Integrated Command, Control and Communication for Autonomous Underwater Vehicles

Clancy Emanuel
Department of Electrical and Computer Engineering
University of Connecticut
Storrs, Connecticut 06269
clancy.emanuel@uconn.edu

Daanish Zaidi
Department of Computer Science
University of Connecticut
Storrs, Connecticut 06269

Shalabh Gupta
Department of Electrical and Computer Engineering
University of Connecticut
Storrs, Connecticut 06269
shalabh.gupta@engr.uconn.edu

Shengli Zhou
Department of Electrical and Computer Engineering
University of Connecticut
Storrs, Connecticut 06269

***Abstract*—The paper presents a novel integration of underwater communication, localization, command and control for the purpose of point-to-point movement of an *Autonomous Underwater Vehicle* (AUV). An acoustic communication network is used to command the AUV to a desired point within the coordinate frame of the network which provides two-dimensional localization. This information is fused with path planning and control algorithms. A PID controller is developed, tested, and tuned, allowing the AUV to maintain a desired heading to accurately maneuver to the destination point. A robust multi-process software architecture is developed to eliminate single-point failures during mission critical routines. The integrated algorithms and hardware developed are experimentally tested and validated in a swimming pool.**

I. Introduction

Sensing, communication, and data processing technologies for underwater exploration are rapidly developing leading to a growth in utilization of *Autonomous Underwater Vehicles* (AUVs). AUVs are desired to have the capabilities of autonomous exploration [1], monitoring [2], [3], and search and rescue [4], [5] in an underwater environment. To achieve these tasks, the AUV must integrate sensing, communication, localization, and command and control to allow it to navigate and observe its surroundings. However, localization in an underwater environment must be achieved using an acoustic communication network which requires acoustic signal propagations with low data rates. This can lead to large estimation errors in the localization, thus making it very difficult to control the AUV to navigate to a specific location.

To address the challenge of localization in an uncertain underwater environment, technical literature has proposed rigorous models to understand the effects of fluid dynamics and thrust. These approximations rely on a state-space representation of the AUV system with nonlinear models for simulating movements and disturbances [6], [7]. Once the system dynamics of AUVs are thoroughly understood, path planning algorithms have been proposed to optimize on a variety of constraints such as coverage [8], obstacle avoidance, fuel consumption, etc. [4], [9], [10]. The foundation for the synthesis of system dynamics and path planning algorithms lies in the ability to communicate underwater. Researchers have developed underwater modems that provide both a channel for communication and a method of localization [11]. GPS [12] and Wi-Fi [13] localization have also been proposed but cannot be achieved underwater. Although the above challenges have been addressed to a certain degree, the integration of communication, control, and localization for underwater exploration have not been proposed in literature.

This paper presents the development of an AUV for autonomous exploration that integrates underwater communication, localization, command and control. A PID controller is developed, tested, and tuned, to allow the AUV to maintain a desired heading for the purpose of point-to-point navigation. A robust multi-process software architecture is developed to eliminate single-point failures during mission critical routines. The platform developed is experimentally validated based on multiple pool tests and show that the integration of communication, localization, command and control is achieved.

The findings are presented in the following sections. Section II explains the hardware design of the AUV used in this project. Section III describes the software architecture while Section IV describes the algorithms used in the project. Finally Section V presents the results from the pool tests.

II. Hardware Design

The AUV designed in this work is a combination of custom electronics coupled with a Neptune Thunder Tiger RC submersible which provides no control or propeller feedback.

Inside the hardware stack are a custom DSP board capable of 21-27 KHz communication, a PandaBoard computer, an Arduino, and a modified remote control device. The DSP board receives mechanical waves via the hydrophone array, shown in Figure 1, which are then translated into digital signals for the PandaBoard. The PandaBoard is the on-board computer with an ArmV7 architecture, 1.2 GHz processor and 1 GB of RAM. This is connected to an Arduino via FTDI serial-to-USB chips. The Arduino is connected to the modified remote control board by a custom wiring solution. The radio frequency component from the stock handheld controller is installed inside of the hardware stack, giving the PandaBoard wireless control of the AUV motors. In order to communicate with the AUV, an AquaSeNT shallow water modem, shown in Figure 2, is used. The signal flow chart shown in Figure 3 describes the communication between these components.

