Long-Range Near-Optimal Path Planning for Gliders in Complex High-Energy Environments

Donald A. Sofge & Julian S. Whitman
Naval Research Laboratory
Washington, DC USA
Email: donald.sofgeATnrl.navy.mil

***Abstract*—Autonomous gliders may be instrumented with sensors and used to gather data and perform surveillance in strategically critical areas, but their use is currently limited to low-energy environments due to the challenges of path planning and navigation in high-energy environments, including variable environments (e.g., eddies and currents resulting from weather and seasonal effects), and particularly where ocean currents exceed glider speeds. In this effort we develop a navigation planner for gliders in complex high-energy ocean environments to achieve point-to-point navigation, and explore the use of this planner for optimizing a team of underwater gliders. These techniques will extend the capabilities of teams of autonomous vehicles to perform missions in strategically important areas.**

I. Introduction

Autonomous systems offer tremendous potential for surveillance and other missions in the littoral zone. However, coordinated route planning and navigation for autonomous underwater vehicles operating in a complex, dynamic, partially unknown undersea environment is a challenging problem. In this effort we address planning and navigation of autonomous vehicles (such as gliders) operating in dynamic and/or unknown environments, and explore the optimization of a team of underwater gliders to improve accuracy of assimilative ocean prediction models for undersea warfare.

Gliders may perform long-term sensing and data collection due to their low energy expenditures, since they use only gravity for propulsion. On-board computation requires little power, and the gliders may be equipped with solar panels to recharge their batteries while on the surface. However, as highly under-actuated systems (only buoyancy control for diving, and control surfaces for turning, are used) they also present a challenge for route planning and navigation. Glider navigation and control solutions developed to date generally assume that no ocean currents are present, or that their effects are negligible and can be treated as error effects. As such, gliders are typically used only in ocean environments where the ocean current speeds are substantially less than glider surface speeds (i.e. speed of the glider projected onto the ocean surface). In such placid environments gliders often follow a radiator pattern for data collection, or simple waypoint following for a pre-planned paths is used. However, many Navy operations require using gliders in much more dynamic environments where currents may exceed glider surface speeds. To accomplish this we need a planner that will allow gliders to utilize localized ocean currents to move from one position to another, just as a sailor on a sailboat utilizes winds to maneuver a sailboat.

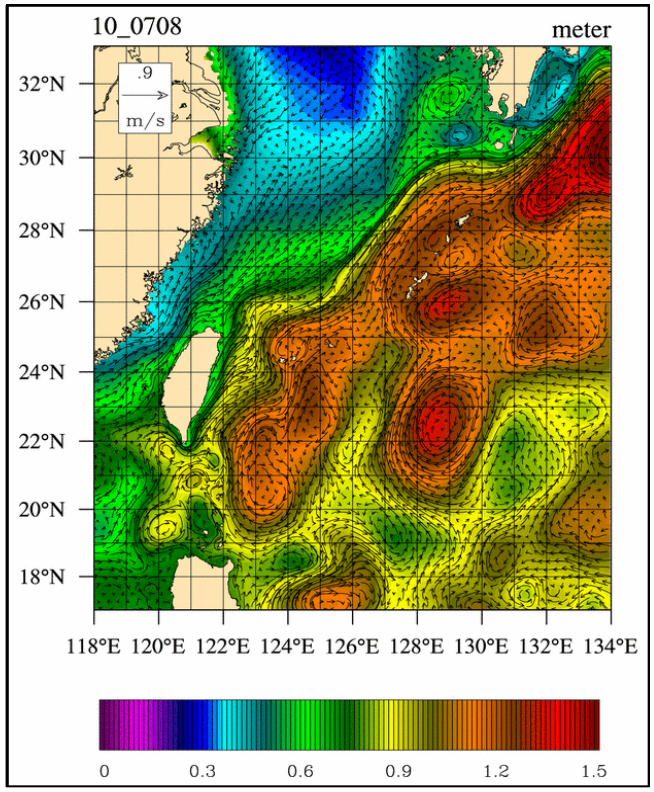

Figure 1. Complex high-energy ocean environment

Simple glider navigation systems often rely upon the use of dead reckoning to navigate the glider along a predetermined set of waypoints. We started by solving the 2D path planning and navigation problem for long-term autonomous sailboat navigation in a dynamic environment [1], and then adapted the planner for use with an unmanned glider.

