An Environment-Adaptive Navigation Model for Small, Sail-Powered Autonomous Surface Vehicles

EW402 Final Report

Author: MIDN 1/C D.A. Robinson
Date: May 4th, 2020

Adviser: CDR P. J. Frontera, USN

__________________ ___________
(signature) (date)

Department Chair: Prof. B. E. Bishop

__________________ ___________
(signature) (date)

An Environment-Adaptive Navigation Model for Small, Sail-Powered Autonomous Surface Vehicles

Student: MIDN 1/C D. A. Robinson Adviser: CDR P. J. Frontera, USN

***Abstract*—The current navigational approach within the USNA Sailbot program and many other entities developing small, sail-powered autonomous surface vehicles for long-distance travel relies on a preplanned set of waypoints for the vessel to follow along its voyage. This approach, however, does not utilize evolving weather patterns to the advantage of the vessel in the form of informed, adaptive route planning. This project aims to develop an active approach towards the maintenance of an accurate navigational picture by considering the effect of wind and surface current patterns on a small sailing vessel. Utilizing satisficing optimization techniques established within grid-based path planning, the model will be tested through tens of thousands of simulations of a west-to-east transatlantic crossing in an effort to assess and increase the likelihood of a successful voyage. The resulting solution will aid the USNA Sailbot program in developing a world-class autonomous sailing vessel capable of executing long-distance missions for competition and research.**

I. Background & Motivation

A. State of Technology

The field of autonomous vehicles is rapidly growing as applications for drones and driverless cars have grabbed worldwide attention as the next frontier of technological innovation. Among these high-profile platforms are robotic sea-going vessels, to include surface and sub-surface vehicles capable of impacting the maritime domain as drones have the sky and cars have the ground. As the need for renewable energy sources looms, wind is looked to as an attractive option for powering surface vessels. In combination, a small, sail-powered, autonomous surface vehicle (ASV) may represent the next breakthrough in maritime capabilities [1].

The current state of oceanographic research tools and methods is evolving rapidly as climate change, increased drilling, and interest in the Arctic have led to the need for more dynamic and cost effective solutions in collecting data in remote areas. Manned research vessels are capable of collecting the desired data, but are limited in their endurance in extreme environments and can be prohibitively expensive to operate [2]. Remote sensing methods such as satellite imaging and aerial photography are able to capture brief snapshots of large areas, but need a method of cross-checking large-scale imaging data with the reality on the surface [3]. Gliders and other underwater vehicles have relatively high stamina, but are dependent on small boats and other super-surface means of communication and positioning and thus are less suitable for open ocean missions [4]. Buoys are low-cost oceanographic data collection platforms that come in two main forms, moored and drifting. Moored buoys offer a specialized form of data collection by remaining on station, but are limited in broader application by immobility [5]. On the other hand, drift buoys are, by nature, uncontrolled and often become floating debris by the end of their service lives [2].

In an effort to bridge the gap between expensive, manned ships and uncontrolled drift buoys, the USNA Sailbot team is developing an economical, sail-powered ASV capable of maneuvering to desired locations while operating with unparalleled endurance, which will be demonstrated through the completion of a transatlantic voyage. The United States Naval Academy established an educational research program in 2006 with the intent of competition; however, while the program has enjoyed success in several short distance competitions, the goal of a transatlantic crossing has yet to be achieved [6].

B. The Microtransat Challenge

The Microtransat Challenge, founded in 2010, is a competition for surface vehicles to complete a transatlantic voyage under their respective class (sailing or non-sailing) and division (unmanned or autonomous). Each entry must pass through a designated start and finish line at opposite ends of the Atlantic Ocean, must be under 2.4 meters in length, and must provide frequent positional updates in order to avoid disqualification [7].

Figure 1. Current Hull in USNA's Santee Basin.

Each year, a small handful of teams from around the globe participate in the challenge; out of 26 entries since the competition's inception, the U.S. Naval Academy has six. As of the time this paper was written in May 2020, no team has ever completed the Microtransat Challenge under the sailing class, autonomous division. However, in August 2018, the *SB Met* from Norwegian company Offshore Sensing completed the voyage in 79 days in the sailing class, unmanned division, becoming the first to ever do so [8].

Figure 2. Satellite Data of SB Met's Successful West-to-East Voyage, In Red [9].

C. USNA Sailbot Program Developments

The current navigational system relies on a preplanned set of navigational waypoints. For many years, the Sailbot team would launch from Cape Cod, MA. This launch point proved to be in one of the most heavily fished areas on the Eastern seaboard, and the Sailbot would be picked up by a trawler before reaching the starting line of the Microtransat Challenge. In an independent research course in 2017, MIDN 1/C Matthew Marino revisited this navigation problem and selected a new route and set of waypoints that avoid heavily fished areas and also utilize the Gulfstream current, launching from Cape Hatteras, NC. This proposed route is shown below in Figure 3.

Figure 3. MIDN Marino's Proposed West-to-East Route.

While this route attempts to follow the general direction of the Gulfstream current, its rigid nature does not track the intricacies of the current and thus leaves this natural force underutilized. Of note, the Microtransat start and finish lines have since been expanded since MIDN Marino's work was conducted. This project will further expand on the idea that the Gulfstream current offers an advantageous effect on the Sailbot and its likelihood of crossing the Atlantic Ocean, a ~3000 NM voyage.

D. Overview of Optimization Principles

Metric path planning is generally concerned with finding the shortest distance or shortest time path from a known starting point to a desired endpoint in a 2D, gridded environment [10]. Traditional path planning algorithms for these scenarios measure the benefit of moving from one node to another through a cost function, a heuristic, or a combination of both. The basic evaluation function is

$$f(n) = g(n) + h(n) \quad (1)$$

where $f(n)$ is the relative benefit of moving to node n, $g(n)$ is the cost from the start point to node n, and $h(n)$, the heuristic, is the cost from node n to the desired endpoint [11]. In general, the more nodes that the algorithm has to search, the longer the processing time. For example, Dijkstra's algorithm and the A* algorithm perform exhaustive searches of every node in the environment, as do some wavefront algorithms such as Breadth-First Search (BFS) [12]. Thus, some algorithms are modified to narrow the search are to result in a faster solution, though this may cause the solution to be suboptimal.

Satisficing search algorithms are designed to determine near-best solutions. Ideally, the solutions are only slightly suboptimal but have a significant reduction in computation time [13]. An example of a suboptimal algorithm is the greedy best-first search algorithm, which derives its name from operating very locally and myopically. It relies on a heuristic designed to find the "next-cheapest" node in the immediate vicinity of the current node, under the assumption that piecing together a string of low-cost nodes will likely lead to the desired endpoint. The main limitation of this approach is that it is very susceptible to falling into local extrema as its global view is non-existent [14].

Zermelo's Navigation Problem (ZNP) is a classical control problem in which a boat is crossing a current field, and is concerned with finding the shortest-time path between Points A and B by optimizing the heading of the boat. Course notes from the Massachusetts Institute of Technology frame ZNP as set of functions and equations [15]. The problem formulation is to approximate a solution for the cost function

$$\min_{\theta} J = t_f \quad (2)$$

where t_f is the final time minimized with respect to heading θ, subject to the dynamics

$$\dot{x} = Vcos(\theta) + W_x \quad (3)$$

$$\dot{y} = Vsin(\theta) + W_y \quad (4)$$

where V is the constant reference velocity and $W_{x,y}$ are the magnitudes of the current in the x and y-directions, respectively. The constraints are defined as

$$x_{min} \leq x(t) \leq x_{max} \tag{5}$$

$$y_{min} \leq y(t) \leq y_{max} \tag{6}$$

$$000°T \leq \theta \leq 359°T \tag{7}$$

where x represents the longitudinal distance and y represents the latitudinal distance of the target area, and θ is essentially unbounded in the unit circle. The initial conditions are of the form

$$x(0) = lon_n \tag{8}$$
$$x(t_f) = lon_{n+1} \tag{9}$$

$$y(0) = lat_n \tag{10}$$
$$y(t_f) = lat_{n+1} \tag{11}$$

$$\theta(0) = \tan^{-1}\frac{y(t_f)-y(0)}{x(t_f)-x(0)} \tag{12}$$
$$t_f - unspecified \tag{13}$$

where $x(0)$ represents the initial x-position, $x(t_f)$ represents the desired final x-position, and y follows the same convention. $\theta(0)$ is defined as the straight-line heading between the start and end point as it is a reasonable selection for the initial heading.

The solution to Zermelo's Navigation Problem is well-suited for a continuous navigation problem. However, in the case of an autonomous, sail-powered vessel only capable of intermittent communication with an operating scope as large as the Atlantic Ocean, a solution using continuous navigation is both unnecessary and infeasible. Instead, it is useful to discretize the problem to a level that balances local decision making with forward projecting capabilities.

II. PROBLEM STATEMENT

Consider an ASV navigating from waypoint to waypoint, subjected to a variety of wind and current fields that are unaccounted for with traditional navigation models. Due to the limited control authority of the test vessel (under 2 meters and wind-powered), the following assumptions will be made.

Assumption 1: The boat sails at a low velocity (< 3 kts), making surface current and wind near-equal factors in affecting navigation.

Assumption 2: The boat will, in general, follow the ocean surface current but is capable of overcoming small current forces (< 1.6 kts) with sailpower.

Assumption 3: An off-hull server can process an online database of weather and oceanographic data based on GPS position, and subsequently transmit an updated route solution to the boat via satellite communication.

Under these assumptions, this project will demonstrate a weather-adaptive navigation model that considers wind and surface current in determining new waypoint information to guide the navigation of the Sailbot during a Microtransat voyage.

