Faculty and Researcher Publications

Faculty and Researcher Publications

1992

Experimental verification of mission planning by autonomous mission execution and data visualization using the NPS AUV II

EXPERIMENTAL VERIFICATION OF MISSION PLANNING BY AUTONOMOUS MISSION EXECUTION AND DATA VISUALIZATION USING THE NPS AUV II

A. J. Healey, D. B. Marco
Naval Postgraduate School
Monterey, Calif. 93943
(408)-646-2586 (Phone)
(408)-646-2238 (Fax)

ABSTRACT

This paper describes recent results in mission execution, and post mission data analysis from the NPS AUV II testbed underwater vehicle. Ongoing research is focused on control technology to meet the needs of future Naval Autonomous Underwater Vehicles. These vehicles are unmanned, untethered, free swimming, robotic submarines to be used for Naval missions including search, mapping, surveillance, and intervention activity. The approach taken at NPS combines integrated computer simulation, real time robust control theory, computer architecture and code development, vehicle and component design and experimentation, sonar data analysis and data visualization.

Started in 1987, the major thrusts of this overall research program are in the areas of **mission planning**, both off-line and on-line, **mission execution** including navigation, collision avoidance, replanning, object recognition, vehicle dynamic response and motion control, real time control software architecture and implementation, and the issues of **post mission data analysis**.

INTRODUCTION

This paper focuses on systems having to do with the Navy's use of Autonomous Underwater Vehicles (AUVs). AUVs are a class of underwater vehicles that are independent from mother ship support with respect to power and control. AUVs are **untethered - free swimmers** - with sufficient on-board intelligence to perceive uncharted and unplanned situations and take action in response. We are interested in these vehicles for a variety of military and/or commercial missions where direct human intervention is difficult or dangerous, and where the use of power cables and fibre optic data links are cumbersome. These vehicles will be used to gather data, provide surveillance, and possibly perform tasks in hostile areas. Research at NPS includes the issues of advanced controls for mission execution, vehicle motion control, sonar data processing for object recognition, and the post mission analysis.

Interest in intelligent **untethered underwater vehicles** has been growing recently. University groups include Texas A&M University, (Mayer et. al. (1987)) who have developed a knowledge based real time controller, hosted on SUN 4 computers with particular attention paid to hardware and software reliability; University of New Hampshire, who under the guidance of D.Blidberg (Chappell, 1987), have built and operated EAVE East vehicles since 1977 with ever increasingly complex computer architectures. EAVE III has a modular, hierarchical architecture using Motorola 68000 series computers running separate PSOS operating systems allowing for multi-processing operation. Lower level tasks are run in "C" while upper level tasks have been run in LISP, with the NIST RCS-3 real time control system (Albus, 1988). At MIT the Sea Grant Program has funded work conducted by Bellingham (1990a), who is exploring the demonstration of intelligent behaviors with a vehicle running on a GESPAC computer having a 68020 CPU with the OS-9 operating system and control code written in "C". The behaviors are hierarchically prioritized using the "Layered Control" concept (Brooks, 1988) although more recently, (Bellingham, 1990b) has seen fit to introduce a state based layered control to coordinate mission specific behavior. The University of Tokyo has recently developed an underwater vehicle for bottom contour following using neural network techniques, (Ura, 1990). At the Naval Postgraduate School, we have developed an underwater testbed vehicle that is specifically designed to test and verify developments in control technology. It is run in the NPS swimming pool as an environment for experimental mission demonstration using a GESPAC computer with a Motorola 68030 CPU a 2MByte RAM card with control code written in "C". The mission planning interface with the vehicle control computer is embodied in a GRiDCASE laptop MS-DOS machine containing mission details in the form of way points and run times that are obtained from an external pre-mission planning analysis. The NPS AUV II, shown as a sketch in Figure 1, is 72" long displacing about 380 lbs. having 2 propellors, 8 control surfaces, 4 thrusters and, at present, 4 single beam sonar channels (Healey and Good, 1992). Many industry groups (UNH Conferences 1987, 1989, the IEEE AUV-90, 91, 92 Conferences) as well as Navy Laboratories, DARPA and the DRAPER Laboratory have work ongoing in this area.

NPS AUV II SYSTEM OVERVIEW

While no formal control system structure has been adopted by all - in fact there are as many as there are investigators - our opinion is that a structure that includes the ability to first perform extensive (if time permits) simulations to verify that the predictable aspects of any mission will be executed in a feasible way, will be necessary. This would be regardless of the mission details. In our structure, this is done with the **Mission Planning Expert System** as shown conceptually in Figure 2. The output is a planned series of geographic way points that avoid charted problem areas and lead the vehicle to its operational site(s) with task descriptors at each target point. This plan encompasses launch, transit to the area, operating in the area, returning to home and recovery.

