SENSOR AND BEHAVIOR FRAMEWORKS FOR IMPLEMENTING BACKSEAT DRIVER ON A SLOCUM GLIDER

By

SOPHIE LOUISE VASCONCELLOS SCOPAZZI

A thesis submitted to the

School of Graduate Studies

Rutgers, The State University of New Jersey

In partial fulfillment of the requirements

For the degree of

Master of Science

Graduate Program in Oceanography

Written under the direction of

Travis Miles

And approved by

New Brunswick, New Jersey

October 2024

ABSTRACT OF THE THESIS

Sensor and Behavior Frameworks for Implementing Backseat Driver on a Slocum Glider

By SOPHIE LOUISE VASCONCELLOS SCOPAZZI

Thesis Director:

Travis Miles

The Slocum glider's long-duration and cost-effective oceanographic data collection capability, coupled with remote operation from shore, makes them ideal platforms for oceanographic research. However, once submerged in standard operation, instrument control, waypoints, behaviors, and more are set until the vehicle resurfaces. This lack of underwater adaptive control can lead to limitations in scientific sampling, limiting the glider's ability to meet the mission's scientific goals. Such goals could be oil spill detection or processing large data files from onboard sensors. Both require either optimizing the autonomous underwater vehicle's (AUVs) flight path intelligently and adaptively, which allows more data collection in scientifically rich areas, or an increase in onboard computing power for an increase in data processing capability.

Due to the need for enhanced control capability while underwater, the Slocum glider Backseat Driver (BSD) architecture has been in development for over ten years, but its utilization has been limited by the availability of resources and expertise on the subject, resulting in limited end-user implementation.

In this work, I present a manual for using the Slocum BSD architecture on a G3S Slocum glider. It contains examples for setting up an external controller (a Raspberry Pi 4), managing file transfers, incorporating real-time sensor data for decision-making, and advancing the ocean simulation capability of a glider simulator using a Slocum Fleet Mission Control (SFMC) script. I also demonstrate two Slocum BSD architecture test cases to autonomously navigate with a ninety-degree offset from the depth-averaged current (DAC), then on a heading calculated to minimize deflection distance while crossing a large-scale current.

Four simulations were run, two with the standard industry practice of using a distant waypoint to control the heading when crossing large-scale fast-moving currents, and two with the Slocum BSD architecture use-case examples. The missions utilizing the BSD architecture updated the heading less frequently and did so with a more optimal average heading. All examples listed previously, and more, are explained in far greater depth in the accompanying manual titled: *Implementing the Backseat Driver Architecture on a G3S Slocum Glider: A Manual for Operators.*

Through using the Slocum BSD architecture, we can fully realize the glider's true capability for long-duration and cost-effective persistent oceanographic sampling presence in challenging regions throughout the globe. This research and accompanying manual provide a starting point for glider pilots, technicians, scientists, and others when working through how to use the Slocum glider's BSD architecture. The ability to use this incredibly powerful way to enhance the Slocum glider's capability and data collection is only limited by our imagination and ingenuity.

ACKNOWLEDGEMENTS

Throughout my life, I've been blessed not only by my family but also by incredible teachers and mentors. I am ever thankful for both my parents, who taught me the importance of showing up and putting in the work. My Grammy, Jude Vasconcellos, for challenging my artistic creativity in all my endeavors. My best friend in the whole word, Jasper Scozzafava, for being a constant companion through our sometimes tumultuous life, and to their parents, Ingrid and Gary, who have been my second parents for as long as I can remember. Marine O'Neill, thank you for all the long involved talks about life. You who has been an ever-present source of inspiration in my life.

There are far too many mentors and teachers to list, so to name but a few, I first thank Scott Sellers for teaching me a just and honorable way of living. Michael "Doc" Wing for his dedication to fostering curiosity and discovery in students starting their high school journey. Lenny Gucciardi for hours upon hours of discussions about the natural world around us as we hiked throughout California (and for making me go that one time). To more hikes through the mountains, deserts, and coasts of our beautiful world.

