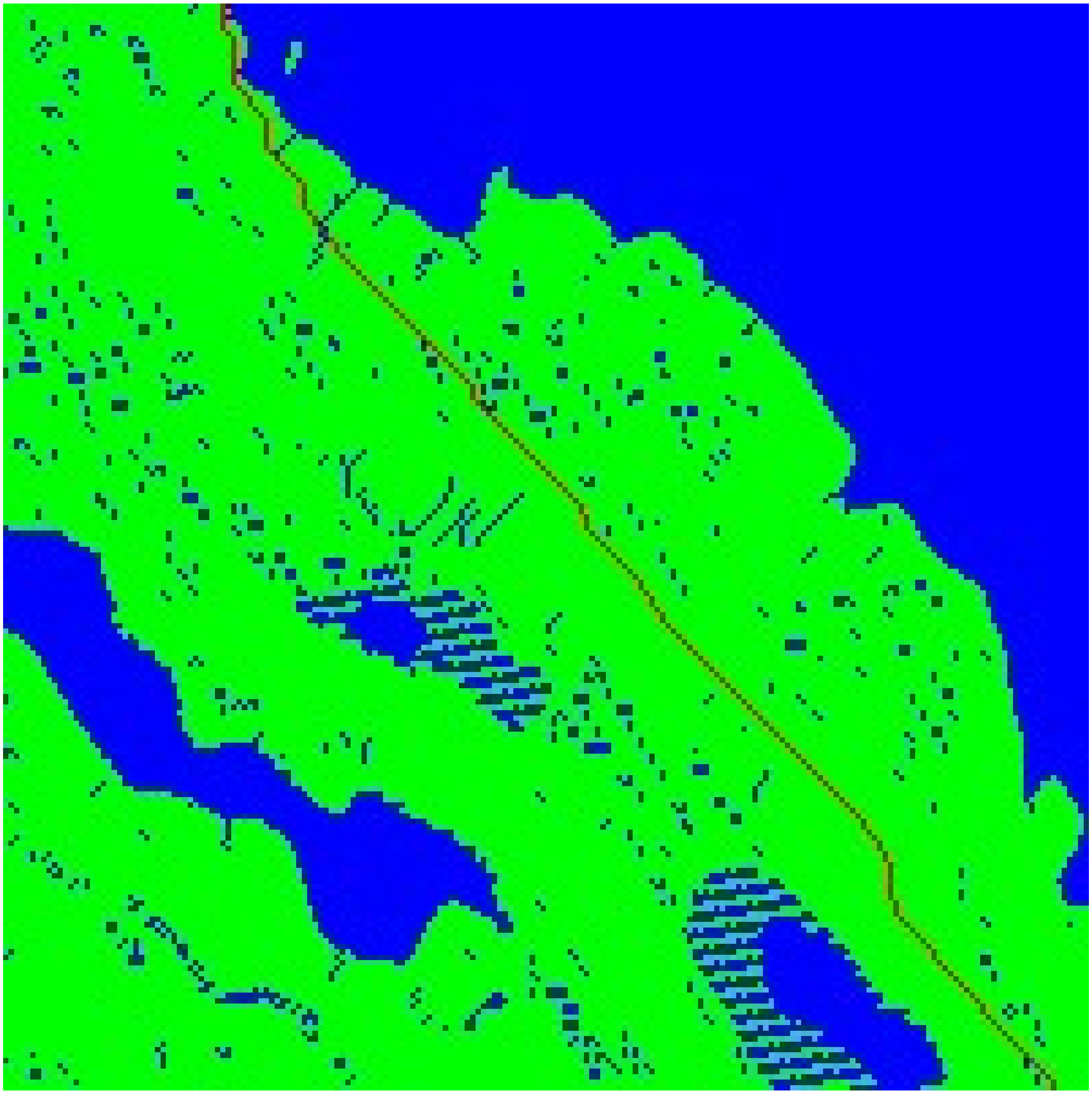


Figure 186: 9000 BP Spring

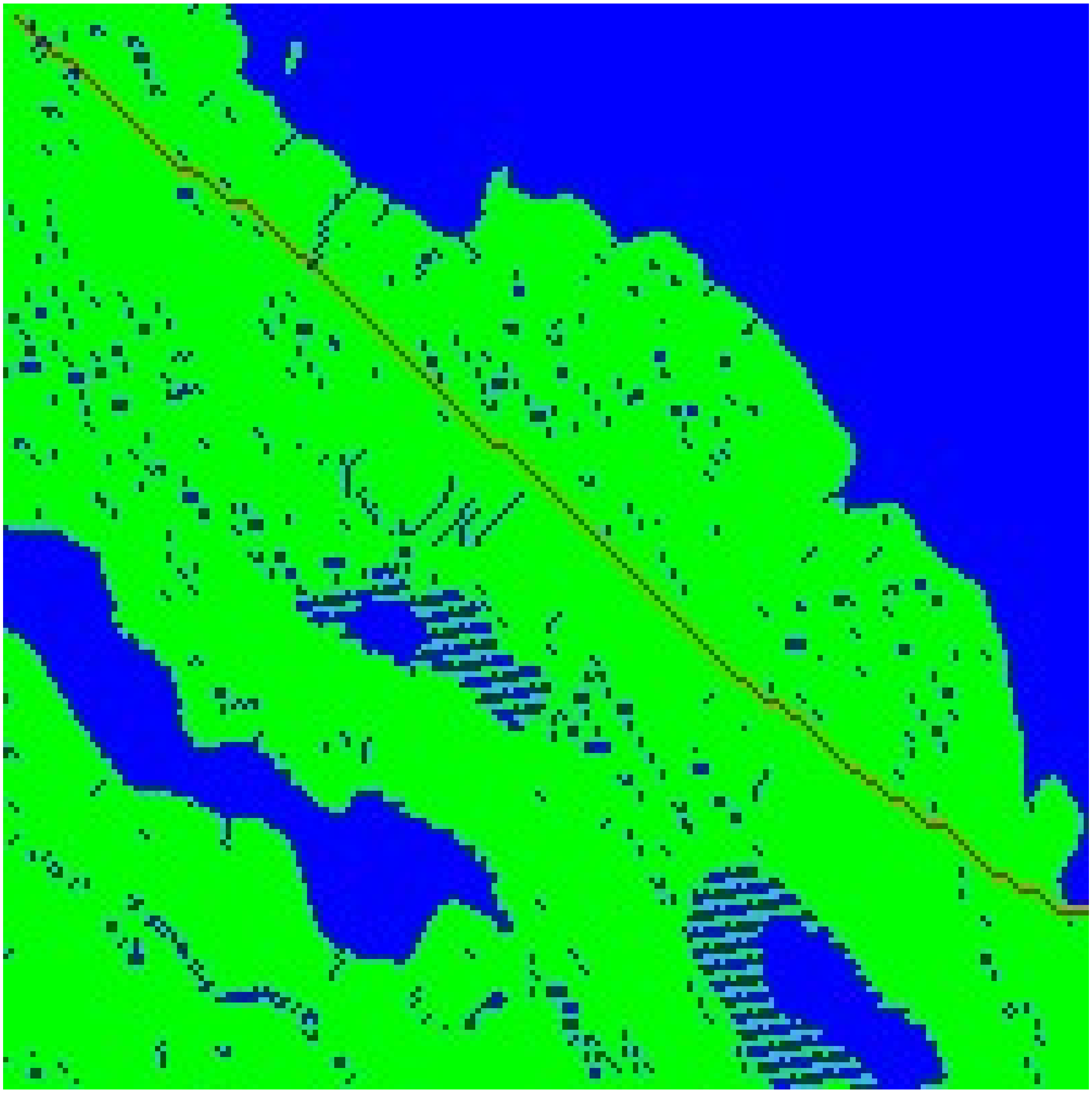


Figure 187: 9000 BP Fall

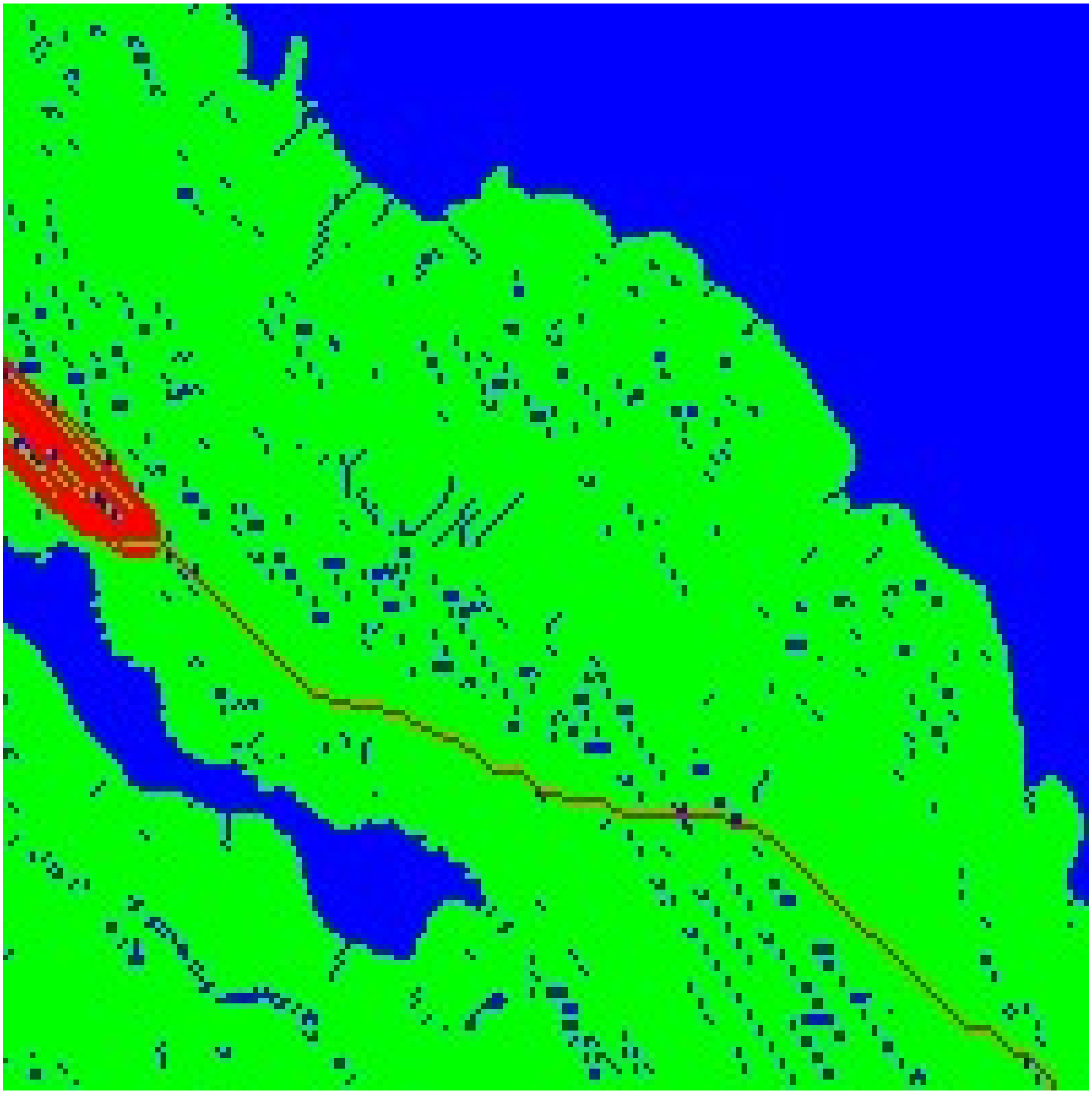


Figure 188: 8800BP Spring

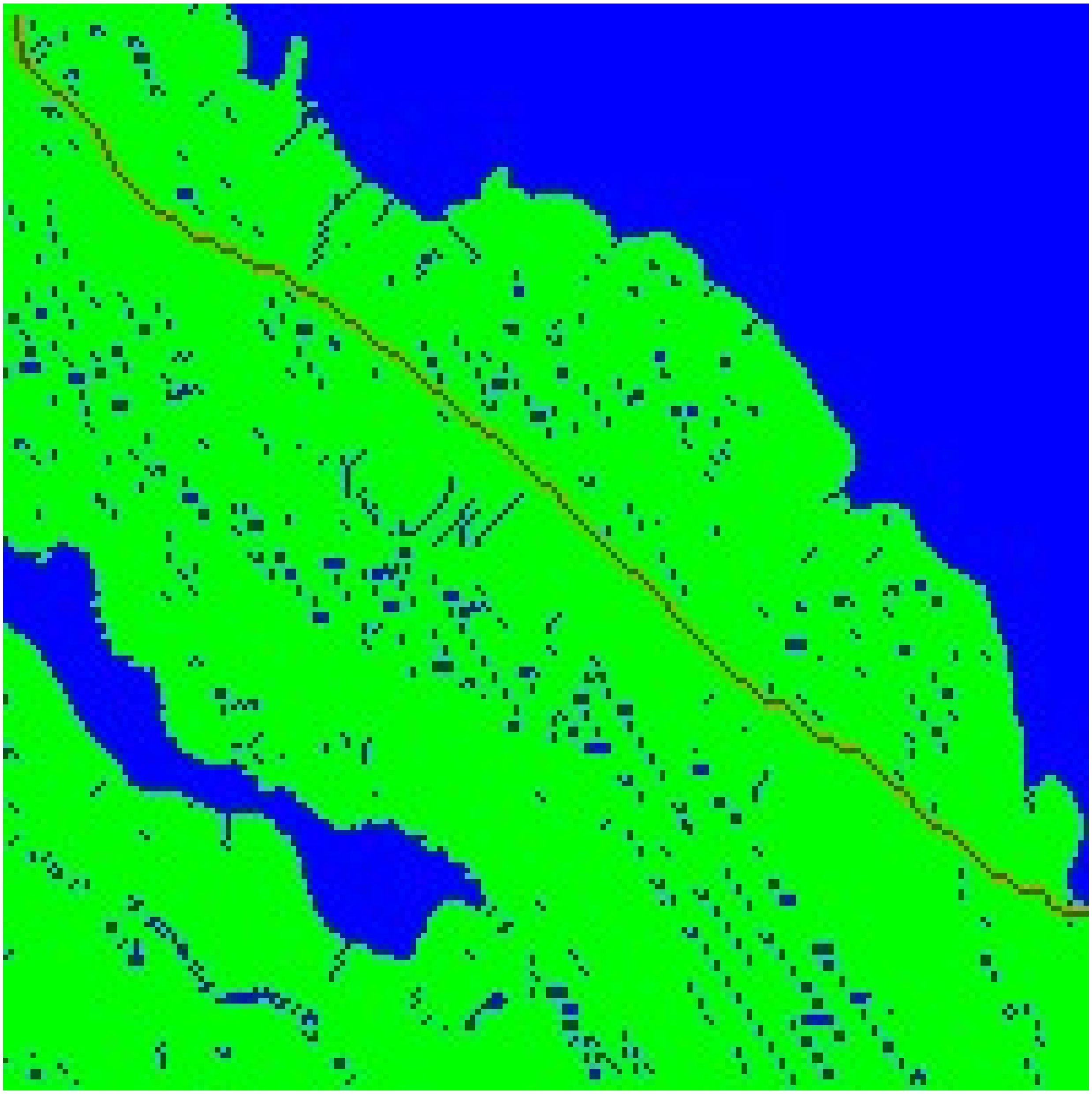


Figure 189: 8800BP Fall

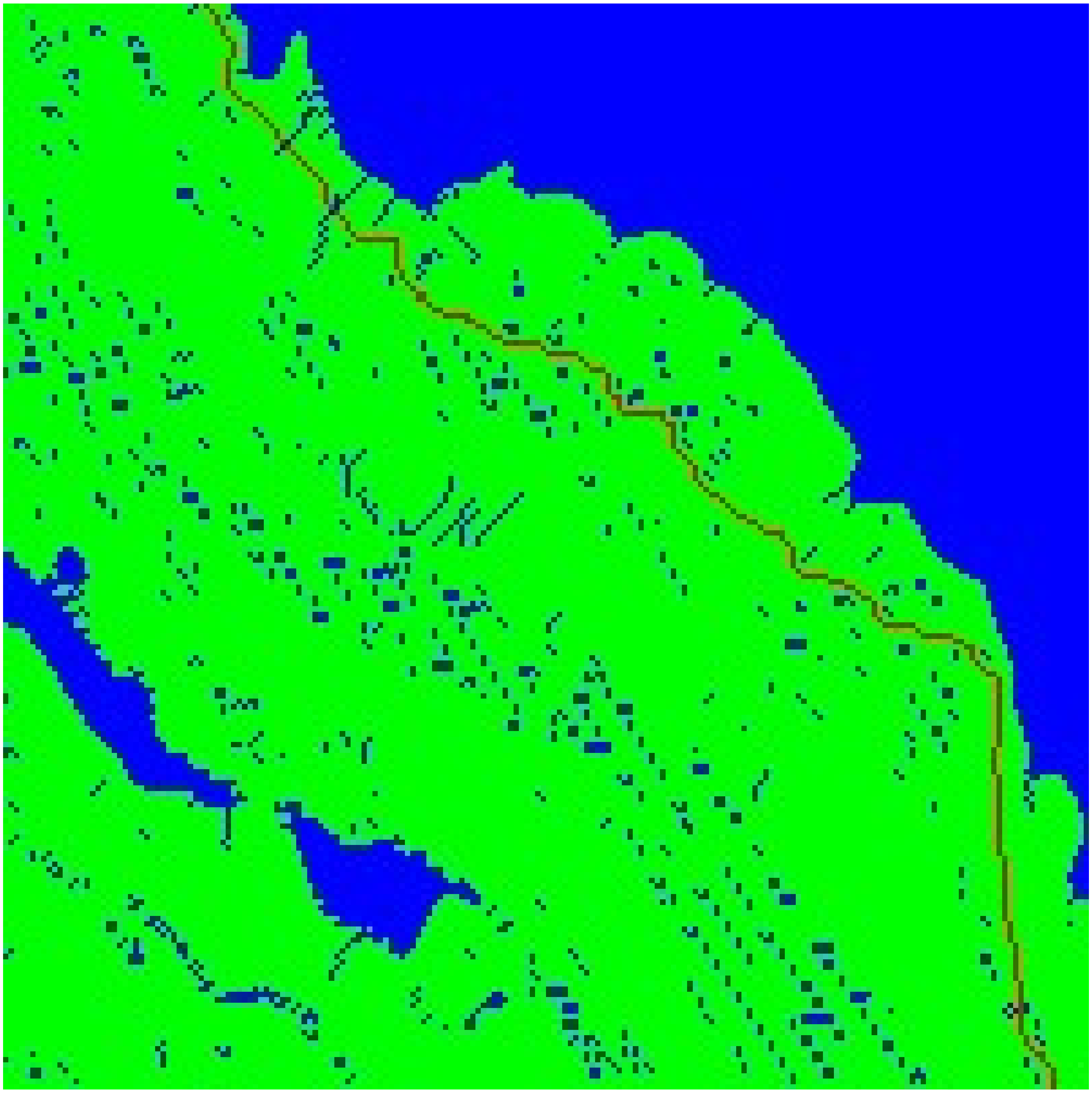


Figure 190: 8600BP Spring

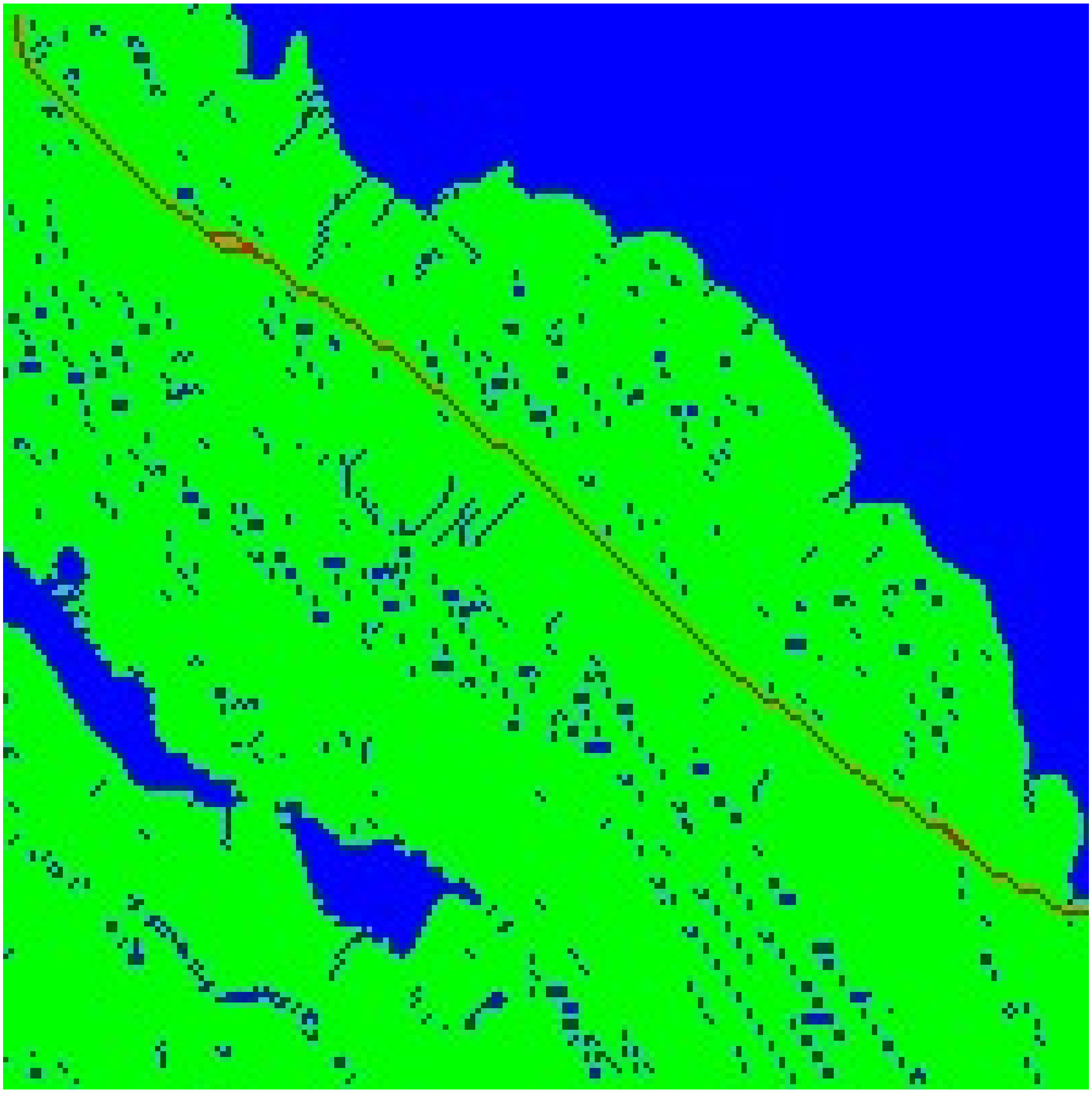


Figure 191: 8600BP Fall

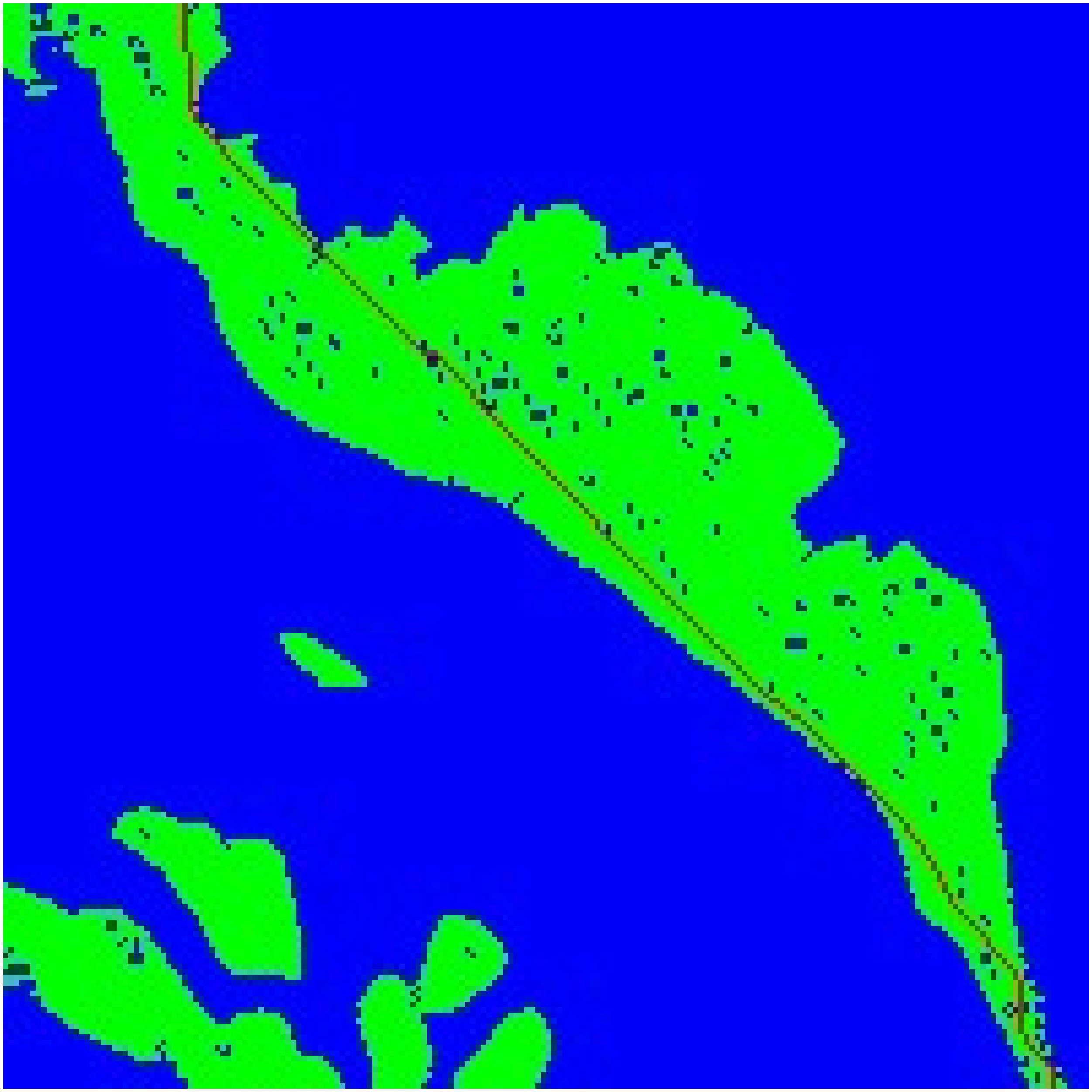


Figure 192: 8400BP Spring

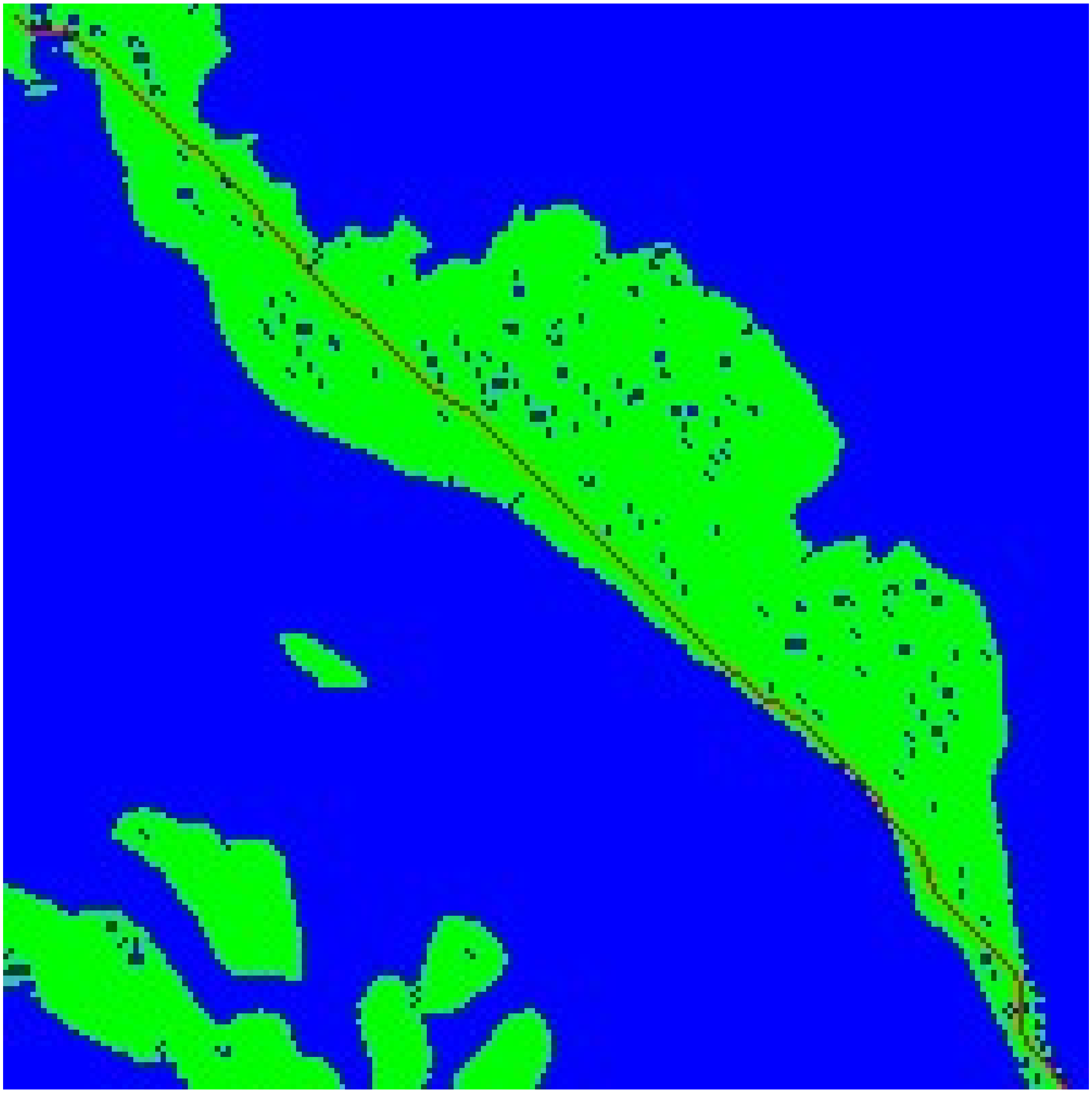