The **Mission Execution** phase is shown by the structure of activity in Figure 3. **Mission Execution** after launch is conducted between the **Mission Executor** and the **Guidance System** by breaking down the planned mission into a sequence of intermediate way points defined on a finer grid possibly having an adjustable spacing. In more critical areas the spacing would be suitably refined. The **Guidance System** thus interpolates the baseline grid to provide a refined series of way points which are passed to the vehicle **guidance law** and selected according to the degree of precision in path tracking desired by the mission plan. Three guidance laws have been studied viz. line of sight, cross track error, and the cubic spiral laws (Healey et. al., 1990). The **guidance law** generates the commands for vehicle's heading, speed, and depth. These commands are then sent to the vehicle **autopilot systems**. Three autopilots are installed for control over the vehicle's speed, heading, and depth. The servo levels of the vehicle's controller then provide final commands to the vehicle's propulsion plant, control surfaces, and thrusters to drive the vehicle to its planned path.

Obstacle avoidance and reflexive maneuvering logic are to be built into the vehicle's guidance system as a command override structure to respond to signals from the **Obstacle Avoidance Decision Maker (OADM)**. The **OADM** will receive input from **Pattern Recognition** software which correlates information from

0-7803-0704-6/92$03.00 ©IEEE

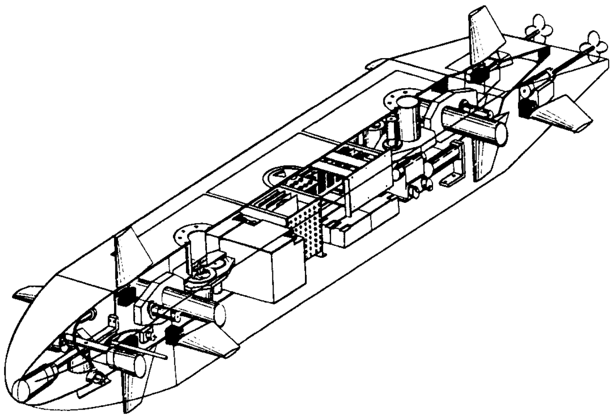

Figure 1
Sketch of the NPS AUV II Autonomous Underwater Vehicle

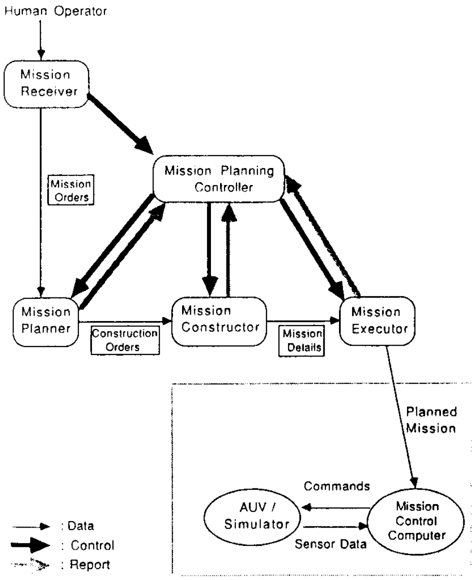

Figure 2

Mission Planning Expert System

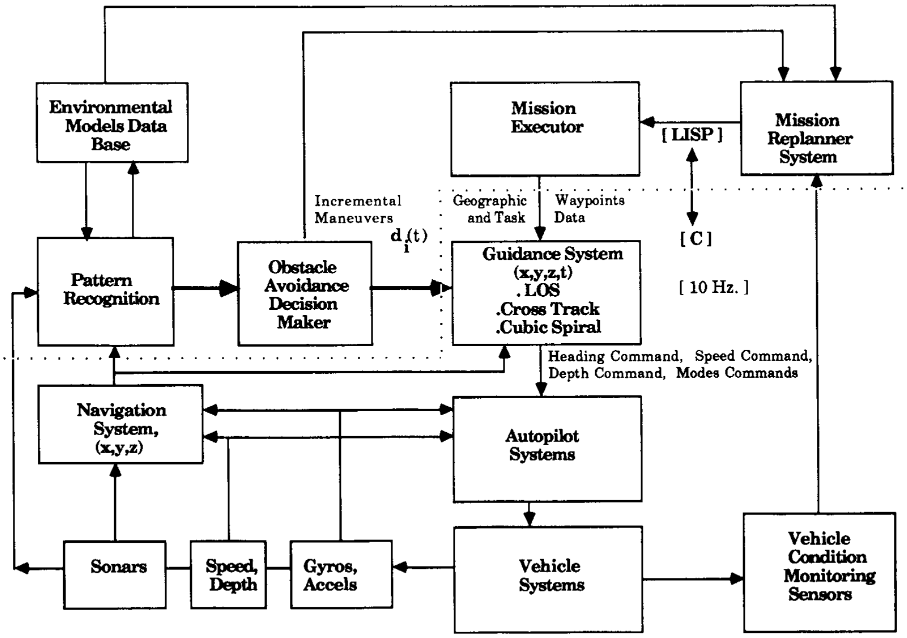

Figure 3

Mission Execution System Diagram

the sonars and estimates of present location and attitude from the **Navigation System** with an **Environmental Data Base** within the **Mission Executor**. The impending presence of an obstacle is thus flagged. At that time, status as to whether the object is stationary or moving, is to be reflexively avoided, or gradually outmaneuvered (either slow down, speed up, change course, which direction, etc.) is computed. Incremental modifications, $\mathbf{d}_i(t)$, to the planned way points and time are then made. Note that for all $\mathbf{d}_i(t)$ moves, $\mathbf{d}_i(t)$ will be said to tend to zero as t tends to infinity so that the originally planned path will be finally joined. Status signals are sent to the **Mission Replanner** from internal sensors concerning the condition of the internal equipment such as motor and battery status, motor controller system status, servo power and signal conditioning equipment, and power and internal temperature of the main CPU / Data Acquisition / Data Storage hardware.