978-0-933957-43-5 ©2015 MTS

Fig. 1. The full setup for the AUV.

Fig. 2. One of the AquaSeNT shallow water modems used during testing. The modems operated at 22KHz during our tests.

This work encountered several issues related to the on-board compass and lack of feedback. The motors on the AUV do not have encoders making the compass measurements the only signals fed back to the controller. The compass in the hardware stack is an Ocean Server 5000 series that provides full IMU capability. This is used for the yaw measurements which have an accuracy of $\pm 0.5°$ [14]. The first issue faced was the mounting location of the compass. Initially, it was directly above the motors of the AUV and this led to severe distortion of the compass readings due to the magnetic field generated by the motors. Placing the compass on the front of the AUV led to accurate measurements. Another problem is that the compass' $0°$ heading was not aligned with the pool's $0°$ heading. This was expected and is dealt with in section IV-B. The problem that this work was unable to solve was the dilation of the angle intervals in the southwestern direction. Several hard-iron and soft-iron calibrations were performed and they did not solve the problem. This phenomenon may have occurred due to the test facility where the experiments were conducted.

III. Software Design

The core functionality of the AUV was separated into logical, individual subsystems that operate concurrently. We developed the modem, path-planning, attitude control, motor control, and compass processes using Robot Operating System (ROS) and C++. Robot Operating System is a "middleware" that allows for modular software design involving a multi-process architecture. Figure 4 gives a high level design overview of the code operating the AUV system.

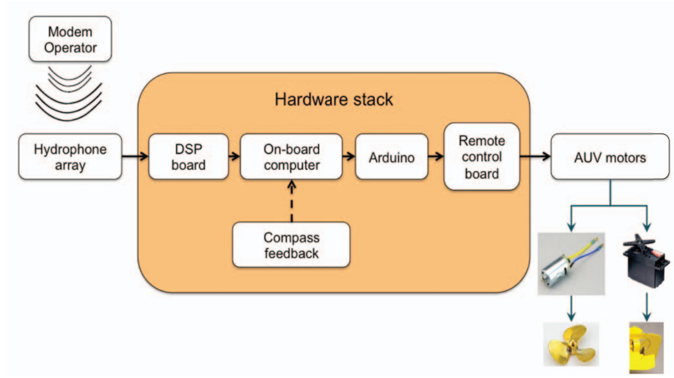

Fig. 3. Signal flow chart for the AUV.

While the three algorithms developed for this project run sequentially, the platform developed allows each algorithm to have its own process, thus avoiding a single point of failure.

IV. Algorithms Implemented

Several algorithms run while the AUV is operating. The order in which the algorithms are explained below matches the order in which they are executed. Upon receiving a destination coordinate the AUV localizes and passes this information to the guidance algorithm, where the PID controller commands the motors to reach the desired coordinate.

A. Localization

The first algorithm is Asynchronous On-Demand Localization, developed by Mahmood et. al. [11]. While the AUV is stationary it broadcasts a request containing an ordered list of acoustic modems, initiating the localization procedure. The modems respond sequentially on the same channel in a round-robin fashion. The time stamp of each response is used to compute time-of-arrivals, allowing the AUV to triangulate its position [11].

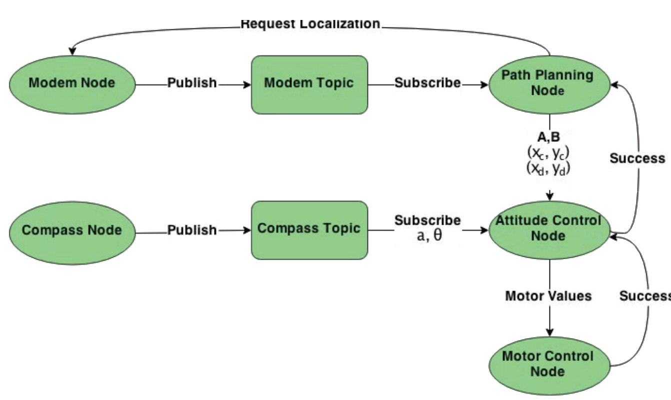

Fig. 4. Custom ROS software architecture developed for the AUV. The ovals represent ROS nodes, which can be thought of as separate processes or subsystems. The rectangles symbolize the data topics which provide a publish/subscribe platform for inter-node communication.