Figure 4. Flowchart of Data Processing, to Formulation of Route Solution, to Navigation Action by the Sailbot.

III. Literature Review

Recent developments in maritime navigation have focused on creating algorithms for optimal ship routing. Optimal ship routing has wide applications in national defense, commercial travel, oceanography, shipping, logistics, and the oil industry as organizations seek to reduce expenses while maintaining the safety and comfort of personnel on board. In general, "optimal" comprises a variety of factors such as shortest distance, shortest time, lowest fuel consumption, and safest route. Depending on particular application, an algorithm may include one or more of these elements when determining the optimal route.

Traditional algorithms, such as Dijkstra's algorithm, are entirely concerned with shortest distance from point to point. Dijkstra's algorithm does not involve a heuristic, only relying on information gained along the search. This, however, can be computationally expensive and thus limiting in its ability to provide a timely route solution. One group has improved upon this model by reducing calculation cost through informed search techniques that lower the number of nodes that must be processed in order to find the shortest path [16]. This practice of decreasing the search area proved just as accurate as Dijkstra's algorithm but computed in a shorter amount of time.

Wavefront propagation is used to describe the expansion of nodes in various search algorithms, and is particularly useful in visualizing path planning along graph or grid-based domains. The name is intuitively derived from a wave pushing out in all directions analogous to conducting a search. Ghai and Shukla discuss slight nuances within this method, such as Focused Wave Front (FWF), Modified Wave Front (MWF), and Optimally Focused Wave Front (OFWF) algorithms [17]. The main differences between these are search area size, weighted costs, optimality threshold, and degree of greediness (disregard for heuristics).

Many path planning algorithms involve some element of a heuristic, or estimation of cost/distance to go until the goal is reached. This, combined with gained information about the path already traversed, makes up what is commonly known as the basic A* algorithm. An important characteristic of the A* algorithm is that the heuristic used can vary the search outcome along the spectrum of suboptimal but faster, or optimal but slower [18]. Techniques have been introduced to the traditional A* algorithm so that a better balance can be found, i.e. very close to optimal yet computed much faster than truly optimal.

Du and Kang introduce a route planning technique for autonomous sailboats based on the A* algorithm that takes boat heading and speed as decision variables that feed into state variables of boat position and sailing time [19]. Utilizing a gridded coordinate system from Point A to Point B, the model is tested by minimizing sailing time in both fixed and variable wind fields while considering the polar diagram of the vessel and penalties associated with tacking and jibing. The authors describe two techniques of dynamic programming, forward and backward, that intuitively originate from the start and endpoints, respectively. In their model, they use forward dynamic programming and note a positive correlation between total length of the route and minimum sailing time, as long as there is a stern or beam wind.

Heusner, Keller, and Helmert explore the less-understood realm of non-optimal, or *satisficing*, algorithms. They determined that the characteristics of a good heuristic used for optimal algorithms such as A* can actually be worse for suboptimal algorithms. A popular example of a non-optimal search algorithm is the greedy best-first search (GBFS) algorithm, which operates under the assumption that "states with lower heuristic values are on a cheaper path to the closest goal state." [20]. As such, GBFS is susceptible to local extrema and does not guarantee that a resulting solution is optimal. Additionally, within the family of GBFS algorithms, most only differ in their tie-breaking strategy.

Zyczkowski discusses a technique based on Dijkstra's Algorithm that specifically relates to sailboat routing in a gridded environment. The boat's modeled position along the grid is a function of all of the points and edges in the domain as well as the boat speed and combined wind and current force [21]. The model has an adjustable resolution in which the boat can move among 4, 8, 16, or 32 units around its current location. A matrix is produced out of the polar diagram of the boat and the associated potential speeds to each gridpoint given the wind and current information, acting as a pseudo-cost function for the model to determine shortest travel time.

Lopez analyzes the capabilities and limitations of a variety of path planning algorithms. He argues that if the domain can be modeled graphically, A* or Dijkstra's algorithms are well-suited for the task [12]. However, these two models are limited in that they perform a complete search of the environment, which is not ideal for complex and/or large problem sizes. Another classification he discusses is global vs. local planning, though a combination is usually necessary. Global planning is when the operating environment is fully known ahead of time. Likewise, local planning is useful in uncertain and dynamic domains, exhibiting reactive and adaptive characteristics. A* and Dijkstra's algorithms are less capable in these types of real-time planning domains.

These references all contribute to this project as potential methods for implementing an improved navigation model for the next Sailbot to be launched. The short-term goal of the project—to complete the Microtransat Challenge under autonomous classification—can usefully be classified as a suboptimal navigation problem, as the start and finish lines are large enough to render a truly optimal solution unnecessary. Furthermore, the ocean environment can be constructed into a grid comprised of latitude and longitude, and the vector fields of wind and current play a large role in the cost associated with moving from one point to another. Complexity of this problem will be reduced where possible through assumptions and techniques aimed at reducing search area while preserving local and global awareness. As such, the environment-adaptive navigation model proposed most closely resembles a greedy best-first search algorithm.

IV. Materials & Methods

This project uses computational studies to aid model development. Once a comprehensive model is constructed, a simulation environment is developed to test the model. These methods provide insight into the limitations of the current navigational approach and offer a more learned and adaptive alternative.

A. Computational Studies

In MATLAB, the *fmincon* function is used to solve ZNP. This function considers the cost function, linear and nonlinear constraints, bounds, and initial conditions of the problem scenario to return an array of headings that comprise the shortest-time path between two points in a vector field. For this computation, the *fmincon* function is run over a 1° x 1° area with the u- and v-components of the current in this area averaged separately. Additionally, 20 timestamps are created to force constraints to be met at regular intervals along the path. As the boat dynamics rely on a reference velocity V, two simulations are run, one modeling the boat's velocity in near-zero wind (essentially drifting) and one modeling its speed in a fair amount of wind around 10 kts. The results for the fair wind condition ($V = 1\ kt$) are shown below in Figure 5.

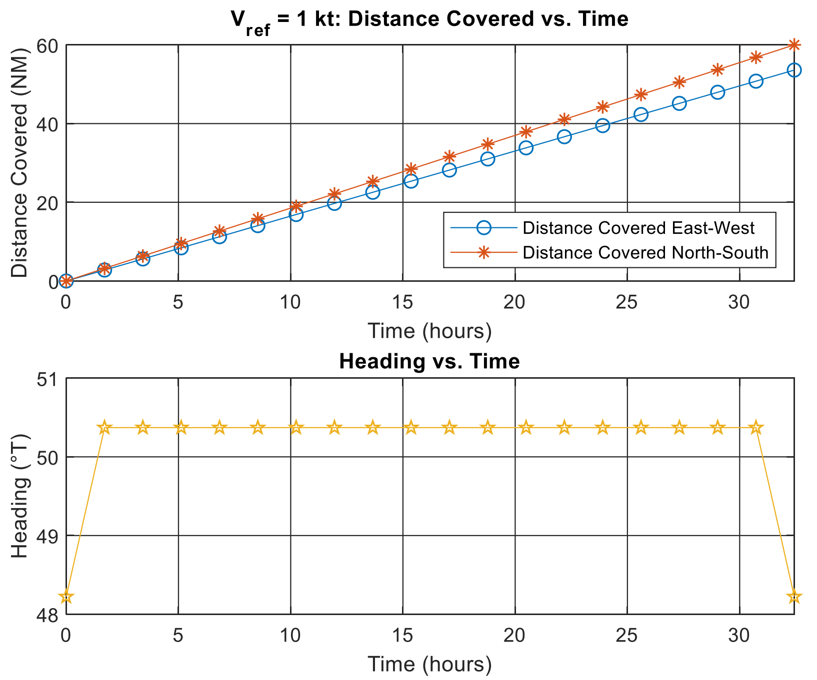

Figure 5. Distance Covered and Heading vs. Time for V = 1 kt.

Figure 5 reveals that the total time from the start point to the desired endpoint is 32.4 hours, covering 53.6 NM east and 60 NM north for a total of 80.5 NM. Over this distance, the heading stabilizes between 50 and 51°T in order to reach the desired endpoint. The two endpoints on the Heading vs Time subplot represent the straight-line heading (SLH) between the start and endpoint if there were no environmental factors at play. Thus, deviation from the SLH indicate correction for the effects of current. An interesting property of the *fmincon* output is that the algorithm essentially calculates a "crab angle" or "side-slip angle" that the boat points to, and then is carried by current along a course to the desired endpoint. Figure 6 below gives a better visual representation of the boat's desired straight-line path (SLP) to the desired endpoint, the path of the current, and the heading necessary to correct for the current.

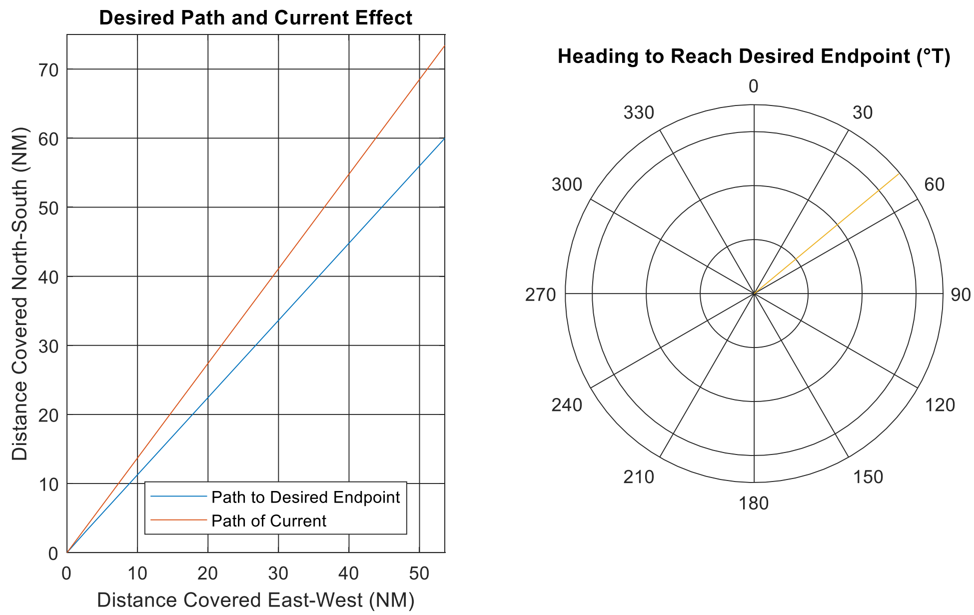

Figure 6. Desired Path, Current Effect, and Adjusted Heading for V = 1 kt.