Post mission data analysis is presently accomplished by down loading data that is stored in onboard RAM storage (19 channels of double precision data at a 10 Hertz rate) to the GRiDCASE laptop computer and then displayed on the data postprocessing computer. The postprocessor at this time lies in an IRIS graphics workstation containing graphics modules that replicate the environment in which the vehicle is operating together with modules for analysis of the vehicle motion data and the sonar sensory data obtained from the mission run (Brutzman et. al., 1992). The results of the planned mission are both simulated prior to mission approval using an IRIS workstation as the environment and vehicle simulator, and then displayed at mission completion. Details of the sonar imagery, or the bottom contour, or other mission specific results can be output in a user-friendly format.

MISSION EXECUTION SYSTEMS AND RESULTS FOR THE NPS AUV II

The execution of a mission begins with downloading the mission plan to the on-board **Mission Executor** followed by the vehicle launch. A time delay must be built into the executor to allow for the launching delay. It has been found important that during this launch phase, and especially with a fully autonomous vehicle, some indicator that the internal systems are functional is desirable - we have used a small movement of one of the control surfaces as this indicator. Upon program initiation, the mission execution plans (defined by the way points and time), are contained in an MS-DOS GRiDCASE laptop computer which is connected via serial link to the vehicle onboard GESPAC MPU30HF single board computer (based on the Motorola 68030 CPU, 25Mhz. with 2Mb of RAM and a 68882 math coprocessor) running with the OS-9 multi-tasking operating system and 2 GESDAC-2B 8 channel 12 bit DA/AD converter cards. Control code is written in "C" language. The GESPAC system is the interface between the mission planning phase and the vehicle hardware, and it houses the **Guidance System**, the **Navigation System**, and the speed, diving, and steering **Autopilot Systems**, each of which can operate under robust Sliding Mode Control. Figure 3 shows a diagram of the execution functions. Details of the design of the Sliding Mode autopilots have been given elsewhere (Healey, Papoulias, and Lienard, 1990, Papoulias and Healey, 1990) and will not be repeated here. The major interfacing in the execution phase is between the **Mission Executor** and the **Guidance System** and some interplay with the **OADM**. These systems, (in their future embodiment) are to be hosted in PROLOG or C++ language on an interface card running MS-DOS within the GESPAC computer while the **Guidance System** runs in "C" on the main processor board. Missions verified to date by experimental results have included way point following, sonar data analysis and object reconstruction, bottom contour following, and solid object avoidance.

Guidance and Motion Control

Experimental verification of line of sight guidance with PD and Sliding Mode autopilots has been accomplished in several missions run in the NPS swimming pool during the last year. Selected results will be described for missions including a figure eight maneuver using a coarse grid of way points, a depth control and altitude control mission, an obstacle avoidance mission, and missions to record and interpret sonar ranging data for object shape reconstruction.

Initial plans for testbed missions were to perform oval track runs in the NPS swimming pool. The missions were planned where the vehicle, operating under closed loop speed control, closed loop diving and steering control, followed a path with switch points defined at predetermined times at which heading commands were incremented from 0 to 180 to 360 degrees. In this way, the walls of the pool were avoided. This class of test run is helpful to identify the essential characteristics of the autopilot systems and has provided some interesting results shown in the series of Figures 4-7. Other runs including figure of eight, zig-zag, and spiral maneuvers have been completed. In Figure 4 the pertinent steering response variables are shown, Figure 5 gives the corresponding diving variables and Figure 6 shows the vehicle speed response. The path, as identified by dead reckoning ignoring side slip errors, is shown in Figure 7. Many other runs have been made recently and the results here will show a comparison of sliding mode controllers with more standard designs, bottom following performance with a downward looking sonar, and way point following in a figure eight maneuver.

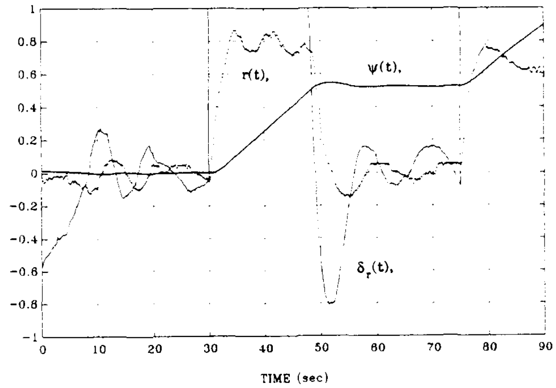

Figure 4

δ_r(t), r(t), ψ(t), versus Time. [Run 7-8-91-5]; Oval Track Run; Combined Diving, Steering, and Speed Control; Standard Control Laws; Scaled as δ_r(t)/0.4(rad), r(t)/0.2(rad/sec), ψ(t)/6.0(rad)