Figure 193: 8400BP Fall

ABSTRACT

CAPSO: A MULTI-OBJECTIVE CULTURAL ALGORITHM SYSTEM TO PREDICT LOCATIONS OF ANCIENT SITES

by

SAMUEL DUSTIN STANLEY

August 2019

Advisor: Dr. Robert Reynolds

Major: Computer Science

Degree: Doctor of Philosophy

The recent archaeological discovery by Dr. John O'Shea at University of Michigan of prehistoric caribou remains and Paleo-Indian occupational structures underneath the Great Lakes has opened up an opportunity for Computer Scientists to develop dynamic systems modelling these ancient caribou routes and hunter-gatherer settlement systems as well as the prehistoric environments that they existed in. The Wayne State University Cultural Algorithm team under Dr. Robert Reynolds has created such a dynamic virtual world system. We contributed by providing a rule-based expert system designed to predict locations potentially containing undiscovered occupational structures in the Alpena-Amberley Ridge Region. In order to evolve the rules and thresholds within this expert system, we also developed a Pareto-based multi-objective optimizer called CAPSO, which stands for Cultural Algorithm Particle Swarm Optimizer. CAPSO is fully parallelized and is able to work with modern multicore CPU architecture, which enables CAPSO to handle "big data" problems such as this one.

The crux of our methodology is to set up a biobjective problem with the objectives being locations predicted by the expert system (minimize) vs. training set occupational structures within those predicted locations (maximize). The first of these quantities plays the role