Figure 6 shows that the current wants to push the boat to the north of the desired SLP, so the heading is adjusted to the south to compensate. With a reference velocity of one knot, this compensation is relatively subtle. The *fmincon* function is run again with a reference velocity of 0.1 kts to simulate near-zero wind, results shown below in Figure 7.

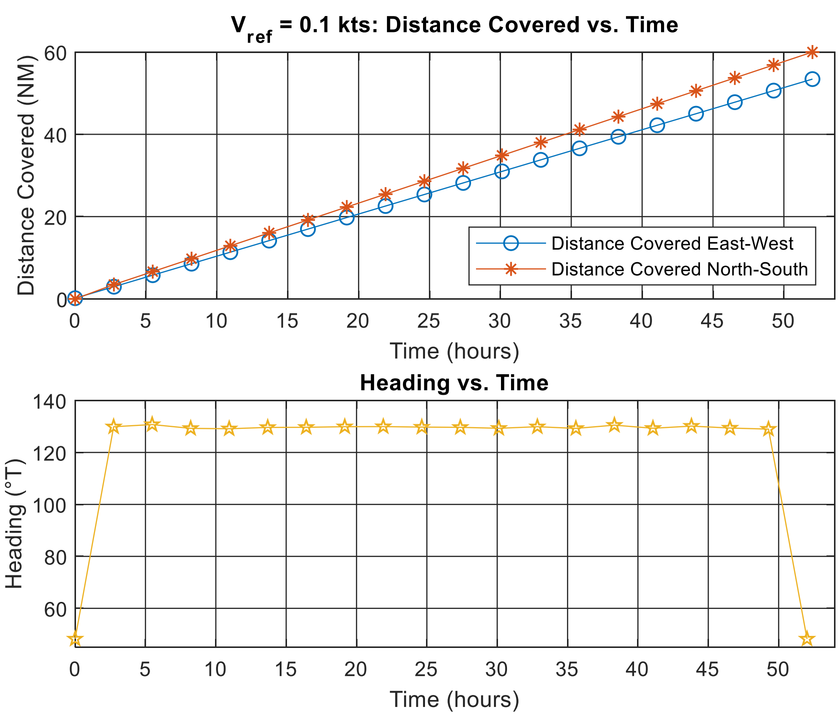

Figure 7. Distance Covered and Heading vs. Time for V = 0.1 kts.

Figure 7 shows that the total time of this leg is 52 hours, over 50% longer for the same distance than with a reference velocity of one knot. Additionally, the heading is altered significantly to about 130°T. Figure 8 below gives a better visualization of the extremity of this heading in the context of the desired SLP and effect of current.

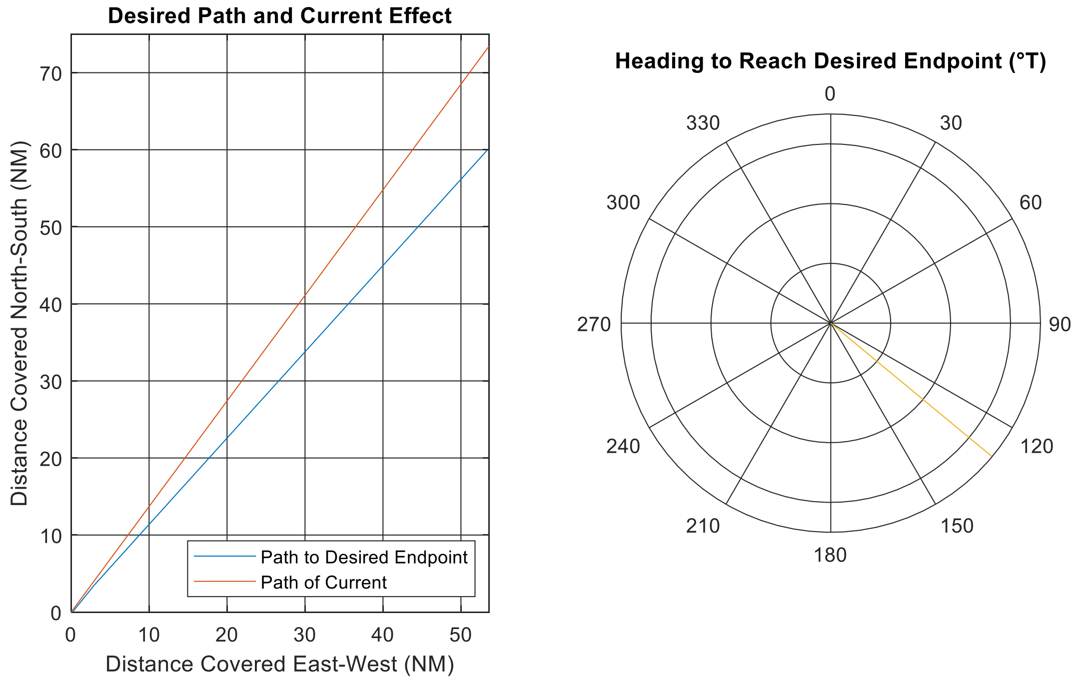

Figure 8. Desired Path, Current Effect, and Adjusted Heading for V = 0.1 kts.

At this slow speed, the algorithm is forced to select a very extreme side-slip angle to compensate for the effect of the current. With no means of self-propulsion, this algorithm essentially commands the boat to sail broadside and drift along with the current.

The key limitation of the *fmincon* approach to this problem is that it is designed for a vessel operating at a constant velocity. For a small sailing vessel that derives a majority of its speed from the variable weather fields it is in, constant velocity is not a reality. Thus, a more sensitive approach must be developed that lends itself to utilizing the environment as its primary source of speed, not necessarily as a factor to be overcome. Additionally, ZNP and *fmincon* are concerned with computing the shortest-time path. While time is a factor, the nature of autonomous sailing means it is minimally limited by endurance and therefore less concerned with getting to the destination quickly. Because the scope of the Microtransat Challenge is so large, continuous optimization provides diminishing returns when a discrete, satisficing optimization approach may provide similar results with much less complexity.

B. Model Development

The nature of the Microtransat route optimization problem as discrete and satisficing suggests a more nuanced and intuitive approach. The Gulfstream current offers a favorable route due to its strength relative to the Sailbot's projected speed and its natural direction of movement towards the northeast Atlantic and the Microtransat finish line. Figure 9 below shows an intensity map of the Gulfstream current.

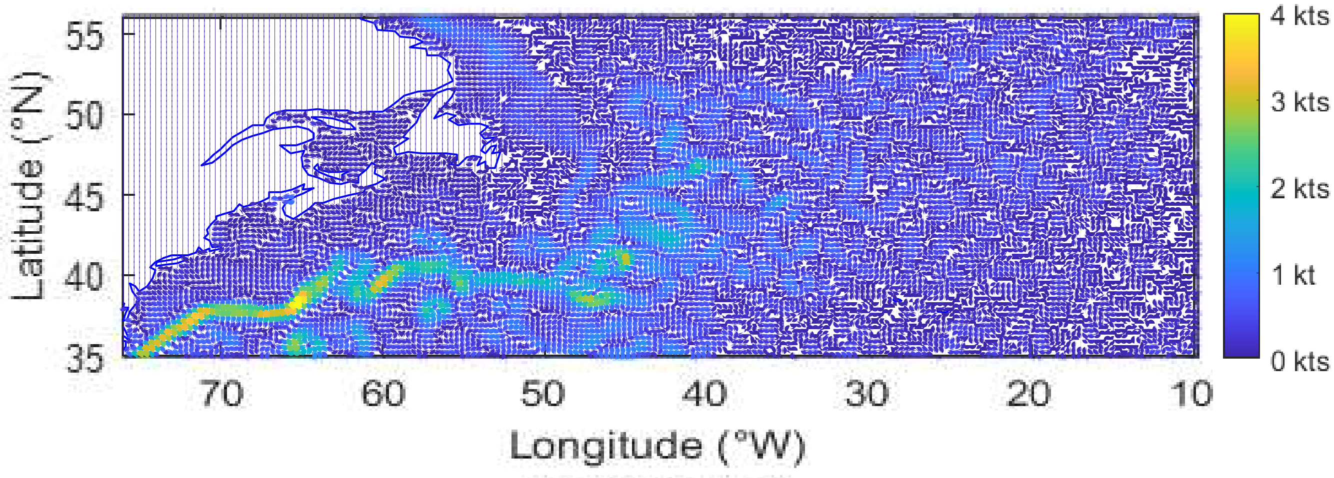

Figure 9. Snapshot Intensity Map of Gulfstream Current.