Steering Response

The steering response is shown in Figure 4. The Figure shows the behavior of a PD steering controller given by,

$$\delta_r(t) = K_r r(t) + K_\psi(\psi(t)-\psi_{com}(t))$$

where the rudder command signal time history is shown together with the corresponding yaw rate and heading angle. In the first thirty seconds, the vehicle is accelerating to speed, diving to depth, and controlling to the desired heading. At thirty seconds, the command to turn is entered and the response in the turn is clearly seen. The performance of the controller, however, is not elucidated when the turn is entered because the rudders are saturated. It is the control when the vehicle exits from the turn during the period 50 - 70 seconds that is key. The controller represents a balance between responsiveness and stability in controlling the turn and has been designed to have somewhat higher proportional gain than would be necessary if tight turns were not needed. The corresponding heading angle is clearly shown in Figure 4. The oscillatory part of the yaw rate during the period 35 - 45 seconds is possibly generated by inertial cross-coupling that potentially exists between the pitch / yaw modes although nominally assumed to be negligible. Later experiments with tighter control suppressed this phenomenon to a large degree. This is evidence that high gain robust controllers are indeed needed for these separate autopilots in compensating for the induced mode coupling.

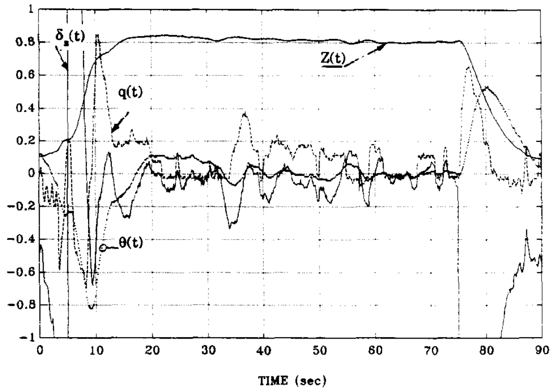

Figure 5

$\delta_s(t)$, q(t), θ(t), and Z(t) versus Time. [Run 7-8-91-5]; Oval Track Run; Combined Diving, Steering, and Speed Control; Standard Control Laws; Scaled as $\delta_s(t)/0.4$ (rad), q(t)/0.08 (rad/sec), θ(t)/0.25 (rad), Z(t)/2.5 (ft.)

Diving Response

The diving response is indicated in Figure 5. There is an initial flurry of dive plane control action as the vehicle accelerates to speed and goes below the water surface. The initial launch is on the surface and the transition to depth is smooth but initially the vehicle speed is slow and the control is less effective than at the nominal running speed about 2 ft/sec. (0.61 meters/sec.). The pitch rate and angle are shown. The pitch angle reaches 0.2 radians then is reduced quickly and the nominal depth of 2 feet (0.61 meters) is achieved. At the end of the test run, the mission calls for a depth change to surface as indicated at the time of 75 seconds. The pitch control law for which the results are shown was a three state proportional law without the nonlinear term, given by,

$$\delta_s(t) = K_q q(t)+K_\theta \theta(t)+K_z\,[Z(t)-Z_{com}(t)]$$

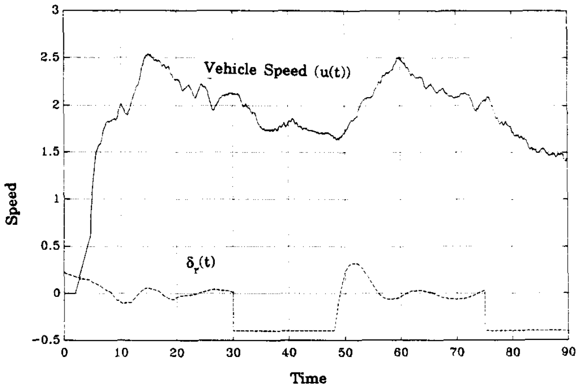

Figure 6

Vehicle Speed (u(t)) versus Time. [Run 7-8-91-5]; Oval Track Run; Combined Diving, Steering, and Speed Control; Standard Control Laws

Speed Response

The vehicle speed control was initially provided by a PI control law (the Sliding Mode version is now implemented) including an integral term, where with abuse of notation,

$$n_{com}(t) = (n_0/u_0)u_{com}(t) + K_p e(t) + K_i \sum_{i=1}^{10} e_{k-i}$$

the integral term as a sum over the last ten points was present to help in maintaining speed during the turn where the large centrifugal force and the added plane drag causes significant loss of speed. The response in Figure 6 shows the output from the paddle wheel sensor indicating good acceleration followed by an overshoot at 2.5 ft/sec with a controlled speed reduction during the period 20 - 30 seconds. The speed reduction during 30 - 50 seconds is the effect of the added drag terms which would be much larger without the corresponding increase in propeller speed not shown. (Shown in later runs). The speed gain during the period 50 - 70 is the result of the vehicle coming out of the turn and the speed controller taking over in stabilizing to the set point of 2 ft./sec. (0.61 meters /sec.).

The path obtained by dead reckoning using the paddle wheel speed sensor and the heading gyro output but neglecting side slip errors is given in Figure 7. Recognizing the limitations of the accuracy of this navigation scheme, we have found the results sufficient to guide the vehicle without a collision with the pool walls.