From undergrad, Captain Tamara Burback has forever shaped not only who I am, but also how I spend my activist energy and my career. I am now who I have always been; for that, I thank you. Dr. Alex Parker has created the Oceanography Department where nothing was before. Dr. Alejandro Cifuentes-Lorenzen who asked if I wanted to take a class about making a CTD (my path down all of this is really his fault). Dr. Julianne Chisholm for telling me to apply to the Marine Technology Society EMERGE Program (and so much more poetry this sentence doesn't do justice too), where I met Dr. Josh Kohut and Jackie Veatch and first heard about the Masters of Operational Oceanography program. I thank all of you. And to all of those not mentioned by

name (hi Garret Dasigan and Leon Adams!): I could not have done it without you. The words are never enough. I cannot wait till I can swing by your office again or see you again.

From the first day I arrive in New Jersey, before the school year had even begun, I was welcomed into the Rutgers family at the Department of Marine and Coastal Sciences. It is thanks to Jackie, and all the other OGSA students, that what could have been a scary first few months were instead full of new friends and experiences. I instantly felt welcomed and like I had a new home; for that, I thank you all.

This work would not have been possible without the technical support from many individuals. To Joe Gradone, thank you for always welcoming my interruptions in your office and for the hours you helped theorize, troubleshoot, plan, and edit, making this endeavor possible (and Leah Hopson for the candy!). Without your help this would not have been nearly as successful. Dave Aragon, Salvator Fricano, and to everyone in my cohort, thank you for all the conversations and questions about my work, pushing me towards ever better explanations. Ella Wawrzynek, for asking me many questions, greatly helping the creation of my manual, and Zach Silva for answering questions of my own, I thank you both. Cassidy Gonzalez, for providing immense invaluable feedback towards improving the manual. To Alex Lopez, for putting in the work, leading this program towards ever greater depths of learning. And finally to my advisor Travis Miles. Thank you for making sure I never lost sight of the forest that lies beyond the day-to-day weeds.

I could not have asked for a better place to spend a year of my life. Together, we have worked hard, had immeasurable fun, and we continue to change the world.

INTRODUCTION

Slocum gliders are well suited to sample the world's oceans. They are highly cost-effective and capable of multi-month missions gathering thousands of vertical casts of oceanographic data [1]. Slocum gliders' capabilities coupled with their remote programmability makes them ideal platforms for oceanographic research. However, like any system purely reliant on onboard power, operators of Slocum gliders must make energy consumption decisions causing tradeoffs between scientific sampling, communication with shore for real-time data telemetry, and flight trajectory. The interplay between these variables must be accounted for by glider pilots as they ensure ample data collection for a successful scientific mission.

During an ocean deployment, the standard operating procedure is to have the glider communicate with the shore via the Iridium satellite network. This method is bandwidth limited and connections are lost as the glider is submerged. Therefore, pilots ashore cannot make decisions for an underwater glider in real-time and mission parameters about sampling and waypoints are only configurable during surface events. While Slocum gliders have enabled a persistent oceanographic sampling presence in challenging regions ranging from polar oceans [2], beneath tropical cyclones [3], [4], and across ocean basins [5], the lack of adaptive control between vehicle surfacings can limit scientific sampling.

The implementation of a BSD architecture onboard the autonomous underwater vehicle (AUV) can help meet this gap in capability. A system with a BSD can help answer questions ranging from detecting and mapping the extent of oil spills [6], [7], mapping the underwater portion of icebergs [8], following thermoclines [9], coastal upwelling fronts [10], haloclines [11], processing ADCP [12] or side-scan data [13], and optimizing the flight of the glider as it navigates ocean currents, as described in this work. Each of these examples requires optimizing the AUV’s flight path intelligently and adaptively, which allows more data collection in scientifically rich areas. To address this desired increase in vehicle capability, the Slocum glider BSD architecture has been in development for over ten years, with significant development efforts in recent years. Its use has been limited by availability of resources and expertise on the subject, resulting in limited end-user implementation.

METHODS: REQUIRED HARDWARE

There are three main components to this work: a G3S Slocum glider or a shoebox glider simulator, an external controller, and the hardware and software connection between the two. A fourth component, a Slocum Fleet Mission Control script was made to verify proper external controller operation.