of "cost" while the second plays the role of "benefit". Four separate such biobjective problems are created, one for each of the four relevant occupational structure types (hunting blinds, drive lines, caches, and logistical camps). For each of these problems, when CAPSO tunes the system's rules and thresholds, it changes which locations are flagged and hence also which structures are predicted. By repeatedly tuning the rules and thresholds, CAPSO creates a Pareto Front of locations flagged (i.e., "cost") vs. occupational structures predicted (i.e., "benefit") ordered pairs for each of the four occupational structure types. A visualizer system can produce a geographic map of the locations flagged and structures predicted corresponding to each of these ordered pairs, and archaeological teams can composite these maps in order to create an entire expeditionary season plan that suits their individual budgetary means and research goals.

We also analyzed the data trends within each of our Pareto Fronts, which can be thought of as "cost curves". Our analysis revealed that as the number of structures predicted (benefit) increases linearly, the number of locations predicted (cost) increases exponentially. Nonetheless, the low end of each of the cost curves was inexpensive enough such that even teams of more limited means could create a season plan using the low end of the cost curves. Finally, analysis of CAPSO's learning curves generated when constructing each of the Pareto Fronts demonstrated that complex learning was necessary in order to construct each of them.

AUTOBIOGRAPHICAL STATEMENT

Samuel Dustin Stanley received a B.A. degree from Kenyon College in 2008 with a major in Mathematics with a concentration in Statistics, a major in Philosophy with earned Distinction, an interdisciplinary concentration in Scientific Computing, and a minor in Astronomy. He received a M.S. degree from Wayne State University in 2013 with a major in Computer Science. 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 four marathons, with a best time of 4 hours and 24 minutes in the 2009 Detroit Marathon.