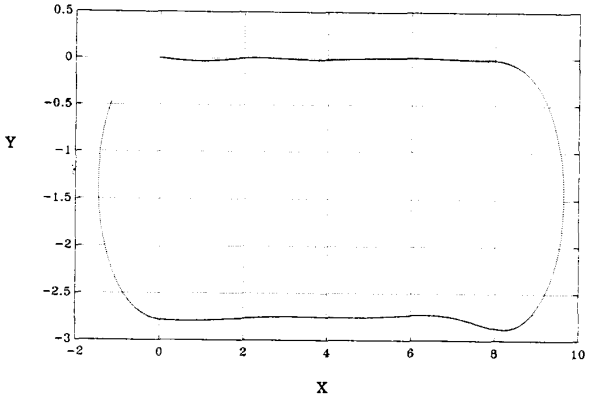

Figure 7

Vehicle Path versus Time. [Run 7-8-91-5]; Oval Track Run; Combined Diving, Steering, and Speed Control; Standard Control Laws; X and Y expressed in Vehicle Lengths

Way point Guidance by Line of Sight

Vehicle autonomous guidance is most simply accomplished by a heading command to the vehicle's steering system to approach the line of sight between the present position of the vehicle and the way point to be reached. In missile guidance this is related to 'proportional navigation'. The difference in guiding AUV's is that the vehicle response is slow compared to the rates of change in command unless the way point is many vehicle lengths away. Separation of guidance and autopilot functions may not always produce stable results underwater. Notwithstanding, we define the line of sight (LOS) to be the horizontal plane angle given by,

$$\psi_{com} = \tan^{-1}\left[\frac{(Y_k - Y(t))}{X_k - X(t)}\right]$$

in which the $[X_k, Y_k]$ are way points stored in the vehicle's mission planner. Care must be taken to keep the proper quadrant in mind when programming the guidance law. The decision as to whether the way point has been reached is made on the basis of whether the vehicle lies within a 'ball of acceptability', ρ_0 defined around the particular way point. Namely, if, for some distance, ρ_0, an acceptable zone around the way point, $[X_k(t), Y_k(t), Z_k(t)]$, the vehicle location [X(t), Y(t), Z(t)] are such that,

$$\rho^2(t) = [Y_k-Y(t)]^2+[X_k-X(t)]^2+\lambda[Z_k-Z(t)]^2<\rho_0^2 \qquad 0<\lambda<1$$

the above condition triggers the selection of the next way point. If, on the other hand, the condition that dρ/dt goes from negative to positive without the above being met then the way point is not reached. At this juncture, the guidance law must contain logic that will either hold the current way point, directing the vehicle to circle, or the next way point could be entered, depending on a mission planning decision. λ is a parameter relating to the importance of including depth dimension in the acquisition of the way point. In this section, vehicle way point control is examined in experiment using the autopilots described above combined with the LOS guidance. The assumption is made that vehicle speed control is obtained from a separate speed command for each separate leg of a transit mission, although that could be accomplished also by an on line speed command as a function of distance to go and the time to go if a desired time is also associated with each way point. The ability of the LOS method to acquire way points is illustrated by the series of results given in Figures 8 - 11. In Figure 8, the steering response variables are shown with rather oscillatory swings that are characteristic of commands changing as way points are reached and subsequent points entered into the controller. The diving performance is given in Figure 9 where a commanded depth of 2 ft again was used. Figure 10 shows the speed controller response as the vehicle is accelerated and slowed by the turning activity. In Figure 10, the propeller speed command is shown as well as the vehicle speed response from the paddle wheel sensor. Separate experiments, not described here, have determined that the response of the inner loop for the control of motor speed to motor speed commands is fast and has negligible lags in this application. Figure 11 shows that each way point was acquired with excellent precision even though the global locations of those way points may not have been uncertain. In other words, the autopilot functions drove the vehicle to the locations that the vehicle 'thought' it had to meet.

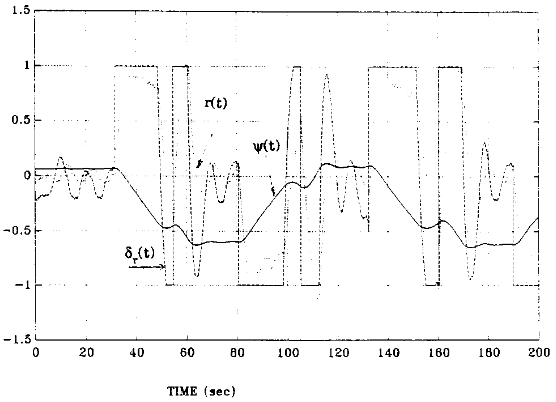

Figure 8

Figure Eight Run: Vehicle Steering Response. Combined Diving, Steering, Speed Control. Standard Control Laws. $\delta_r(t)$, r(t), $\psi(t)$, versus Time. Scaled Valuesas in Figure 4