Slocum G3S Glider Simulator

Termed the “shoebox” due to its oblong rectangular shape, Teledyne Marine makes a dedicated G3S glider simulator. It has a flight computer and science computer with all standard science ports allowing for connection of sensors in the typical manner.

The external controller must exist within the size and energy consumption constraints that come from being inside a Slocum glider on long-duration missions. Therefore, the external controller should be an energy-efficient small form factor single-board computer. This has led to the adoption of common external controllers such as a BeagleBone Black [14] or a Raspberry Pi 4 [12]. For this work, the external controller selected was a Raspberry Pi 4 Model B due to accessibility of resources and ease of beginner use, making it well suited for educational purposes. Additionally, its Wi-Fi capability is not only useful for the lab setting, allowing remote access through the secure shell protocol, but its ability to generate its own Wi-Fi connection means it is also possible to connect to the Raspberry Pi while it is in a glider.

That said, there are many other computers that could be used as an external controller. Selecting the proper external controller comes from the perspective of the science mission, as excess computing power increases power consumption, along with size and/or shape, connection ports and useable communication protocols, and weight considerations.

Backseat Driver Concept, Wiring, and Implementation

In this thesis the term: BSD is the conceptual context of adding additional processing capability onboard a vehicle outside its standard control scheme for improved adaptability; Slocum BSD architecture is the hardware and software required to implement the BSD concept onboard the Slocum glider (the physical cable connections, communication protocols, and the like); external controller is the additional computer added to the existing glider control structure.

A Slocum glider has a flight computer and a science computer. The flight computer handles all behavior characteristics and communication protocols with the shore. The science computer handles all scientific sensor and data collection protocols. Although different, the two computers talk with one another via a logic pathway called the clothesline. However, the flight computer typically does not adapt its behaviors based on what the science computer "sees" of the ocean. Like the clothesline between the two glider computers, the science computer and external controller are connected via the science computer's sensor ports. To the glider, the external controller functions like a typical science sensor where specific abilities about when to sample (in this case when to be powered on) are determined by the glider.

Standard behavior arguments (b_args) are set once at mission start by the values throughout the glider files and on the surface mid-mission by pilots altering the files via Iridium. The Slocum BSD architecture allows for b_args to be dynamic while under the surface of the ocean by giving the external controller the ability to request and overwrite certain b_arg values stored in glider memory, allowing for in-situ autonomous adaptive control.

The external controller is used and/or controlled in multiple files on the glider. When a mission begins, the glider initializes values from onboard files and stores them in the flight computer's random-access-memory. The external controller then actively overwrites these values as the glider is flying, altering the behavior at that point in time. By allowing the b_args to be dynamic, new glider behaviors can be actualized through the external controller's dynamic controllability and potential increase in computing power, if greater processing is required.

Simulating a Western Boundary Current with a Glider Simulator and SFMC Script

Our Slocum BSD architecture use case changes the glider's heading mid-mission based on the vehicles calculated values for ocean current speed and direction; however, in standard glider simulation the ocean conditions are set once at mission start and remain static throughout the simulation. Thus, in order to test this application, we first developed a way to update the glider simulator's water direction and magnitude mid-mission by using a Slocum Fleet Mission Control (SFMC) script.

SFMC is the web application used to manage a Slocum glider in real-time [16]. One of SFMC's capabilities is user-configurable scripts to control the glider, freeing a human pilot from directly issuing commands during every surfacing. A typical script will read the surfacing output from the glider and issue commands for sending and receiving files before telling the glider to resume the mission and dive. Utilizing this system for our simulation glider, we created a script that reads the glider's longitude from the surface dialog. The script then issues two commands (!put s_water_speed and !put s_water_direction) based on a user defined representation of an ocean current environment. Once the simulated ocean variables are updated, the glider simulator resumes flight using the new values, creating the effect of glider traversing a changing current field.