The strength of the Gulfstream current is highlighted in this speed map in contrast to much lesser surrounding ocean currents. Catalyzed by MIDN Marino's work, it has become apparent to the Sailbot team that following the Gulfstream current is intuitively advantageous to the boat and will increase its likelihood of a successful voyage. However, the Gulfstream is not enough to carry the boat the full length of the Microtransat Challenge, as evidenced by the rapid decline in flow density and speed about halfway across the Atlantic where it divides into the North Atlantic Current and the Canary Current [22]. The max surface current speed of this particular map is just over 4 knots, though some estimates argue max speeds close to 5 knots [23].

To a small autonomous sailboat, a four to five-knot current is a significant force. For reference, the average speed of the *SB Met* during its successful west-to-east transatlantic voyage in 2018 was about 1.45 knots. The proposed navigation model aims to follow the strength of the Gulfstream current to propel it towards its desired endpoint. The current is roughly 50 NM wide, and can shift seasonally by upwards of 20 NM [24]. Thus, a moderate level of precision is required to maintain navigation within the strength of the Gulfstream, yet discretion is also required so as to not end up in an eddy current that may drag the boat fatally to the north or south. Additionally, the effect of wind on the boat is on a sliding scale relative to current speed. The higher the current speed, the lesser of a role wind plays. Conversely, as the current begins to break down around 40° W, wind becomes the more dominant force for the remainder of the voyage. With these factors considered, a balanced approach is needed to utilize the Gulfstream while it is favorable, then operate with a focus on sailpower once these benefits have been spent.

The Matrix Mapper Method is designed to account for these nuances and dynamically shifts its decision making depending on the balance of wind and current forces at a given location and time. Designed in MATLAB, its inputs are current datasets pulled from the U.S. Naval Academy Oceanography Department's Environmental Data, Monitoring, and Prediction System (EDMAPS) [25], and wind data from NOAA's Physical Sciences Laboratory [26]. The fidelity of the current models are 1/3° latitude and longitude, or about 20 NM. Wind data is given at a lower resolution, but is interpolated to match that of the current. Thus, three layers of equally gridded matrices comprise the environmental map from which analysis takes place.

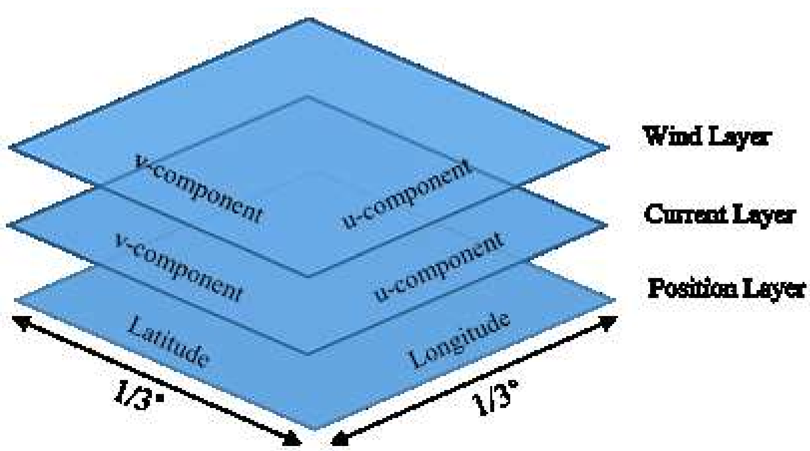

Figure 10. Visualization of Gridded Overlays Forming an Environmental Map.

The Matrix Mapper Method, its name derived from the method's reliance on stitching together multiple layers of matrixed data to form a unified map, works by iteratively selecting the adjacent position that best propagates the boat to the desired endpoint within reasonable consideration of the environmental factors in the immediate vicinity of the position. An example scenario is given below in Figure 11.

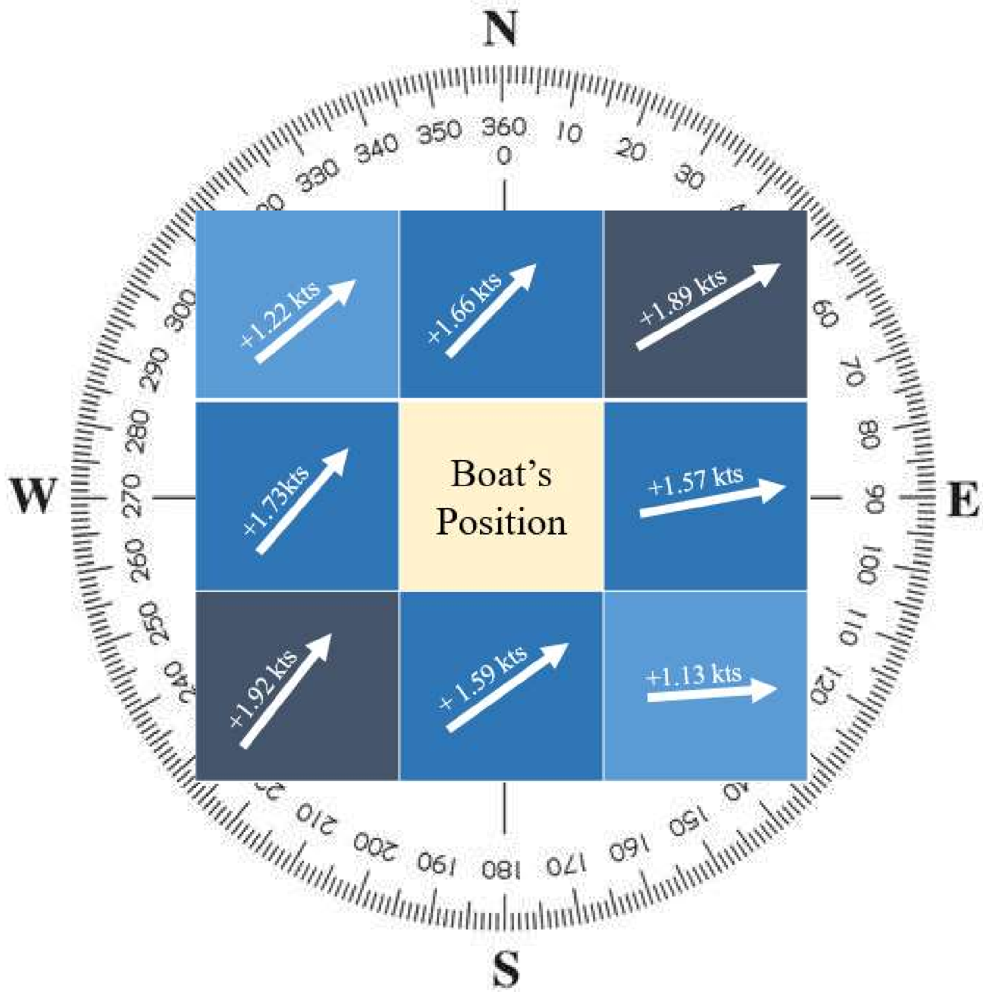

Figure 11. Visualization of Local Decision for a Single Iteration of Matrix Mapper Approach.

Because the search area only looks at the eight surrounding cells, known as the Moore Neighborhood [27], it most resembles a greedy best-first search decision method. The associated speeds are derived from the net effect of current and relative wind in each cell, which is 1/3° by 1/3°. Fundamentally, the Matrix Mapper Method chooses its next box by computing the one with the best balance between speed and desired direction of travel, maintaining positive headway from west to east. In the example in Figure 11, if the desired endpoint were at a bearing of 050° T, the method would select the box in the upper right corner, which would become the boat's new position before the process iterates.

Over the course of the voyage, the method faces dozens of decisions in which the net effect of wind and current oppose the boat in the general direction of desired travel (between 0° and 150° T). In these cases, the assumption is made that the boat can sail perpendicular to this combined force, thus rarely having to make headway into the opposing force. While the Sailbot can sail upwind through tacking and jibing, the assumption of perpendicular sail is conservative in nature to account for the disproportionately large effect that current has on a small sailing vessel as compared to a standard racing sailboat. A visualization of the eight possible movements is shown below in Figure 12. This decision complex is enacted in a slow-current situation (< 1.1 kts), usually when the Gulfstream begins to split into the North Atlantic and Canary Currents.