II. Approach

The autonomous glider navigation problem is to find a near-optimal traversable path from one position (given by latitude and longitude coordinates) to another. The navigation problem can be solved by combining a short-term iterative algorithm with a long-term iterative algorithm. The long-term algorithm provides an estimate of the optimal path to guide the short-term algorithm. The long-term algorithm uses discretized dynamic programming [2] to quickly find a near-optimal path, avoiding land masses and other undesirable locations. For the purposes of simulation the path was optimized for time.

A few simplifying assumptions were made. It is assumed that at launch time the vehicle will have access to an ocean model (including current estimates) for the region it will travel through, and that any shallow or inaccessible areas will be marked appropriately in the model. It is highly probable that there will be some noise in this data, so it is used only for obtaining a near-optimal long-term path. The short-term algorithm does not assume that the initial data is correct. The short-term algorithm is designed to use data about the immediate surroundings in real time to reach the next waypoint quickly and to avoid running aground.

For this data to be useful for planning and navigation, a vehicle model must be employed. The algorithms used assume that the vehicle's capabilities (e.g. maximum forward speed, glide angle, turning radius) are specified, and in this effort an underwater glider simulation model was used.

The glider simulator route planner combines a long-term planner with a short term iterative path optimizer to generate a time-optimal path through ocean currents. This route planner attempts to find an optimal path from an initial coordinate position to another. Gliders are usually used in areas where the ocean currents are negligible or ignored as error, but in reality the currents can greatly affect glider path. The purpose of this planner is to create a path that utilizes or avoids ocean currents. Simple glider navigation often uses dead reckoning navigation. However, this is often not the most efficient route (especially for time-optimization), and in stronger currents (where current speeds exceed maximum glider speed), it may not even be possible for the glider to reach the goal using dead reckoning from the chosen starting point.

III. Algorithm

The path planning algorithm consists of two parts: a long-term planner and a short-term planner.

A. Long-Term Planner

The user selects a Navy Coastal Ocean Model (NCOM) data file, and uses a program to extract needed current data into a more easily accessible data form. The user enters start and end coordinates, and vessel information, such as speed and maximum depth. The program takes the start and end points and creates a grid, with each cell in the grid corresponding to a data point from the selected NCOM file. The current grid is created by averaging the currents at each depth less than the pre-specified maximum glider depth. After initializing variables, the program starts the long-term planner. The program constructs a cost-to-go grid, where areas with water shallower than the glider's deepest depth are assigned a cost of 1,000,000.

The long term planner starts at the start coordinate cell, and checks each cell around it at 45 degree intervals. It chooses the cell that has a lower cost-to-go value but also brings it closer to the goal, adds that to a path waypoint list, and repeats for the new cell. When it reaches the end point's cell, it removes waypoints that do not represent a change in direction.

Each cell in the Cost-To-Go (CTG) grid corresponds to exactly one cell in the grid of current vectors. The goal is assigned a cost of zero and then costs are assigned to the goal cell's children. A cell's children are the eight cells which surround it in 45 degree increments. A child's cost is calculated as the time it would take to travel a linear path from the child cell to the parent cell plus the cost of its parent cell. The resulting total cost shown in the grid is the time it would take to travel from the cell to the goal.

The algorithm continues to recursively assign costs to the rest of the grid. If a cell is revisited with a lower cost, the lower cost overwrites the previous cost unless it is has a cost of 1,000,000 (which is used to indicate an impassable area).

The lowest total cost path from the start to end is constructed by starting at the cell containing the start point, and selecting the lowest of the eight adjacent CTG cells. The next waypoint selected is the center of the selected cell. This process is repeated until the goal point is reached.