SIMULATIONS

Using the shoebox simulator, the Raspberry Pi 4 as the external controller, and the associated wiring and programming, four simulated missions were run in an SFMC controlled current field roughly approximating the Yucatán Channel's Depth Average Current (DAC). All four began at the same location just offshore the Yucatán Peninsula and represent 22 days and 7 hours of simulation time. The first two missions were run using the goto_list behavior: Mission 1, with a waypoint due east of the start location, and Mission 2, with a waypoint due southeast of the start location. Both waypoints were extremely distant, on the same longitude as the westernmost part of Africa, a common practice to effectively program the glider to fly on a constant heading. Two missions (3 and 4) were run utilizing the Slocum BSD architecture with the set_heading behavior using the external controller to control the glider's heading based upon the glider's calculated DAC. When the glider surfaced and updated its DAC calculation, the external controller would read the new current value and update the glider's heading.

RESULTS

Given acceptable time and deflection distance constraints, the Slocum glider utilizing the BSD architecture better met the mission objective of crossing the ocean current. Although Mission 2 had the shortest northward deflection distance it took an excessive amount of time. Mission 3 and 4, with headings between Mission 1 and 2, crossed the currents at a more optimal angle, better meeting the time vs distance trade-off (Figure 2A). In other words, the missions controlled with the Slocum BSD architecture crossed the fast moving currents in less time and with less northward deflection than when using standard practices for piloting gliders in areas of fast-moving currents.