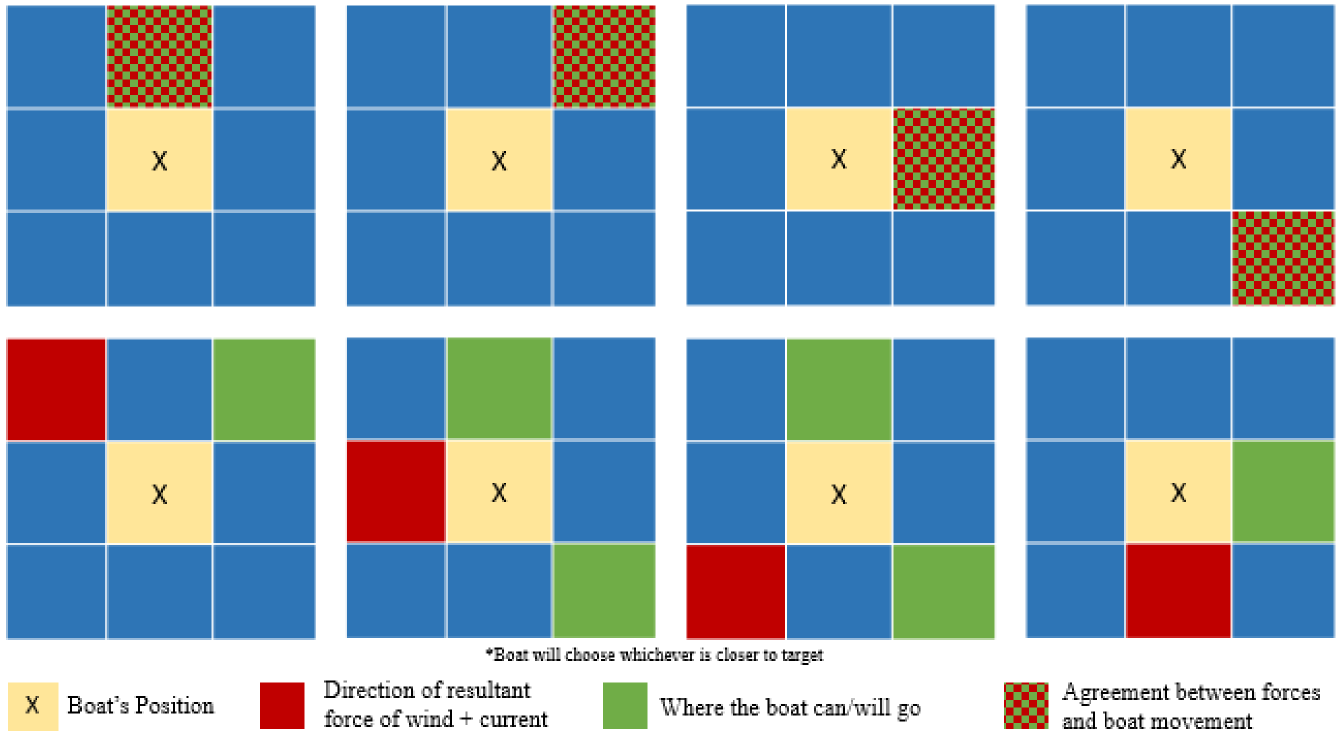

Figure 12. Decision Complex for Slow-Current Scenario.

Of note, the only instances where the boat has to sail at an acute angle into the wind and current are when the resulting force is pushing west or southwest. In these cases, survival of the boat depends on its ability to sail at a lower angle into the force. This assumption, along with the rest of the method, relies on two key thresholds that determine the balance between wind and current. The average wind speed in the Microtransat operating region across 609 datasets collected from early May through early September 2019 is 13.1 kts. While the Sailbot program has limited testing data available on Hull 14, previous groups have conducted on-water testing for similar MaxiMOOP hulls. Figure 13 below shows a polar diagram for a MaxiMOOP hull with comparable sail area and displacement as Hull 14.

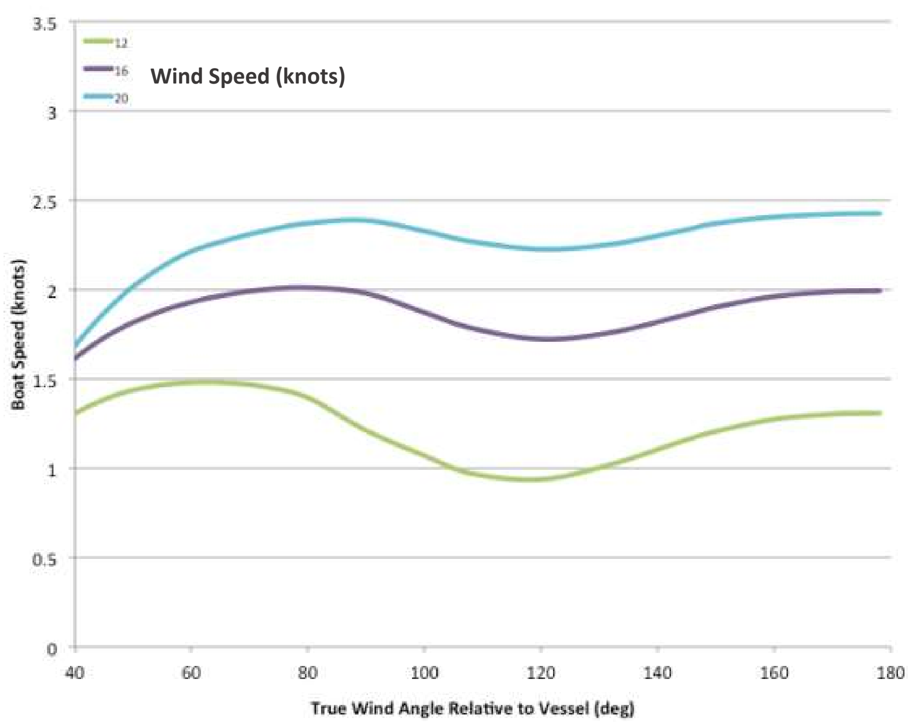

Figure 13. 2014 MaxiMOOP Performance Against Various Wind Speeds [28].

From this chart, two key thresholds are gleaned. First, assuming average wind conditions of 13 kts, the minimum boat speed is interpolated to be 1.1 knots, occurring when the relative wind angle is roughly 120°. This threshold is used in the analysis of the current field around the boat's position; when the average surrounding current strength is less than 1.1 kts, the boat can expect wind power to become the dominant force and capable of overcoming current, if necessary. This condition forms the basis of the decision complex shown above in Figure 12. Next, the maximum boat speed in average winds is interpolated to be 1.6 kts, occurring when the relative wind angle is about 60°. This threshold is used to determine the "sweeping current" condition, or when the average surrounding current is greater than the maximum achievable speed by wind. In this case, the boat can be expected to favor moving with the current. In the middle condition, where the combined magnitude of wind and current is between 1.1 and 1.6 kts and directionally positive, the boat chooses whichever box is closest to the desired endpoint. In most cases, the wind and current are not at odds with one another, and the boat is able to utilize the resultant force of the wind and current together to progress along its route. These thresholds act as important signals to the boat to use the environment to its advantage, a cornerstone of this proposed model. Figure 13 above is converted to a discretized polar diagram that applies a scale factor to the wind speed to return the resulting boat speed due to wind in the Matrix Mapper Method.

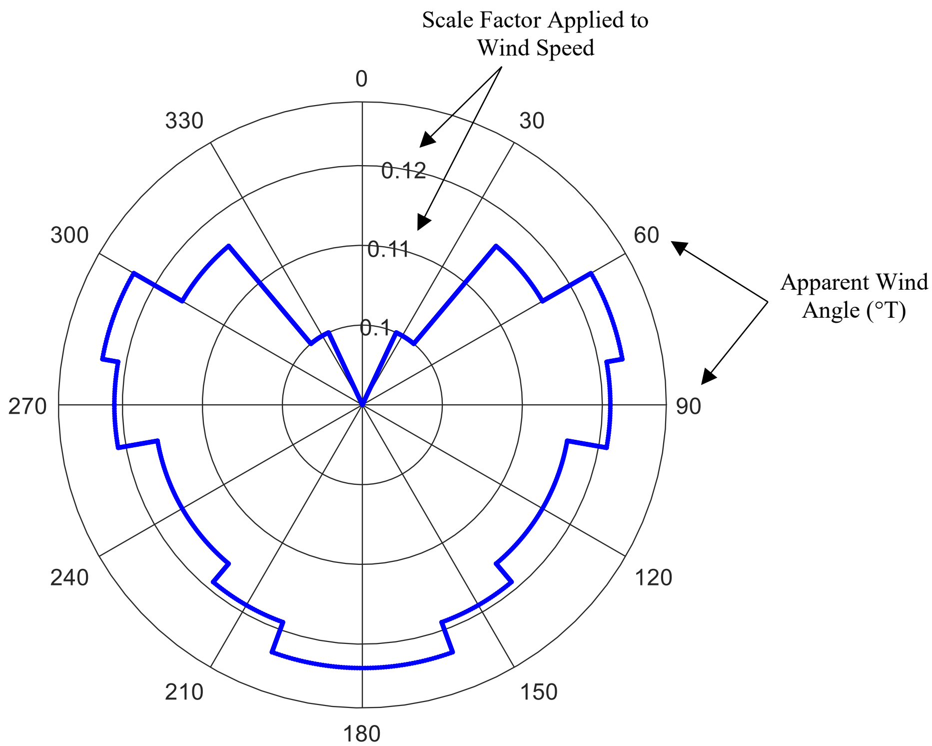

Figure 14. Discretized Polar Diagram for Use in Simulation.

To determine the boat speed in a particular wind field, the apparent wind angle is measured which corresponds to a scale factor on the blue outline. This scale factor is multipled by magnitude of the wind to determine the boat speed in the forward direction, or 0° relative. The output value associated with this polar diagram is combined with the current field using vector addition. The resultant vectors in each of the eight surrounding boxes then become inputs for the Matrix Mapper Method to decide the boat's next waypoint. It is important to note that the Sailbot has a wind sensor onboard that will return the apparent wind angle in real-time for the continuous, minute-to-minute sailing that is done throughout the voyage. However, in order to project out, forecasted data must be used and thus this model is sensitive to the accuracy of those forecasts.

C. Simulation Development

With the Matrix Mapper Method governing the behavior of the boat while ensuring progress along the Microtransat voyage, a simulation environment is constructed to test the method in various wind and current conditions while introducing uncertainty. Known as the Lead-Lag Environment, it hosts two variations of the Matrix Mapper Method moving through time given real data, simultaneously. One iteration is given wind and current data in "real time," or update period *n.* The route that the boat takes given this data is known as the "lead" route. The second iteration, the "lag" route, is given wind and current data one update period behind the lead, or $n - 1$. This mimics the gap between what the boat thinks is happening as it plans and projects out the next portion of its route (lag), and what is actually happening to the boat (lead). The update period for current data is available every 5 days. Wind data is available every 6 hours, but is utilized every ~12 hours in simulation to represent the optimistic transmission period that the Sailbot team expects to communicate with the boat via Iridium satellite.