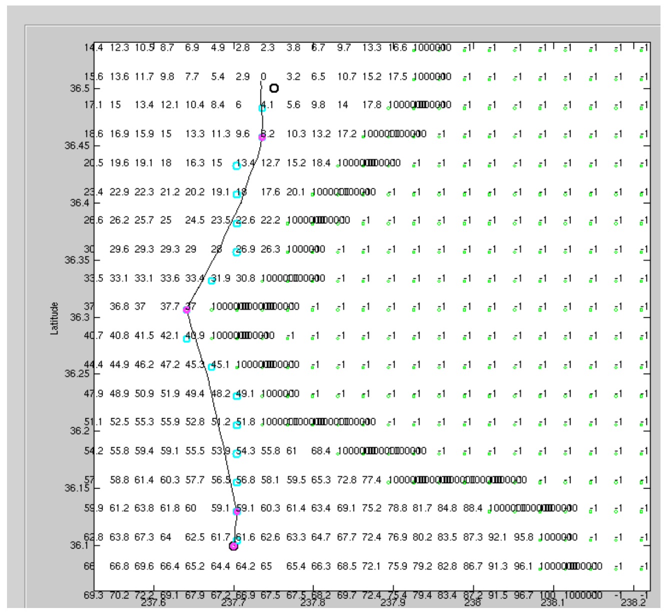

Figure 1. Cost-To-Go Grid Data

In the CTG grid shown in Figure 1, the edges of impassible areas are assigned 1000000, inaccessible areas are left with -1, and other cells are assigned numbers based on the time it would take to get from them to the goal. Figure 2 shows a screenshot of the glider simulation GUI using this CTG grid.

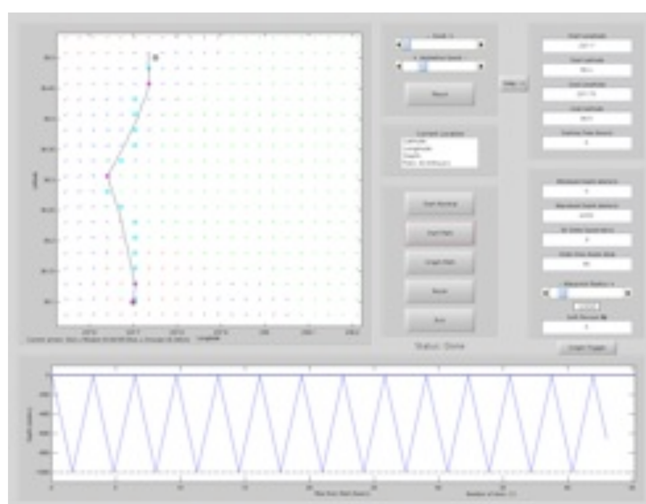

Figure 2. Screenshot of Glider Simulation GUI

B. Short-Term Planner

The short-term planner follows an algorithm that picks the heading that brings it closest to the next waypoint from the long-term planner, also taking the current speed and direction into account. Multiple iterations remove unnecessary turns and waypoints, and cut corners generated by the long-term planner.

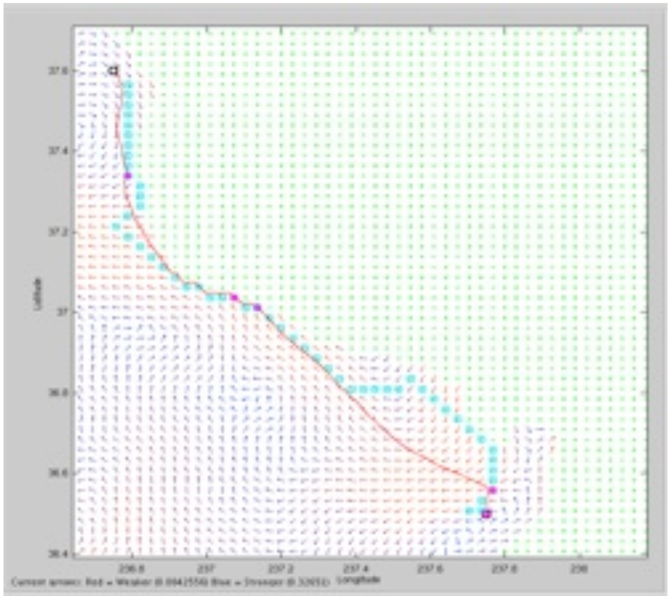

Figure 3. Path optimized for both currents and impassible areas (coastline). Data is from the Monterey Bay area.