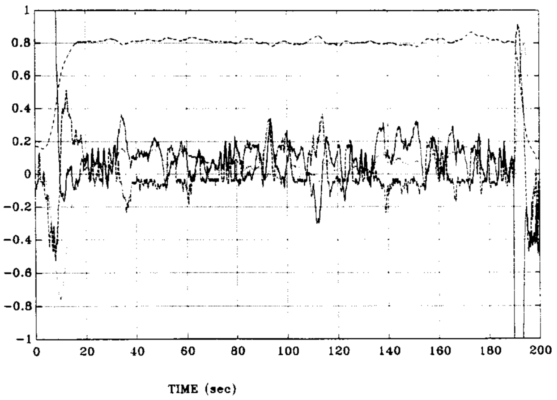

Figure 9

Figure Eight Run: Vehicle Diving Response. Combined Diving, Steering, Speed Control. Standard Control Laws. $\delta_s(t)$, q(t), $\theta(t)$, and Z(t) versus Time. Scaled Values as in Figure 5

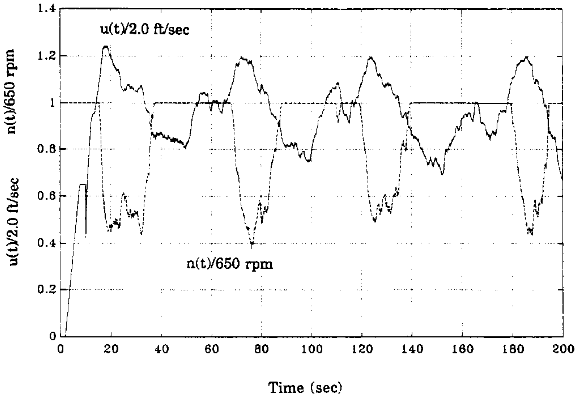

Figure 10

Figure Eight Run: Vehicle Speed Response. Combined Diving, Steering, Speed Control. Standard Control Laws. u(t)/2.0, n(t)/650 rpm versus Time. Scaled Values as in Figure 6

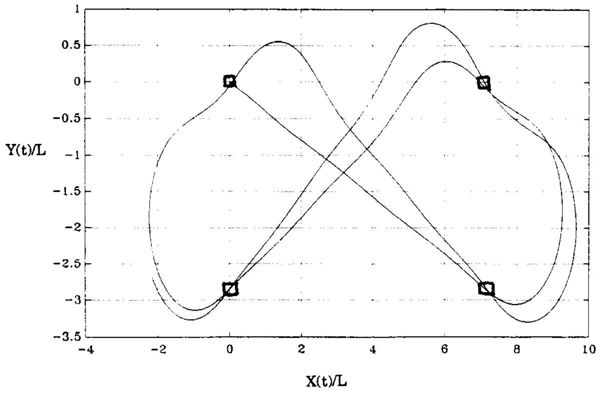

Figure 11

Figure Eight Run: Vehicle Path Response. Combined Diving, Steering, Speed Control. Standard Control Laws. Way Points Shown. Y(t)/L versus X(t)/L.

What we see is a vehicle that is capable of tight turns; its steering and diving systems are stable under conditions of combined maneuvering at speeds that are changing; and planned paths, in terms of way points, can be followed with precision consistent with the limits of the vehicle's turning capability.

Sliding Mode Control Compared

The performance of sliding mode control has been compared in a series of runs using the same oval path as in the first series with the steering control law,

$$\delta_r(t)=k_{22}r(t)+k_{23}\dot{r}_{com}(t)+\eta_r\tanh(\sigma_2(t)/\phi_2)$$

where $\sigma_2(t)$ is the sliding surface.

In Figures 12a and 12b, three controller's results are superimposed with the rudder responses shown in Figure 12a, and the corresponding yaw rate responses shown in Figure 12b. Each shows the effect of increasing nonlinear gains. The effect on the yaw rate response out of the turn is not as strong as thought and increasing gain appears to increase the levels of activity on the control surfaces. However, the overall response is very rapid and much improved over the initial PD controller. It is believed that the sliding mode control is easy to implement and even easier to tune in the field as only one parameter needs to be modified to adjust the speed of the controller from slow to fast and stability is not compromised.

Figure 13 is provided from the same series of runs to illustrate that the propulsion control is disturbed by the continual turning and added drag forces being applied. A careful examination of Figure 13 at the time of 20 seconds shows that the sharp increase in vehicle speed coincides with the control surface changing from a positive to a negative value quickly. Whenever a control surface is brought to a null position there is an attendant change in drag that occurs almost instantaneously. The results reveal that a considerable amount of oscillatory changes occur. It may be evidence of the dynamics and nature of the propeller thrust response having lags. This is the subject of further investigation.

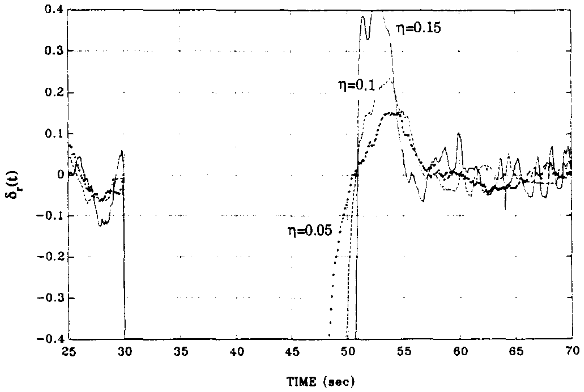

Figure 12a

Oval Track Run; Vehicle Rudder versus Time; Sliding Mode Steering Control; Varying Non-Linear Gains η=[0.05, 0.1, 0.15]