The two iterations begin at the same point, and begin propagating in a manner similar to a greedy best-first search, with each movement constituting a new waypoint with a latitude/longitude coordinate. The distance between two waypoints is calculated, and the speed is assumed to be the magnitude of the combined wind and current vector. Dividing distance by speed yields the time, which is then used to iterate through the wind and current datasets. When the "time" reaches 12 hours, the wind datasets iterate, with the lag scenario trailing the lead by a 12-hour difference in data availability. When the "time" reaches 120 hours (5 days), the current map iterates, with the lag scenario trailing the lead by a 5-day difference in data availability. In nearly every case, the endpoints of the 5-day segment are at different coordinates, representing the actual position of the boat and where it projected it would end up. Inspection of simulations in the Lead-Lag Environment is useful to determine if the boat risks projecting out a fatal route while it is essentially operating blindly while waiting for the next iteration of datasets to come. Once the new current map arrives, the boat can reconcile the difference between its projected location and its actual location, and repeat this process out for another 5 days.

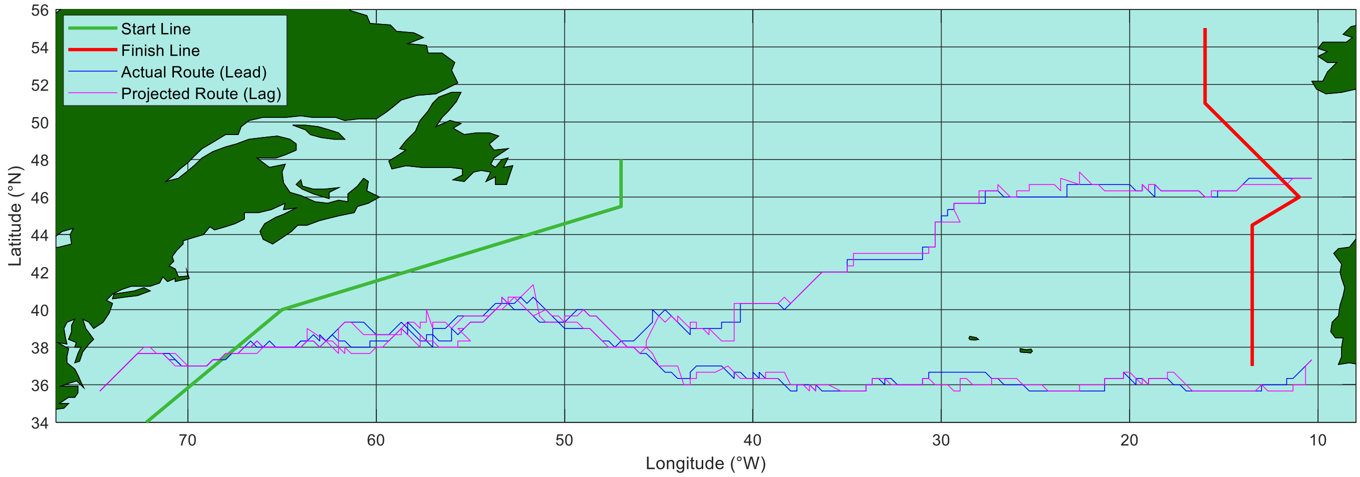

Figure 15. Two Simulations of Matrix Mapper Method in Lead-Lag Environment.

One of the Sailbot team's concerns was that with the Sailbot operating on non-real-time weather information, conditions could change rapidly enough to push the Sailbot fatally off track. Figure 15 shows that while the projected route varies from the actual, the boat running on outdated data still behaves very similarly to the boat operating with the most up-to-date data. This finding is important, as it shows that the update periods associated with the data are sufficient to sustain the boat during its journey. Figure 15 also displays a successful route, and an unsuccessful route that misses just south of the finish line. Because the Matrix Mapper Method considers the bearing and distance to target to determine its next move, this raises the question of whether or not there are more desirable target endpoints within the large finish line than others. In order to test this hypothesis, five target locations are chosen that span much of the latitude and longitude of the established finish line. Figure 16 below shows the location of these five target coordinates that will be tested in simulation.

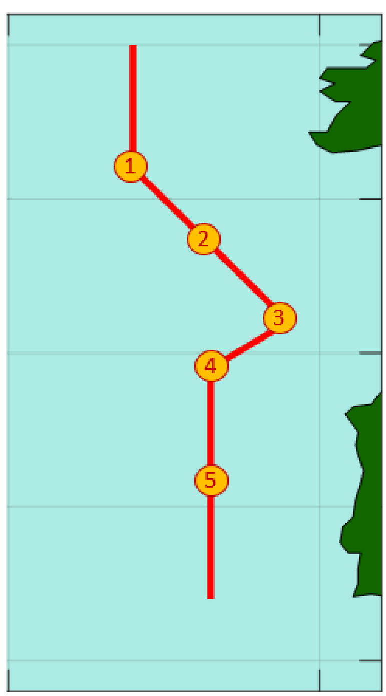

Figure 16. Five Target Coordinates to be Tested in Simulation.

Of note, the finish line spans 18 degrees of latitude, or over 1000 nautical miles. While the Sailbot team would be content with crossing the finish line at any point, it is in the interest of the team to gain understanding of the precision and accuracy associated with waypoint navigation for future competitions that may involve a much narrower finish line.

D. Property Measurement

The relevant properties to measure largely concern the simulation performance as a predictor of Microtransat success. The ultimate goal of the Sailbot Program is to successfully complete the Microtransat Challenge; if this cannot be done in simulation, then it certainly will not be accomplished in practice.

Table 1. Performance Evaluation Criteria.

Score	Property
0	< 50% of simulations result in Microtransat success
1	50-59.9% of simulations result in Microtransat success
2	60-69.9% simulations result in Microtransat success
3	70-79.9% of simulations result in Microtransat success
4	80-89.9% of simulations result in Microtransat success
5	≥ 90% of simulations result in Microtransat success

The logic behind Table 1 is simple, as the model is intended to project achievable routes based on realistic environmental inputs and behaviors. The more often the boat completes the Microtransat competition in simulation, the more confident the Sailbot team can be in its chances in the real voyage.

E. Schedule

Despite many challenges surrounding the learning environment, this project stayed sufficiently on course with a few modifications. From the timeline in the Appendix, the major objective for January included incorporating wind into the existing model created in the Fall semester, which was achieved albeit with a singular (not iterative) dataset. In February, the Sailbot team projected that MIDN Marino and Plzak would have Hull 14 on the water and ready for testing to create a polar diagram, but delays and the COVID-19 pandemic prevented further testing from taking place. During this phase, the thresholds for this project were developed and the simulation refined so that when speed polar data became available, it could easily be integrated into the program. Progress in March was slowed down after the COVID-19 crisis transitioned the Sailbot team to home learning environments. During this time, however, a deep literature review was conducted that related many concepts in this project to previous work in optimization and grid-based path planning. In April, after the cancellation of Capstone Day and the poster presentation, much effort was devoted to developing the simulation environment so that results could be gleaned from the project. Additionally, speed polar information was derived from previous literature written by a former Sailbot team, which is critical to this model. The program was consolidated and cleaned up into functions that will prove useful for future Sailbot teams who wish to implement this model or borrow pieces from it. Final deliverables were turned in on time after careful review.

E. Budget

Table 2. Budget.

LABOR	Category	Hours	hourly rate	Cost
	Midshipman	288	$25	$7,200
	Faculty	64	$60	$3,840
	Staff	16	$40	$640
Labor Sub-total				**$11,680**
OVERHEAD	**Category**	**Base Amount**	**Rate**	**Cost**
	Fringe Benefits	$11,680	35%	$4,088
	Facilities	$11,680	50%	$5,840
	General Services	$11,680	15%	$1,752
Overhead Sub-total				**$11,680**
MATERIALS	**Category**			**Cost**
	In-stock Items			$290
	New Items			$0.00
Materials Sub-total				**$290.00**
TOTAL COST				**$23,650**
OUT-OF-POCKET COST				**$0**

The in-stock items owned by the WRC Department are a Raspberry Pi 3 B+ and a RockBLOCK 9603 Iridium SatComm Module that will be used to implement this model onto the existing hull should it be successful. The Raspberry Pi will provide the processing power necessary to maintain a two-way communication link between the Iridium module outfitted on the hull and the global Iridium satellites that will access the off-hull server running this data processing program. The Raspberry Pi costs $40, and the Iridium module costs $250. This would be implemented by future group members should this project be pursued.

V. Results

In order to assess the model's performance over the long run under a variety of combinations of environmental conditions, ten thousand simulations are conducted using the Matrix Mapper Method in the Lead-Lag Environment. The model is given wind and current data from early May 2019 through early September 2019, and set to aim for the third target location at first because it is in the latitudinal center of the Microtransat finish line. Table 3 below shows the results of these simulations.

Table 3. Results of Simulations at Target Location Three.

Property		Value
Success	Success Rate (%)	90.48%
Failure	Boundary Failure Rate (%)	3.24%
	Finish Line Failure Rate (%)	6.28%
Average Time Length of Successful Voyage (days)		138
Average Speed of Successful Voyage (kts)		1.37
Average Distance Traveled during Successful Voyage (NM)		4495
Average Finish Distance from Target Coordinate (NM)		68
Average Current Map Update Period (days)		5.19
Average Wind Map Update Period (hours)		14

These results demonstrate that the simulation incorporates realistic navigation behavior with characteristics in the expected range of possibility. In order to put these metrics into perspective, a comparison is made to the *SB Met*'s successful Microtransat voyage. The *SB Met* actually sailed about 70% further than the charted, straight-line distance, with the difference caused largely by course changes natural to a sailing vessel subject to environmental conditions [29]. In comparison, the actual sailing distance of this particular simulation was about 55% more than the charted distance, and analysis of Target 2 reveals about 66% further. Additionally, the *SB Met*'s feat of 2750 NM in 79 days results in an average speed of 1.45 kts on a slightly larger boat than Hull 14, comparable to the simulations' average speed of 1.37 kts.