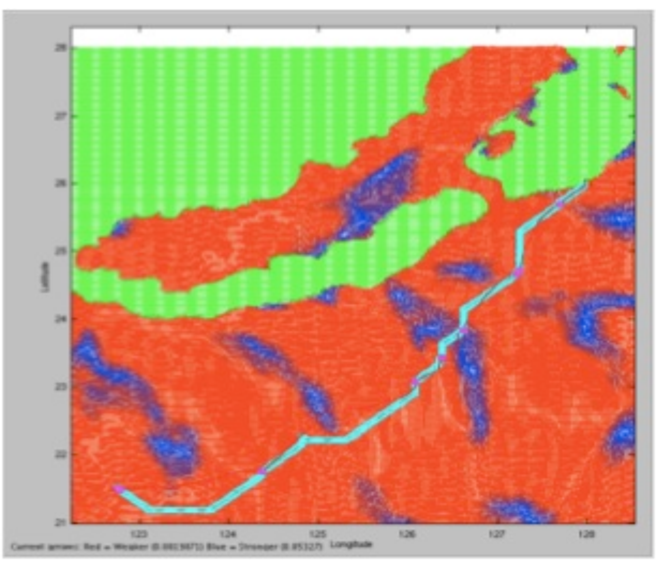

Figure 4. Long-distance glider route in high-energy environment. Data is from the Okinawa Trough area.

Figure 4 shows that the route planner is capable of finding very long routes even in high-energy ocean environments such as the Okinawa Trough area.

In situations where it is feasible for the glider to follow a straight-line path to the goal, if there are currents present, even small ones, it may be able to reach the goal sooner (and conserve energy) by using a current-optimized path. Figure 5 shows an example of this for a path in the Monterey Bay area.

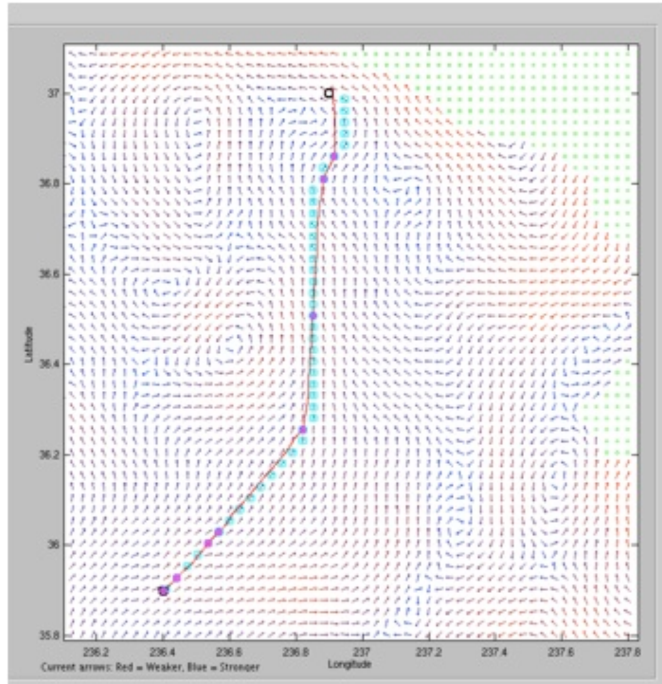

Figure 5. Path optimized to take advantage of small currents. Data is from the Okinawa Trough area.

IV. Analysis

A. Optimized Routes vs. Dead Reckoning

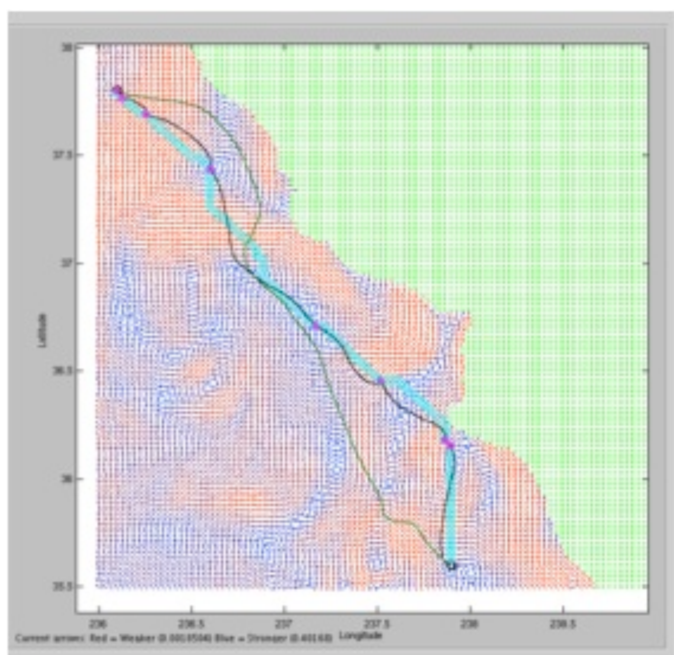

Figure 6. Optimized route planning vs. dead reckoning. Data is from the Monterey Bay area.