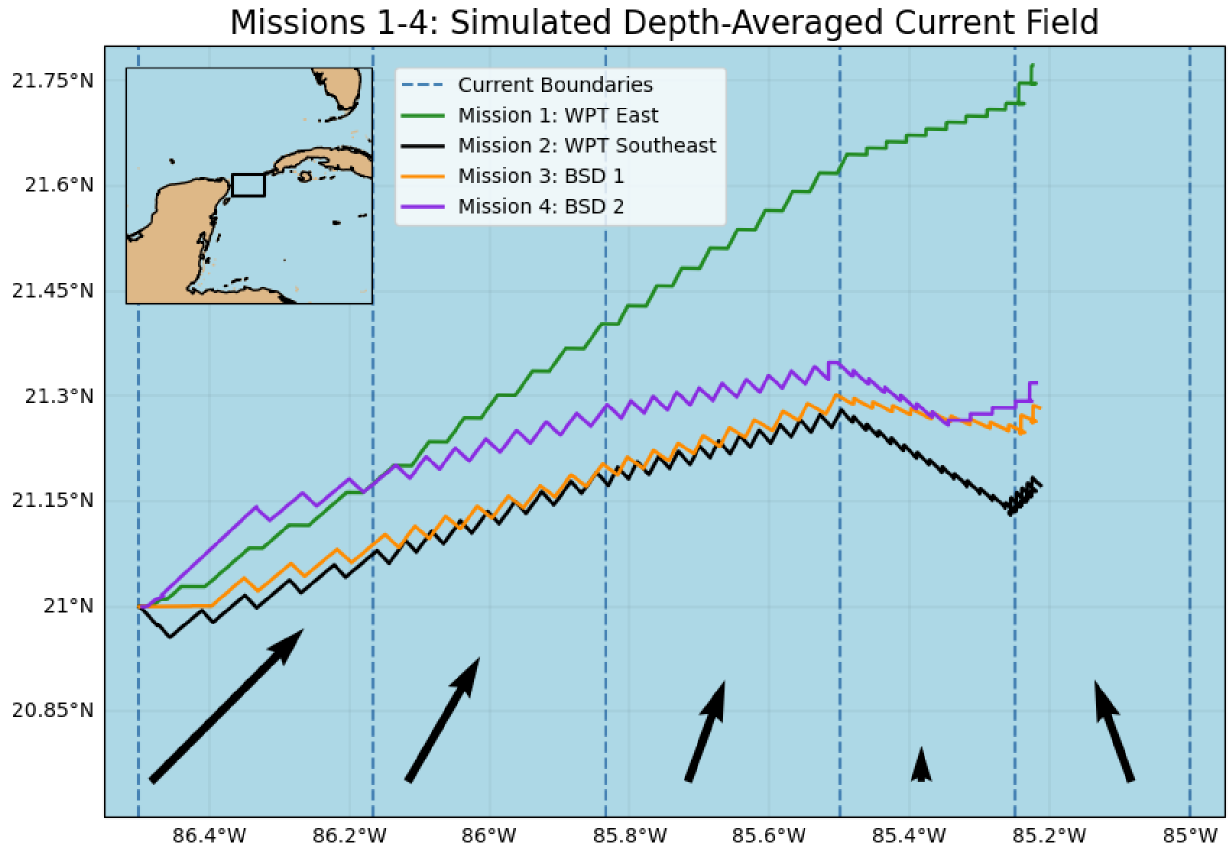

Figure 1: Blue dashed lines delineate the boundaries between changing currents, with the arrow in each vertical slice of longitude indicating the magnitude and direction of currents within. All gliders started at 21° N, 86.5° W and proceeded east. Four missions were ran in this current field: (1) Green has a waypoint due east of the start location near Africa; (2) Black has a waypoint due southeast of the start location, between South America and Africa; (3) Orange is controlled with the Slocum BSD architecture setting the heading 90° to the right of the DAC; (4) Purple also uses the Slocum BSD architecture to control the heading based upon the DAC, with the addition of calculations to pick the heading that should result in the smallest theoretical northward deflection distance. The sawtooth pattern is due to the time underwater vs time on the surface. The simulated vehicle is subject only to currents (there is no wind forcing).

Missions 3 and 4 demonstrate the viability of operating an external controller for a total of 8.59 days. While this is not conclusive for weeks-long or multi-month missions, this indicates an external controller is capable of proper operation for the extended periods of time required for a glider mission.

That said, the external controller's behavior is not always predictable. While the same Python script ran for the duration of Mission 3 and 4, its execution time ranged from 18.46 seconds to 43.19 seconds. This wide variability, while troubling, has some potential solutions [15].

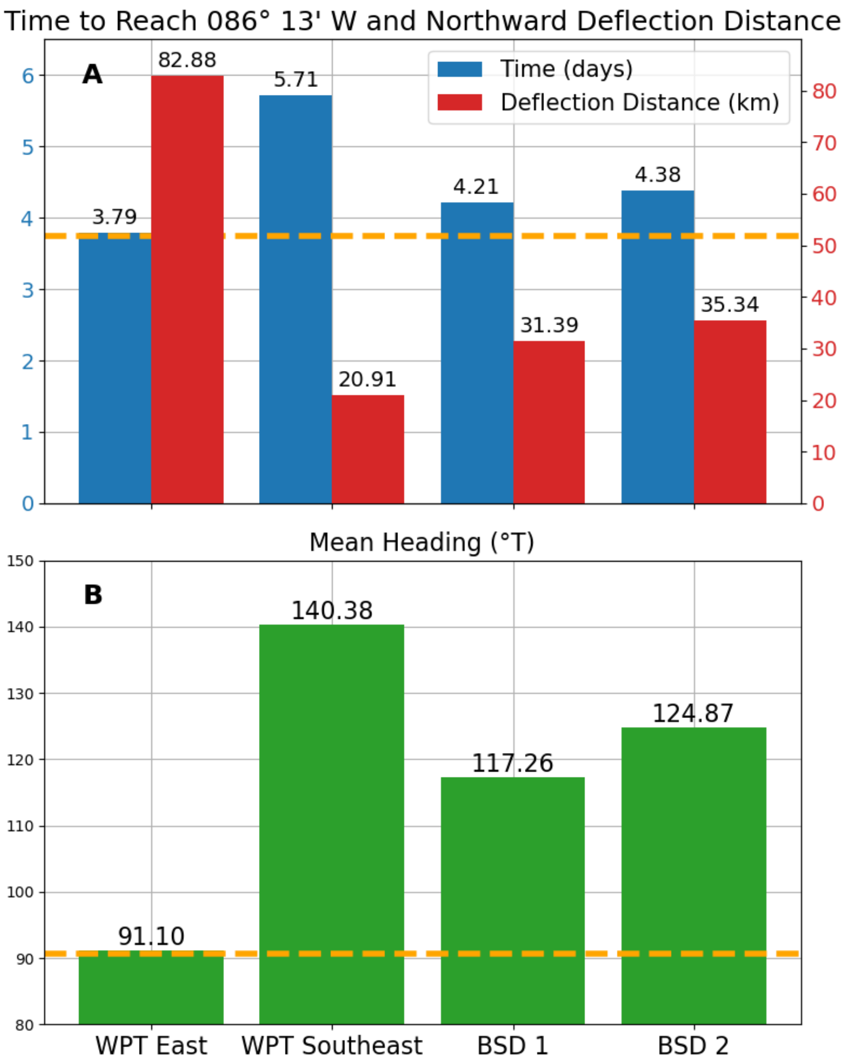

Figure 2: **A –** Mission 3 and 4's times and northward deflection distance is more optimal. **B –** Mission 3 and 4 are between the two waypoint mission heading extremes.