Failures are split into two categories to differentiate the relative "closeness to success." Boundary failures occur when the boat exits the specified operating region to the north or south (56°N and 35°N, respectively). A boundary failure is considered to be an irrecoverable failure. However, finish line failures are classified as simulations that stay within the northern and southern limits of the operating region, but do not cross the finish line. This means that they came within 60 NM of the northernmost point of the finish line, or within 120 NM of the southernmost point of the finish line. Under more favorable conditions, these routes may have very well ended in Microtransat success by coming as close as they did, and are therefore considered recoverable failures. The southern route on Figure 15 shows an example of a finish line failure.

Another set of ten thousand simulations are conducted at each of the five target locations to determine which, if any, yield a higher success rate and/or finish accuracy to the desired target location. The same data inputs are given as the simulation conducted above, with only the target location changing. Table 4 below shows the results of these simulations.

Table 4. Simulation Results at Each of the Five Target Locations.

	Target 1	**Target 2**	**Target 3**	**Target 4**	**Target 5**
Success Rate	83.27%	83.62%	90.48%	93.76%	94.19%
Boundary Failure Rate	2.17%	2.57%	3.24%	3.26%	4.33%
Finish Line Failure Rate	14.56%	13.81%	6.28%	2.98%	1.48%
Avg Time Length (days)	138	137	138	143	152
Avg Speed (kts)	1.36	1.40	1.37	1.34	1.26
Avg Distance Traveled (NM)	4471	4592	4495	4560	4594
Avg Finish from Target Location (NM)	334	117	68	26	75

This table underscores a level of consistency between simulations with regards to speed, distance traveled, and time length. Additionally, it reveals a pattern between target location, success rate, and failure types, highlighted below in Figure 17.

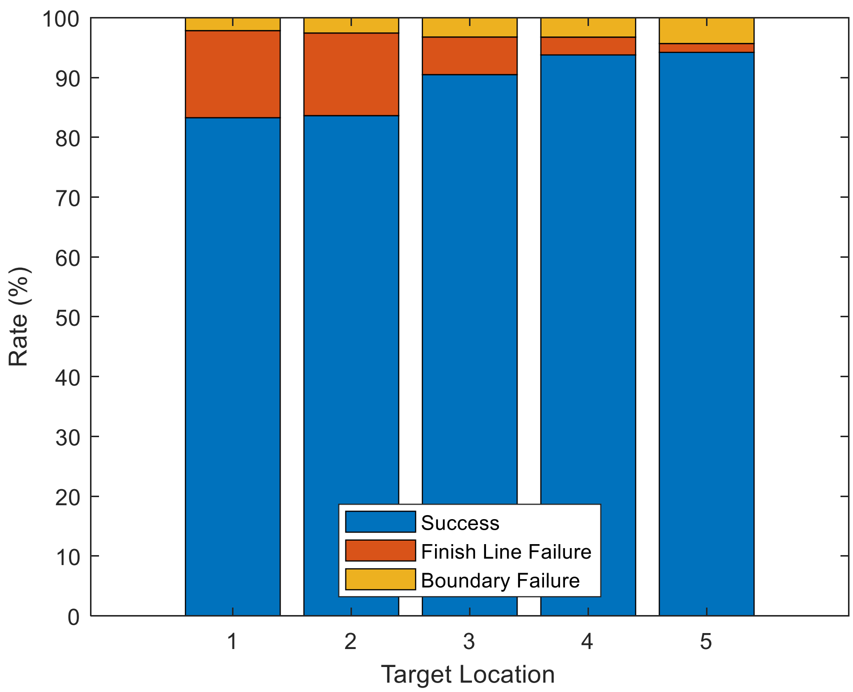

Figure 17. Graph of Success and Failure Rates at each Target Location.

Figure 17 shows that the success rate increases as the target location moves south, as does the boundary failure rate. However, the finish line failure rate decreases from north to south. A likely explanation for these trends lies in the wind and current patterns in the North Atlantic Ocean. Figure 18 below highlights the prevailing environmental force patterns in this region, with the Microtransat finish line added for context.

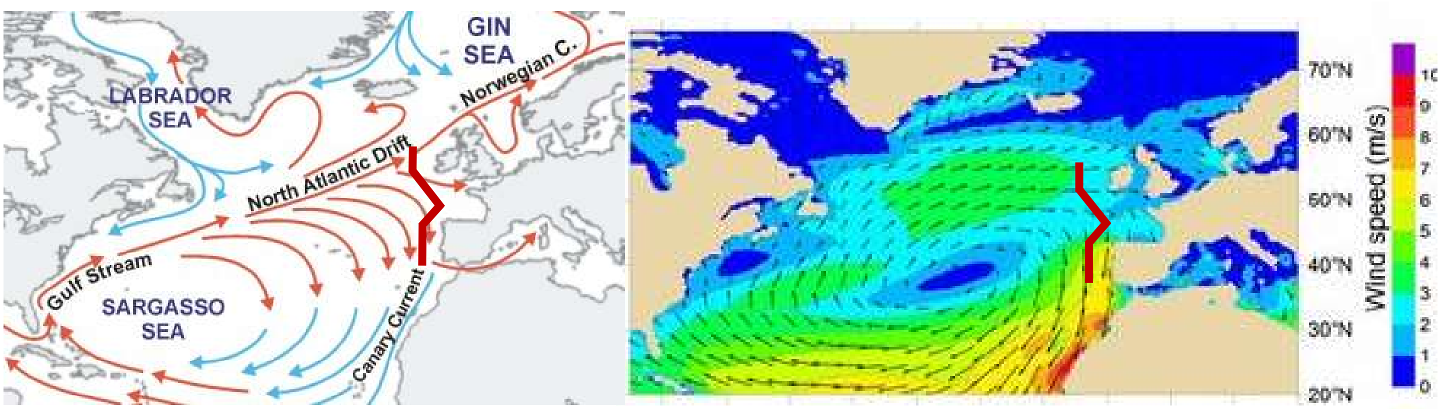

Figure 18. Prevailing Current (Left) and Summer Wind (Right) Patterns in the North Atlantic Ocean [30], [31].

These current patterns suggest that the tendency of the Gulfstream is to curl clockwise to the south once it begins to weaken. The wind patterns also curl to the south near the Microtransat finish line. The further east the boat moves, the more difficult it becomes to correct towards the north as the wind and current patterns shift at increasingly steeper angles towards the south. A key metric in Table 4 that contributes to this idea is that the finish distance from target generally increases as the target location moves north. Simulations aimed at northern targets tend to get pushed south as the boat has a hard time resisting the combined effect of southern winds and currents, whereas the boat has to “fight” less to get to southern targets and can control its finish more precisely. Figure 17 suggests the same, as the finish line failure rate also decreases as the target moves south. The boundary failure rate likely increases in this direction because there is less room for error when aiming south and prevailing forces are also pushing south; sometimes, the weather proves to be too overbearing.

As discussed, the Matrix Mapper Method exhibits “greedy” characteristics and makes decisions off of conditions in the immediate vicinity of the boat’s position. While this feature has its benefits in being able to closely follow flow fields, it is limited in its ability to look globally to make decisions that sacrifice short-term gain for long-term benefit. Because of this, route projections that are likely to lead to failure even just 200 NM out from the boat’s present position may still be selected because they are beneficial within the next 20-40 NM. The vast majority of all boundary and finish failures occurred to the south; because of the shortsighted nature of the Matrix Mapper Method, it does not recognize to compensate to the north early enough before the environmental forces begin to prevail and push the boat to the south.

VI. Conclusion

These results show that the Matrix Mapper Method in the Lead-Lag Environment is able to simulate a transatlantic voyage with realistic behavior. By modeling uncertainty and moving through time with actual wind and current data, the Sailbot team has reasonable confidence in the model’s projections of success and failure. By conducting thousands of simulations at various target locations, the Sailbot team can be more informed when constructing preplanned waypoints as an element of redundancy to this proposed model. With a weighted average success rate of 89.07%, this model scores a four out of five on the performance metric table shown in Table 1, suggesting a sufficient level of confidence in the boat’s success.

While this model is specifically designed for a west-to-east crossing of the Atlantic Ocean via sailpower, it can be applied to a variety of other scenarios after careful parameter tuning. Most importantly, threshold values will be specific to the individual boat and can be gleaned from on-water testing and formulation of a polar diagram. If the vessel at hand is not under sailpower, the program could be modified to disregard wind altogether and follow areas of greatest current strength in the direction of desired travel. The program created in MATLAB offers a scalable framework for unique datasets and decision making procedures to be implemented and tailored for individual purposes.

The end state of this project is to have a program running on an off-hull server that takes the boat’s GPS location, processes the best-available weather information into a set of waypoints that represent an achievable and satisficingly optimal route for the next ~5 days, and then repeats. Unlike preset waypoints that have been used in the past, these waypoints will dynamically adapt to the changing environment in an effort to use these conditions to the boat’s advantage. While this program generates waypoints every 1/3° latitude and longitude, it is able to be scaled back if desired. For example, only sending every third waypoint from the generated list will put roughly 1° between each, sending every sixth, 2°, and so on. This spacing may alleviate some of the challenges associated with waypoint navigation in the open ocean, where the threshold for how close the boat must get to a waypoint before shifting attention to the next one can afford to be looser. Additionally, in order to mitigate the lack of global knowledge that the greedy Matrix Mapper Method operates with, intermediate preplanned waypoints could be introduced to break up the voyage into sections. With this, the method may become less susceptible to falling into traps of local extrema. Further study must be conducted to determine proper coordinates for these intermediate waypoints, and how many there should be. Another way to maintain the benefits of a greedy search but project out further could be to expand the search neighborhood beyond the eight adjacent cells to the second or third ring beyond the immediate vicinity. Though it would add significant complexity, this development could provide early warning to the boat before it sails into irrecoverable regions. Together, these improvements could mitigate the risks associated with shortsighted decision-making on a long-distance voyage.