Since most point-to-point navigation using gliders uses dead reckoning, our analysis of the performance of our planner focused on comparison of the paths produced, and more importantly the transit time from start to goal, using the two methods.

In the Monterey Bay map shown in Figure 6 the glider must navigate from the starting point (shown near the top) to the finish (near the bottom). The dead reckoning path (shown in dark green) took 477.69 hours, while the optimized path (shown in black) took only 385.71 hours.

The navigation optimizer may be most useful in situations where there is no dead-reckoning navigation solution from start to finish. Figure 7 shows an example of this using data from the high-energy Okinawa Trough area.

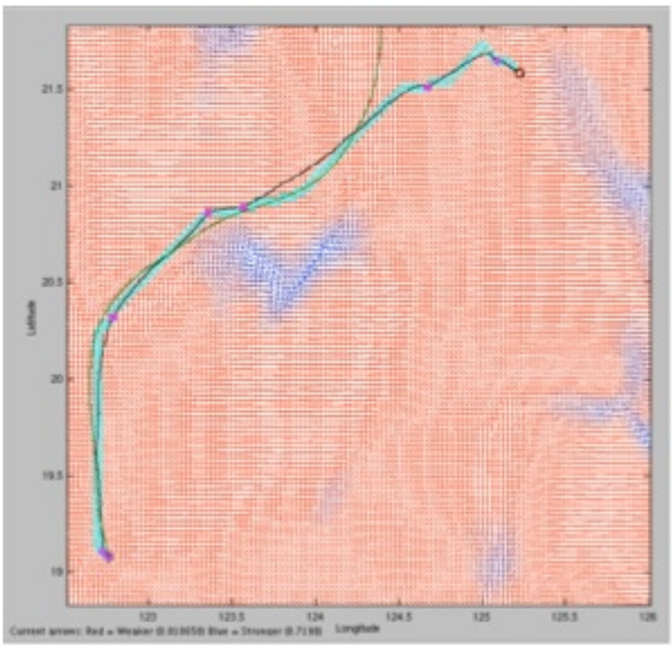

Figure 7. Optimized Route (black) vs. Failed Dead Reckoning Path (dark green)

In Figure 7 the glider starts at the pink dot in the lower left and must navigate to the finish near the upper right. The dead reckoning path (dark green) is unable to compensate for the strong currents of the Okinawa Trough area in time to reach the target, while the navigation planner is able to reach the target.

V. Summary & Conclusions

Glider navigation and control solutions developed to date generally assume that no ocean currents are present, or that their effects are negligible and can be treated as error effects. Simple glider navigation systems often rely upon the use of dead-reckoning to navigate the glider along a predetermined set of waypoints. The autonomous vehicle navigation problem is to find a near-optimal traversable path from one position (given by latitude and longitude coordinates) to another. We solve the navigation problem by combining a short-term iterative algorithm (locally navigating using on-board sensors) with a long-term iterative algorithm (using ocean model data). The long-term algorithm provides an estimate of the optimal path to guide the short-term algorithm. The long-term algorithm uses discretized dynamic programming to quickly find a near-optimal path. The navigation planner has been tested using a glider simulation testbed and various simulated operating environments specified by a Navy Costal Ocean Models (NCOM).

Acknowledgements

This effort was supported by the Naval Research Laboratory under Work Order N0001409WX30016. Thanks to Charlie Barron and his team at NRL for the ideas and NCOM models, and to Brian Bourgeois and his group for providing the glider vehicle simulation code. Thanks also to Max Harper and Rajit Kumar who contributed to an earlier stage of this project under the NREIP internship program.

References

[1] Sofge, D. and M. Harper, "Autonomous Route Planning and Navigation for UxV Teams," *Workshop on Machine Intelligence for Autonomous Operations*, Lerici Italy, October 2009.

[2] White, D. and D. Sofge, Handbook of intelligent control: neural, fuzzy, and adaptive approaches (chapter 12, Van Nostrand Reinhold, 1992).