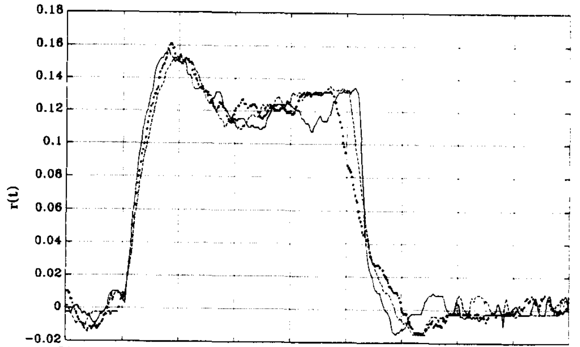

Figure 12b

Oval Track Run; Vehicle Yaw Rate versus Time; Sliding Mode Steering Control; Varying Non-Linear Gains η=[0.05, 0.1, 0.15]

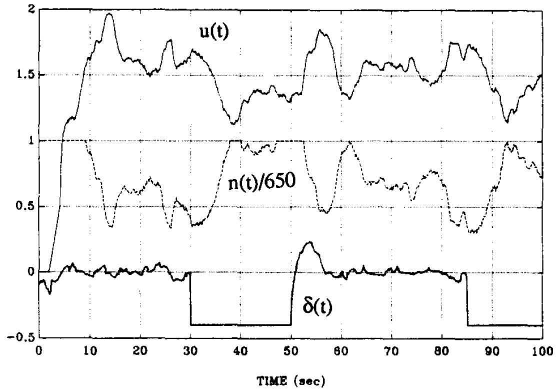

Figure 13

Oval Track Run: Speed Changes [u(t) in ft/sec.], Normalized Propellor Speed [n(t)/650] and Rudder [δ(t) rad] versus Time.

Bottom Following

Incorporation of a downward looking sonar (Datasonics PSA 900) into the depth control system has allowed a test series for altitude control and also using the vehicle depth sensor, a determination of the water column height around the pool. For the basic oval loop, Figure 14 shows the result of the control to a fixed height above bottom, and the attendant estimation of the total.water depth. Since the water depth in the pool is known at any X location, the result is compared with that known profile.

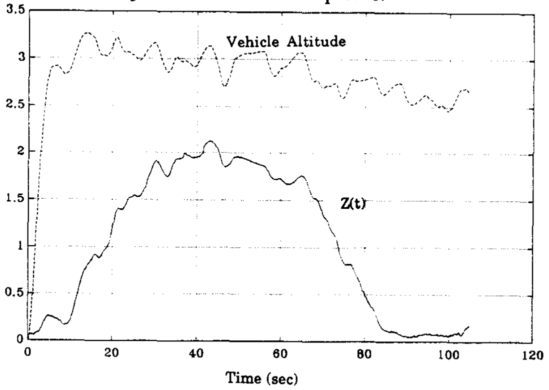

Figure 14
Oval Track Run; Bottom Following; Altitude and Depth versus Time; Sliding Mode Steering Control;

Obstacle Avoidance

While the **Obstacle Avoidance Decision Maker** is a system that has yet to be defined to its fullest extent at the time of writing, The vehicle has four sonar ranging systems on board that have been providing mapping data to the pool sidewalls. One of the most important obstacle avoidance issues is to prevent the vehicle from running into a solid object in its path. The use of a forward looking sonar to provide range to such an object has been demonstrated in pool tests where a limit of 25 feet has been set after which a hard turn to the starboard is triggered. The quality of the range signals from the Datasonics PSA 900 200 KHz.sonar is shown in Figure 15 where it has been clearly shown that an obstacle avoidance maneuver was triggered at the correct time to turn the vehicle away from the pool end wall.

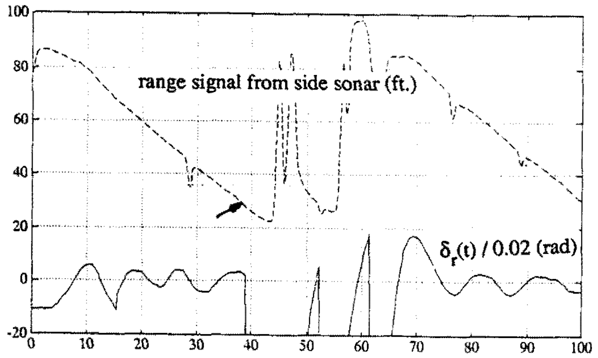

Figure 15

Oval Track Run; Wall Avoidance; Forward Sonar Range and Rudder versus Time;

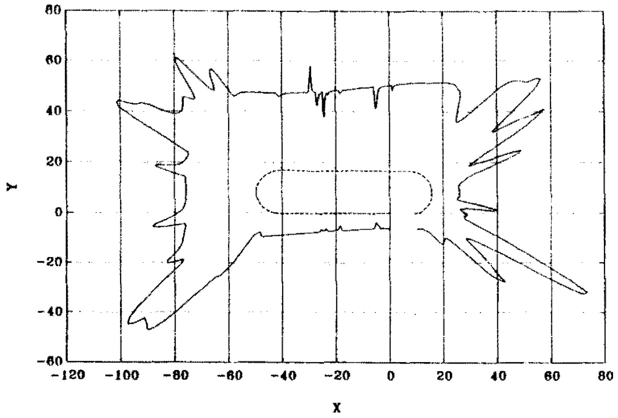

Figure 16

Oval Track Run; Wall Shape from Raw Side Sonar Range Data; Y versus X units in Ft.