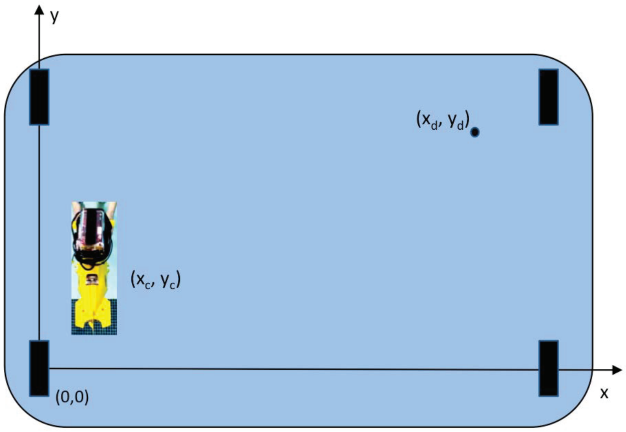

Fig. 5. The AUV knows about (x_c, y_c) and (x_d, y_d) after the localization routine.

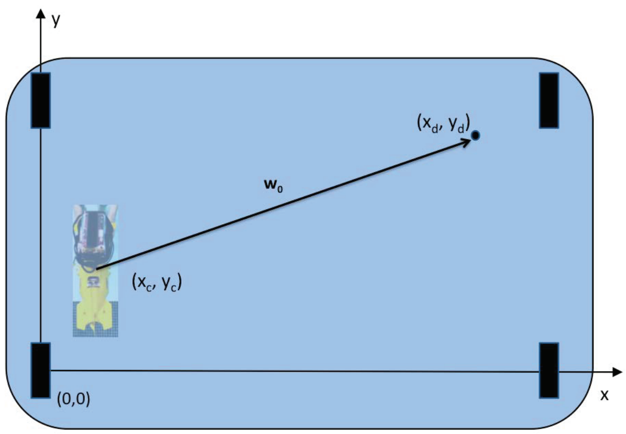

Fig. 6. The AUV computes $\mathbf{w_0}$.

B. Guidance

After finishing the localization routine, the AUV knows its current position and destination coordinate, (x_c, y_c) and (x_d, y_d), respectively, as shown in Fig. 5. With these two points the AUV computes the vector $\mathbf{w_0} = [(x_d - x_c), (y_d - y_c)]$ where the 0 subscript representing the absolute frame of the pool, as shown in Fig. 6.

The AUV performs a homogeneous transformation to reconcile the differences in translation and rotation between the relative frame of the AUV and the absolute frame of the pool. This new vector is defined as $\mathbf{w_1}$. The AUV turns to follow $\mathbf{w_1}$ and once it is lined up with that vector within a certain margin of error, MOE, it starts a timer which controls how long the propeller is on. This time is deduced using the magnitude of $\mathbf{w_1}$ and the constant AUV_SPEED, explained in section V.

C. Control

The yaw of the AUV is controlled by a PID controller shown in Figure 7, where θ_d is the yaw setpoint given by $\mathbf{w_1}$, θ_c is the current yaw given by the OS5000, ϕ is the error signal, Yaw_DP is the output to the motors, MOE is the angular margin of error, T is the amount of time the propeller is left on, and K_p, K_d, and K_i are the PID gains determined by the Ziegler-Nichols method. The Ziegler Nichols method is a heuristic used to tune the three gains of a PID controller: K_p, K_i and K_d[15]. This work used a heuristic approach as opposed to a model-based design because developing a robust plant transfer function was not within the scope of this work. The first step of this method is to use only proportional control and find the value of K_p that gives constant-amplitude steady state oscillations. This value of K_p is defined as K_{cr}. Once K_{cr} is found the next step is to use the following formulas the obtain K_i, K_d, and K_p:

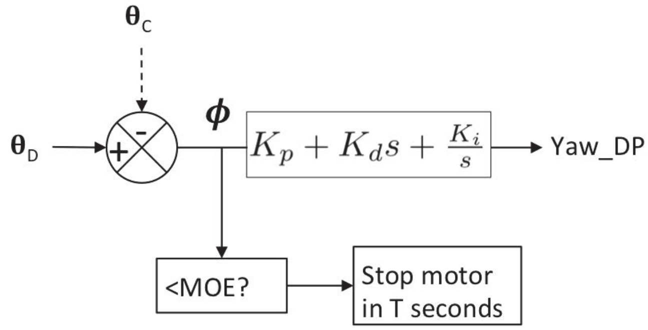

Fig. 7. The block diagram for the PID controller. The dashed line indicates feedback from the on-board IMU.