Although setting a distant waypoint could result in the more optimal mean heading of 117°T from Mission 3 (Figure 2B), doing so would continually update the heading while submerged. In contrast, the two missions utilizing the BSD architecture did not update the heading as often (Figure 3). This is due to the external controller reading the DAC upon surfacing, changing the set_heading behavior's c_heading before diving and resuming the mission. While underwater, the glider's c_heading does not alternate back and forth towards a distant waypoint; instead, it remains at the fixed value set by the external controller during the surfacing.

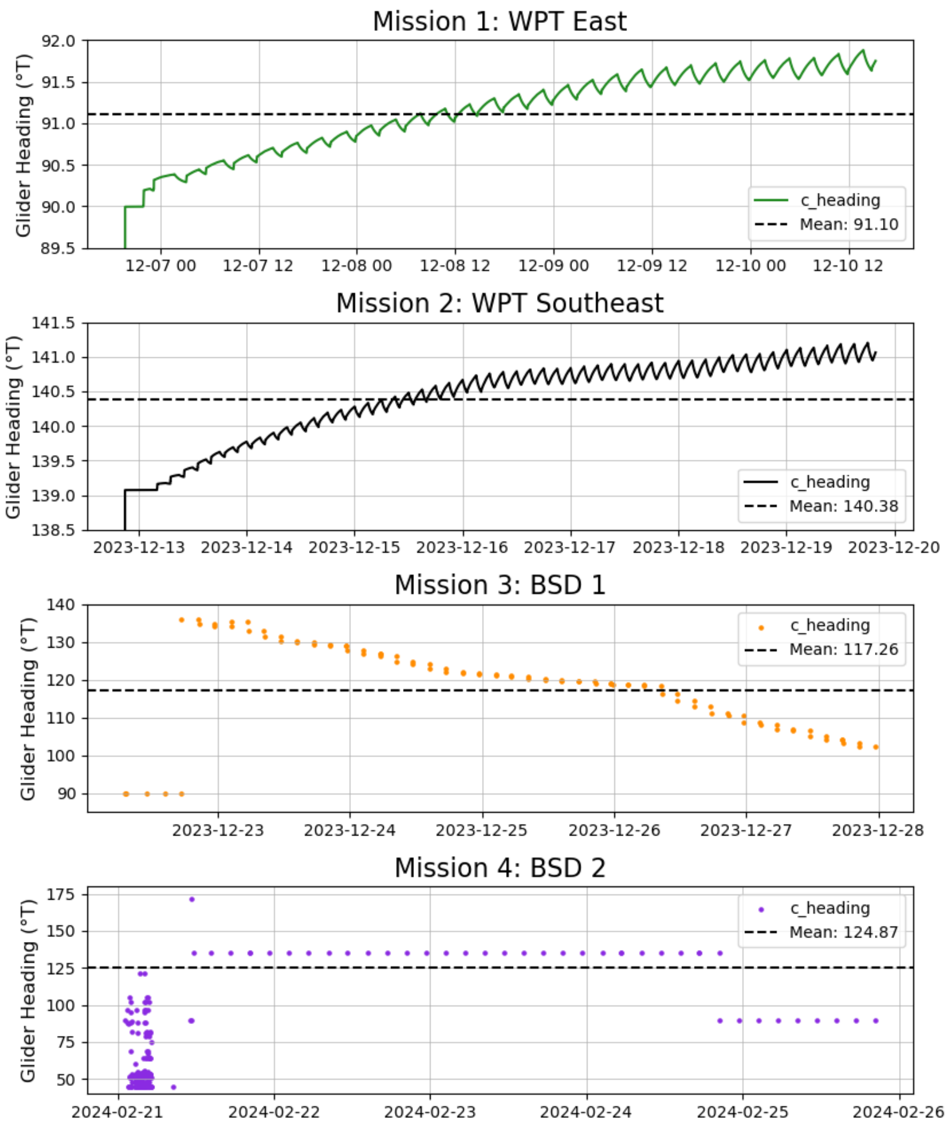

Figure 3: The green, black, orange, and purple corresponds with Figure 1. Mission 1 and 2 are waypoint missions; thus c_heading yaws back and forth as the glider flies towards a distant waypoint. Mission 3 and 4 utilize the BSD architecture updating the set_heading behavior with c_heading only changing when m_water_vel_dir is updated at the surface. At the start of Mission 4 m_water_vel_dir improperly represented s_water_direction causing the external controller to update the heading often and not accurately.

The presence of the external controller enables the fine-tuning of the desired cross-current heading to optimize deflection distance versus time, balancing the trade-offs of spending more time for a lower deflection. This also shows the capability for the external controller, via the Python script, to adapt a behavior on a glider as the end-user sees fit.

Backseat Driver Manual

To further enhance communities' resources, in this work I have written a manual, referred to throughout, which provides numerous use-cases of how to use the Slocum BSD architecture: *Implementing the Backseat Driver Architecture on a G3S Slocum Glider: A Manual for Operators* [15]. The manual is intended as a starting point for anyone to leverage the BSD architecture on a Slocum glider for their own specific needs.

It contains background, practical guidance, and examples for configuring an external controller, integrating real-time glider sensor data, datafile transfer and management, simulating scientific sensors for validating proper external controller behavior, and more. The manual and all relevant files are available on GitHub.

DISCUSSION

There are many ways to implement a BSD in a given system; this is but one. While BSD work alone is not uncommon, this work has gone an extra step by creating an extremely detailed manual so future users can leverage the Slocum BSD architecture to fit their specific use-case. However, it is worth remembering that there is no "set" way to utilize the Slocum BSD architecture. Choice of external controller, type of control and glider sensor values are all use-case specific questions that must be decided upon.