Using the ranging data from both forward and side looking transducers we show in Figure 16 that the dead reckoning system can be used to reconstruct the shape of the pool side walls with reasonable accuracy. Refinements by taking side slip into account have provided additional accuracy and in fact can be used to calibrate a side slip observer for vehicle navigational enhancement.

POST MISSION DATA ANALYSIS

Smoothing of sonar data and the use of Graphics post processing, has been described by Floyd, et. al. (1991), Brutzman, et. al., (1992a) in this proceedings, and Brutzman, (1992b) and will not be further elaborated here.

CONCLUSION

Much more work needs to done in this community to continue with sufficient duplication, but also with sufficiently diverse opinion to illuminate the range and trade-offs of possible structures and technology, hardware and software, needed for precise, reliable control of AUVs in the future.

REFERENCES

Albus, J. S., (1988) "System Description and Design Architecture for Multiple Autonomous Undersea Vehicles" *NIST Technical Note No. 1251*, U.S. Department of Commerce National Institute of Standards and Technology, Sept. 1988

Bellingham, J., Consi, T., Beaton, R., (1990a) "Keeping Layered Control Simple", *Proc. Symposium on Autonomous Underwater Vehicle Technology* June 5-6 1990 Washington DC. Published as IEEE Catalog No. 90CH2856-3, pp 3-9

Bellingham, J., Consi, T., (1990b) "State Configured Layered Control" *Mobile Robots for Subsea Environments*, Proceedings of the 1st NSF IARP Workshop, Monterey Bay Aquarium Research Institute, California, October 23-26, 1990, pp 75 - 80

Brooks, R., (1986) "A Robust Layered Control System for a Mobile Robot" *IEEE Journal of Robotics and Automation*, Vol. 2, 1986

Brutzman, LCDR Donald P., Kanayama, Y., and Zyda, M.J., (1992a) "Integrated Simulation for Rapid Development of Autonomous Underwater Vehicles," presented at IEEE Oceanic Engineering Society *Symposium on Autonomous Underwater Vehicles, AUV-92* Washington D.C., June2-3,1992.

Brutzman, LCDR Donald P., (1992b) "NPSAUV Integrated Simulator", *Master's Thesis, Naval Postgraduate School,* Monterey, California, March1992.

Chappell, S. G., (1987) "A Blackboard Based System for Context Sensitive Mission Planning in an Autonomous Vehicle" *Proceedings of the 5th International Symposium on Unmanned Untethered Submersible Technology*, University of New Hampshire, June 1987 Vol.2 pp 467-476

Floyd, C. A., Kanayama, Y., and Magrino, C., (1991) "Underwater Obstacle Recognition using a Low-Resolution Sonar," *Proceedings of the Seventh International Symposium on Unmanned Untethered Submersible Technology,* University of New Hampshire, Durham, New Hampshire, September 1991, pp.309-327.

Healey, A.J., et. al. (1990) "Mission Planning, Execution, and Data Analysis for the NPS AUV II Autonomous Underwater Vehicle" *Mobile Robots for Subsea Environments*, Proceedings of the 1st NSF IARP Workshop, Monterey Bay Aquarium Research Institute, California, October 23-26, 1990, pp 177-186

Healey, A. J., Papoulias, F. A. Lienard, D., (1990) "Multivariable Sliding Mode Control for Autonomous Diving and Steering of Unmanned Underwater Vehicles " *Modeling and Control of Marine Craft*, Elsevier Publications,1990

Healey, A.J., Good, M., (1992) "The NPS AUV II Autonomous Underwater Vehicle Testbed: Design and Experimental Verification" *Naval Engineers Journal*, ASNE , May 1992 issue.

Mayer, R., et. al. (1987) "Situation based Control Architecture for an AUV" *Proceedings of the 5th International Symposium on Unmanned Untethered Submersible Technology*, University of New Hampshire, June 1987, Vol. 2 pp 430-443

Papoulias, F. A., Healey, A. J. (1990) "Path Tracking of Surface Ships Using Multivariable Sliding Mode Control" *Proceedings of the 9th Ship Control Systems Symposium*, Washington D.C.Sept 10-14 1990

Proc. Symposium on Autonomous Underwater Vehicle Technology June 5-6 1990 Washington DC. Published as IEEE Catalog No. 90CH2856-3

Proceedings of the 5th International Symposium on Unmanned Untethered Submersible Technology, University of New Hampshire, June 1987

Proceedings of the 6th International Symposium on Unmanned Untethered Submersible Technology, University of New Hampshire, June 1989

Proceedings of the 7h International Symposium on Unmanned Untethered Submersible Technology, University of New Hampshire, September 1991

Ura, T., (1990) " Development of AUV 'PTEROA'" *Mobile Robots for Subsea Environments*, Proceedings of the 1st NSF IARP Workshop, Monterey Bay Aquarium Research Institute, California, October 23-26, 1990, pp 195 - 200