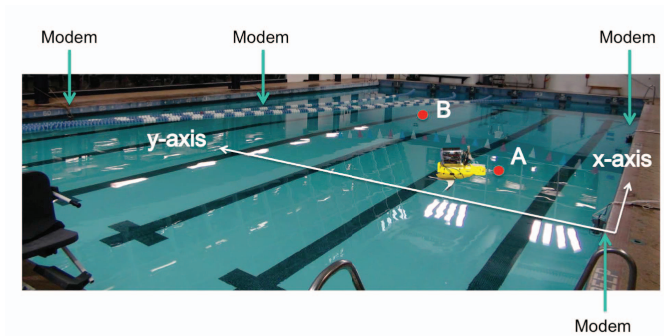

Fig. 8. The full test environment.

$$K_p = 0.6K_{cr} \quad (1)$$

$$K_d = 0.125P_{cr} \quad (2)$$

$$K_i = 0.5P_{cr} \quad (3)$$

These values were found and explain the necessary modifications made in section V.

V. Testing and Results

The testing environment is a 25x13 meter swimming pool shown in Fig. 8. Four AquaSeNT shallow water modems are used for communicating and localization of the AUV.

Given that the localization algorithms were already tested, this work designed the guidance and control algorithms. These algorithms were experimentally validated in over 60 hours of

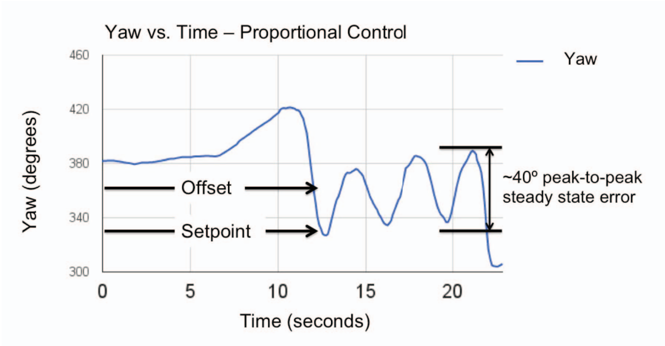

Fig. 9. The plot of the AUV's heading directed by proportional control when given a command to turn 90° starboard. The controller missed its setpoint and experienced unacceptably large oscillations at steady state.

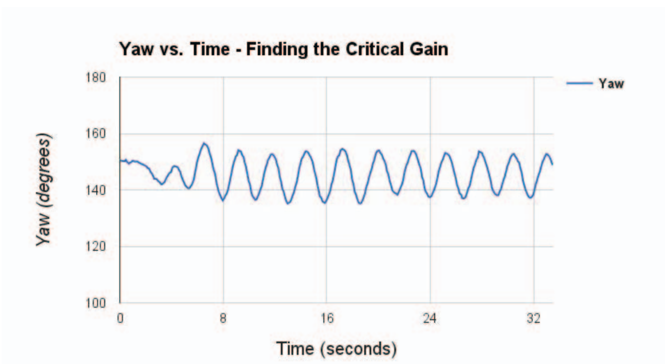

Fig. 10. 20° steady state oscillations with a period $P_{cr} = 2.5$ seconds caused by a proportional gain of $K_{cr} = 13$.

testing. As seen in Figure 9, initial tests proved that a simple proportional controller was not suitable for determining the heading of the AUV, which lead to the development of a PID controller.

The proportional controller provided an approximation of the AUV's linear speed when traveling in a straight line. These tests were impossible without control because the AUV would veer to the right or left immediately upon commanding it forward. Having the minimal control provided by the proportional controller allowed for commanding the AUV in a relatively straight line to determine its speed. This speed, defined as the constant AUV_SPEED, is used in the guidance algorithm to determine the time for which the AUV motors must run in order to travel the necessary distance.