This work has one main limitation: the glider simulator is not designed to accurately represent a real-life ocean environment, requiring the SFMC script and simulated science sensor work-around. With further development, the SFMC script could integrate latitude for a higher-resolution simulated ocean current and more simulated sensors could be made; however, the simulator's present capabilities place restrictions on lab-based testing, necessitating in-situ ocean missions to properly validate glider characteristics.

Furthermore, the time involved with tests of this nature, simulating glider missions, is long, and there is at present no way to speed up the glider's simulation. Moreover, only one simulation can be run at a time, necessitating thorough testing, debugging, and verification before committing to a multi-day, or longer, test.

CONCLUSION

While this work and operational manual created is one step as we continue towards ever greater capability for answering our science questions, there are far more possibilities within the BSD concept than anyone can possibly hope to cover in one work.

As the true strength of using a BSD is the creativity in unique vehicle control it can afford, we must continue to leverage and expand upon our existing structures to better capture relevant data. Thus, using the Slocum BSD architecture should be paramount. To realize this expansion of the Slocum glider's potential, education and outreach is required.

Greater collaboration between the engineers creating these systems, scientists desiring data, and the pilots operating the vehicle is necessary. Translation between the different groups and mindsets to leverage the strengths of each towards the common goal of furthering oceanographic data collection, and ultimately humanities relationship with our planet, is in dire need.

Using the knowledge gained from everyone's backgrounds, we can better implement this incredibly powerful method to enhance the Slocum glider's capabilities for data collection. The Slocum BSD architecture is limited only by our imagination, and our ingenuity.

REFERENCES

[1] O. Schofield *et al.*, “Slocum Gliders: Robust and ready,” *J. Field Robot.*, vol. 24, no. 6, pp. 473–485, 2007, doi: 10.1002/rob.20200.

[2] E. P. Abrahamsen, “Sustaining observations in the polar oceans,” *Philos. Trans. R. Soc. Math. Phys. Eng. Sci.*, vol. 372, no. 2025, p. 20130337, Sep. 2014, doi: 10.1098/rsta.2013.0337.