In the nearer term, this project will serve to further the efforts of the USNA Sailbot program towards completing the Microtransat Challenge. Currently, the control system is not conditioned to dynamically adapt the boat's route based on environmental factors, leaving this resource largely untapped and the route uninformed. With the simulation environment created in this project, the Sailbot team can continue to test a variety of hypotheses in order to maximize the likelihood of a successful voyage. If implemented successfully, this navigation model will be the first of its kind within the USNA Sailbot program.

References

[1] V. Turk, "Why We Need Autonomous Sailboats," [Online]. Available: https://motherboard.vice.com/en_us/article/4x34vd/why-we-need-autonomous-sailboats. [Accessed 19 March 2019]

[2] F. M. Lauro, S. J. Senstius, J. Cullen, R. Neches, R. M. Jensen, M. V. Brown, A. E. Darling, M. Givskov, D. Mcdougald, R. Hoeke, M. Ostrowski, G. K. Philip, I. T. Paulsen, and J. J. Grzymski, "The Common Oceanographer: Crowdsourcing the Collection of Oceanographic Data," PLoS Biology , vol. 12, no. 9, Sep. 2014.

[3] M. M. Verstraete, B. Pinty, and R. B. Myneni, "Potential and limitations of information extraction on the terrestrial biosphere from satellite remote sensing," *Remote Sensing of Environment*, vol. 58, no. 2, pp. 201–214, Nov. 1996.

[4] D. L. Rudnick, "Ocean Research Enabled by Underwater Gliders," *Annual Review of Marine Science*, vol. 8, no. 1, pp. 519–541, 2016.

[5] N. N. Soreide, C. E. Woody, and S. M. Holt, "Overview of ocean based buoys and drifters: present applications and future needs," in *MTS/IEEE Oceans 2001. An Ocean Odyssey. Conference Proceedings (IEEE Cat. No.01CH37295)*, 2001, vol. 4, pp. 2470–2472 vol.4.

[6] P. Miller, M. Hamlet, and J. Rossman, "Continuous Improvements to USNA SailBots for Inshore Racing and Offshore Voyaging," *Robotic Sailing 2012*, 2013.

[7] "The Microtransat Challenge," [Online]. Available: https://www.microtransat.org/rules.php. [Accessed 19 March 2019]

[8] K. Chan, "Robot boat sails into history by finishing Atlantic crossing," [Online]. Available: https://www.apnews.com/f6d0e2a099684468873ab48966590ada. [Accessed 19 March 2019]

[9] "The Microtransat Challenge," [Online]. Available: https://www.microtransat.org/tracking/index.php?nomarkers. [Accessed 19 March 2019]

[10] L. Parker, Class Lecture, Topic: "Navigation and Metric Path Planning." The University of Tennessee Knoxville, Feb. 2007.

[11] F. Tang, Class Lecture, Topic: "Metric Path Planning --Algorithms (Planners)." California Polytechnic State University, Pomona, CA, Feb. 2017.

[12] M. E. A. López, "Reactive Evolutionary Path Planning For Autonomous Surface Vehicles in Lake Environments," p. 18.

[13] C. Taswell, "Satisficing search algorithms for selecting near-best bases in adaptive tree-structured wavelet transforms," IEEE Trans. Signal Process., vol. 44, no. 10, pp. 2423–2438, Oct. 1996, doi: 10.1109/78.539028.

[14] A. Gerdelan, Class Lecture, Topic: "Finding the Shortest Path," Trinity College, Dublin, Ireland, pp. 6-19, 2016.

[15] MIT OpenCourseWare, Topic: "Principles of Optimal Control." Massachusetts Institute of Technology, Cambridge, MA, June 18, 2008.

[16] D. Fan, P. Shi, "Improvement of Dijkstra's algorithm and its application in route planning," *2010 Seventh International Conference on Fuzzy Systems and Knowledge Discovery,* Aug. 2010.

[17] B. Ghai and A. Shukla, "Wave Front Method Based Path Planning Algorithm for Mobile Robots," in Proceedings of First International Conference on Information and Communication Technology for Intelligent Systems: Volume 2, Cham, 2016, pp. 279–286, doi: 10.1007/978-3-319-30927-9_28.

[18] D. Sislák, D., P. Volf, and M. Pechoucek, "Accelerated A * Trajectory Planning : Grid-based Path Planning Comparison," *AAMAS 2009*, pp. 74-75.

[19] M. Du and M. Kang, "Study of Long-term Route Planning for Autonomous Sailboat," ROBOTIC SAILING, p. 8, 2018.

[20] M. Heusner, T. Keller, and M. Helmert, "Understanding the Search Behaviour of Greedy Best-First Search," p. 48.

[21] M. Życzkowski, "Method of Routing Ships Sailing in Dedicated Environment," Annual of Navigation, vol. 24, no. 1, pp. 147–158, Dec. 2017, doi: 10.1515/aon-2017-0011.

[22] National Oceanic and Atmospheric Administration (NOAA), "How fast is the Gulf Stream?" https://oceanservice.noaa.gov/facts/gulfstreamspeed.html (accessed Apr. 28, 2020).

[23] "Gulf Stream | ocean current," Encyclopedia Britannica. https://www.britannica.com/place/Gulf-Stream (accessed Apr. 28, 2020).

[24] J. Gyory, A. Mariano, and E. Ryan, "The Gulf Stream," Ocean Surface Currents. https://oceancurrents.rsmas.miami.edu/atlantic/gulf-stream.html (accessed Apr. 28, 2020).

[25] Environmental Data, Monitoring, and Prediction System (EDMAPS), USNA Oceanography Department, http://edmaps.usna.edu:8080/.

[26] NOAA National Center for Environmental Prediction, "NCEP/NCAR Reanalysis 1: NOAA Physical Sciences Laboratory." https://psl.noaa.gov/data/gridded/data.ncep.reanalysis.html (accessed Apr. 28, 2020).

[27] P. Varshavskaya, "Comp150-07: Intelligent Robotics: Wavefront Planner," Intelligent Robotics. https://www.cs.tufts.edu/comp/150IR/labs/wavefront.html (accessed Apr. 28, 2020).

[28] P. Miller, M. Hamlet, and C. Sauze, "MaxiMOOP: A Multi-Role, Low Cost and Small Sailing Robot Platform," *Robotic Sailing 2014*, p. 9.

[29] P. Ridden, "Unmanned surface vessel successfully crosses Atlantic," New Atlas, Sep. 04, 2018. https://newatlas.com/offshore-sensing-sailbuoy-met-atlantic/56204/ (accessed Apr. 28, 2020).

[30] "The Gulf Stream current system." https://courses.seas.harvard.edu/climate/eli/Courses/misc/201209-Mallorca/Sources/2-THC/from-EPS131/2-RAPID/gulf_stream.php.html (accessed May 03, 2020).

[31] E. Charles, D. Idier, P. Delecluse, M. Deque, and G. Le Cozannet, "Climate change impact on waves in the Bay of Biscay, France," *ResearchGate*. https://www.researchgate.net/publication/236170017_Climate_change_impact_on_waves_in_the_Bay_of_Biscay_France (accessed May 01, 2020).

APPENDIX: TIMELINE

EW402 - 4 Credits - Spring AY2020

Week (M)		Note	Planned Activities	Actual Hrs
1	Jan 6	Begins Tues (M sched)	• Experiment with adding more intuitive constraints to Matrix Mapper Method with current	6
2	J 13		• Develop comprehensive way to incorporate wind into the picture	6
3	J 20	M= MLK	• Continue to refine wind inclusion, may have to interpolate between 1 degree resolution	5
4	J 27		• Clean up Matrix Mapper script to become modular, using functions	6
5	Feb 3		• Ensure functionality of Matrix Mapper (debugging) • Generate plots, documentation	6
6	F 10	Ac Reserve	My 6 week deliverables are: • Updated Final Paper that includes work up to the 6-Week point	4
7	F 17	M = Pres Day 6 wk grds W	• MIDN Marino and Plzak have projected that by this point, they will have Hull 14 on the water for testing • I will collect data from on-water testing to create a better speed polar diagram.	5
8	F 24		• Continue testing to refine speed polar diagram; if it is not accurate, the model is borderline useless	6
9	M 2	M = Columbus	• Summarize progress by updating Final Paper and methods used to collect data	4
SB	M 9			
10	M 16		• Generate plots based on newly collected data from testing	6
11	M 23		• Tweak Matrix Mapper Method to reflect additional nuances and assumptions with newfound information	6
12	M 30	Ac Reserve	My 12 week deliverables are: • Final Paper draft highlighting progress and work to the 12-Week point • Discussion points of what I want to include in poster	2
13	Apr 6	12 wk Grades T	• Share draft poster with adviser for comments. Use template. • Add their input	8
14	A 13		• Submit poster to MSC for printing.	8
15	A 20		• Share draft report with adviser for comments. Use template. • Capstone day	8
16	A 27	T= last day of class	• Schedule technology transfer with adviser • Turn in report	8