Due to the lack of control provided by the proportional controller a PID controller was implemented as described in section IV-C. Using the first step of the Ziegler-Nichols method as described in section IV-C constant steady state oscillations were achieved with a critical gain $K_{cr} = 13$, as evidenced by Figure 10.

After trying these values and tuning them even further the

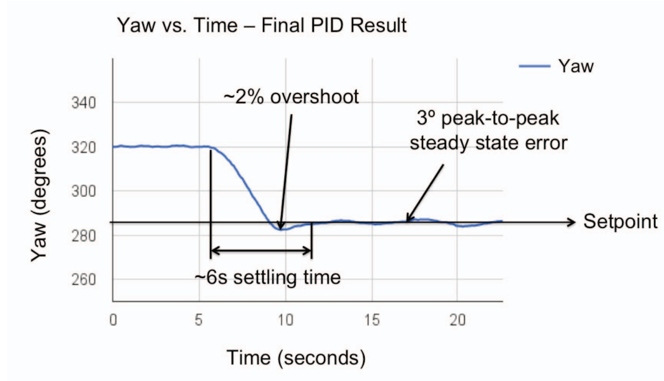

Fig. 11. Successful verification of the PID controller.

following values for the PID gains were achieved:

$$K_p = 6 \tag{4}$$

$$K_d = 0.325 \tag{5}$$

$$K_i = 1.75 \tag{6}$$

Figure 11 demonstrates the final result of the PID controller when given a command to head 45° portside.

Once the full PID control was designed 32 experiments were performed to test the AUV's ability to navigate to a certain location in which several experiments were successful while some were unsuccessful due to modem hardware issues. In the first successful test the AUV was placed in the middle of the pool and commanded to travel to the coordinate $(0, 0)$ in the frame of the pool coordinate space, shown in Fig. 12. Upon receiving the destination coordinate, the AUV initialized the localization routine, used its current heading to determine $\mathbf{w_1}$ as described in IV-B, and finally the attitude node controlled the AUV to arrive within 1 meter of the destination.

Another test was performed in which the AUV was commanded to go from its current position to $(7, 0)$, the position of one of the modems in the pool. The goal of the test is shown in Figure 13. Figures 14 and 15 show the AUV at the beginning of the test and the AUV at the end of the test, respectively.

The above tests are a proof of concept. In every test, the test was quantified as a success or failure based on visual observation and the knowledge of the pool currents, inherent compass errors, and localization errors. Every test in which the modems worked was a success in that the AUV traveled to the destination. These tests served the purpose of confirming that the localization, path planning, and control algorithms function successfully as a unit.

VI. Conclusions and Future Work

This paper presents an integration of communication, localization, command, and control for the purpose of providing simple, point-to-point movement of an AUV. The benefit of this innovation is that AUVs using acoustic communication will be able to determine their attitude and plan a three-dimensional trajectory without surfacing, allowing them to potentially remain at operational depth indefinitely. Further

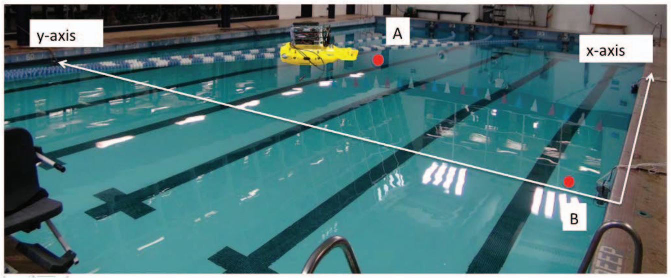

Fig. 12. The AUV was commanded to go from its current position, A, to its the destination $(0, 0)$, labeled as B in this figure.

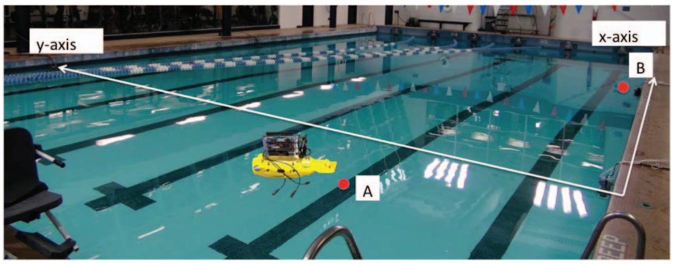

Fig. 13. The goal of the second successful test was for the AUV to travel to point B, which is the location of one of the modems.