[3] T. N. Miles, S. J. Coakley, J. M. Engdahl, J. E. Rudzin, S. Tsei, and S. M. Glenn, “Ocean mixing during Hurricane Ida (2021): the impact of a freshwater barrier layer,” *Front. Mar. Sci.*, vol. 10, Aug. 2023, doi: 10.3389/fmars.2023.1224609.

[4] T. N. Miles *et al.*, “Uncrewed Ocean Gliders and Saildrones Support Hurricane Forecasting and Research,” *Oceanography*, vol. 34, no. 4, pp. 78–81, 2021.

[5] C. P. Jones, “Slocum glider persistent oceanography,” in *2012 IEEE/OES Autonomous Underwater Vehicles (AUV)*, Sep. 2012, pp. 1–6. doi: 10.1109/AUV.2012.6380738.

[6] J. Hwang, N. Bose, H. Nguyen, and G. Williams, “AUV adaptive sampling to delineate subsurface oil plumes,” in *2020 IEEE/OES Autonomous Underwater Vehicles Symposium (AUV)*, Sep. 2020, pp. 1–2. doi: 10.1109/AUV50043.2020.9267920.

[7] Y. Wang, N. Bose, W. Thanyamanta, C. Bulger, and S. Shaikh-Upadhye, “Adaptive control for follower gliders mapping underwater oil patches,” *J. Hazard. Mater.*, vol. 436, p. 129039, Aug. 2022, doi: 10.1016/j.jhazmat.2022.129039.

[8] M. Zhou, R. Bachmayer, and B. deYoung, “Mapping the underside of an iceberg with a modified underwater glider,” *J. Field Robot.*, vol. 36, no. 6, pp. 1102–1117, Sep. 2019, doi: 10.1002/rob.21873.

[9] Y. Zhang, J. G. Bellingham, M. A. Godin, and J. P. Ryan, “Using an Autonomous Underwater Vehicle to Track the Thermocline Based on Peak-Gradient Detection,” *IEEE J. Ocean. Eng.*, vol. 37, no. 3, pp. 544–553, Jul. 2012, doi: 10.1109/JOE.2012.2192340.

[10] Y. Zhang, M. A. Godin, J. G. Bellingham, and J. P. Ryan, “Using an Autonomous Underwater Vehicle to Track a Coastal Upwelling Front,” *IEEE J. Ocean. Eng.*, vol. 37, no. 3, pp. 338–347, Jul. 2012, doi: 10.1109/JOE.2012.2197272.

[11] Y. Zhang *et al.*, “Autonomous Tracking of Salinity-Intrusion Fronts by a Long-Range Autonomous Underwater Vehicle,” *IEEE J. Ocean. Eng.*, vol. 47, no. 4, pp. 950–958, Oct. 2022, doi: 10.1109/JOE.2022.3146584.

[12] J. C. Gradone, E. J. Hunter, J. Mullison, and T. N. Miles, “Development of Onboard Processing Capabilities for a Slocum Glider Acoustic Doppler Current Profiler,” in *OCEANS 2021: San Diego – Porto*, Sep. 2021, pp. 1–5. doi: 10.23919/OCEANS44145.2021.9705895.

[13] P. K. Igoe, S. R. Sideleau, and R. Balasubramanian, “Real-time Sensor Data Processing and Reaction in the Context of Autonomous Systems,” in *OCEANS 2018 MTS/IEEE Charleston*, Charleston, SC: IEEE, Oct. 2018, pp. 1–5. doi: 10.1109/OCEANS.2018.8604749.

[14] “JMSE | Free Full-Text | A Backseat Control Architecture for a Slocum Glider.” Accessed: Oct. 24, 2023. [Online]. Available: https://www.mdpi.com/2077-1312/9/5/532

[15] “sscopazzi/Slocum_Backseat_Driver_Manual” GitHub. Available: https://github.com/sscopazzi/Slocum_Backseat_Driver_Manual

[16] “SFMC - About Slocum Fleet Mission Control.” Accessed: Apr. 21, 2024. [Online]. Available: https://glider.marine.rutgers.edu/sfmc/about