Fig. 14. The AUV at the beginning of the test.

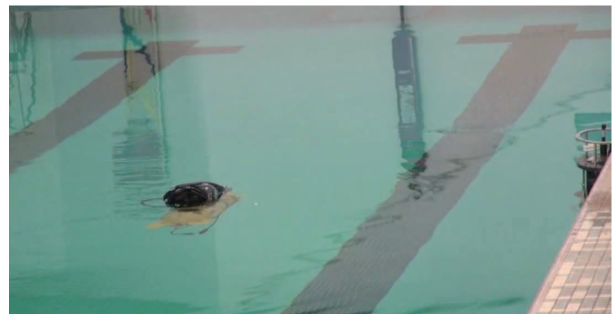

Fig. 15. This shows the AUV when it stopped. It successfully traveled to the correct location in the pool within the margin of error of the modems.

work includes verifying these algorithms in other environments, adding state estimators while localization is not active, and improving the localization algorithm to provide data while the AUV is moving.

ACKNOWLEDGMENTS

The authors would like to thank Mr. Matt McPhail and Ms. Samantha McNellis for designing, constructing, and running all the hardware; Mr. Tony Pham, Mr. Sam Ginzburg and Mr. James Hare for their support at various levels, and every other student that has been involved in this project.

REFERENCES

[1] J. Li and M. Johnson-Roberson, "Multi-altitude multi-sensor fusion framework for auv exploration and survey," in *OCEANS, 2014*, 2014.
[2] H. Kondo, M. Sato, T. Hotta, A. Withamana, M. Osakabe, and Y. Matsumoto, "Development of a marine ecosystem and microstructure monitoring auv for plankton environment," in *Autonomous Underwater Vehicles (AUV), 2014 IEEE/OES*, 2014.
[3] P. Ramos, N. Cruz, A. Matos, M. Neves, and F. Pereira, "Monitoring an ocean outfall using an auv," in *OCEANS, 2001*, 2001.
[4] M. Purcell, "Use of remus 6000 auvs in the search for the air france flight 447," *OCEANS*, 2011.
[5] W. Lin, C. Chin, and E. Mesbahi, "Remote robust control and simulation of robot for search and rescue mission in water," in *OCEANS, 2014*, 2014.
[6] T. Fossen, *Gudiance and Control of Ocean Vehicles*. John Wiley and Sons, Inc., 1994.
[7] G. Tao and J. Zhao, "Modeling and identification of autonomous underwater vehicle for advanced control," in *International Conference on Modelling, Identification and Control*, 2012.
[8] J. Song, K. Qiu, S. Gupta, and J. Hare, "Slam based adaptive navigation of auvs for oil spill cleaning," in *OCEANS 2014, St. Johns, Canada*, 2014.
[9] J. Goldman, "Path planning problems and solutions," in *Aerospace and Electronics Conference*, 1994.
[10] K. Carrol, S. McClaran, E. Nelson, D. Barnett, D. Friesen, and G. Williams, "Auv path planning: An a* approach to path planning with consideration of variable vehicle speeds and multiple, overlapping, time-dependent exclusion zones," in *Autonomous Underwater Vehicle Technology*, 1992.
[11] K. Mahmood, K. Domrese, P. Carroll, H. Zhou, X. Xu, and S. Zhou, "Implementation and field testing of on-demand asynchronous underwater localization," in *Asilomar Conference on Signals, Systems, and Computers*, 2013.
[12] F. Maeda, J. Osaku, A. Asada, Y. Yamagata, and T. Kanamaru, "R&d of the maneuver control technizques for autonomous surface vehicle using 'dual-extended kalman filter' for the actual harbor area surveillance," in *Underwater Technology Symposium (UT), 2013 IEEE International*, 2013.
[13] E. Marques and J. e. a. Pinto, "Auv control and communication using underwater acoustic networks," in *OCEANS 2007 - Europe*, 2007.
[14] *Digital Compass Users Guide*, Ocean Server.
[15] N. Nise, *Control Systems Engineering*. John Wiley and Sons, Inc., 2010.