# AUV Bathymetric Mapping
# Depth Planning for Bottom Following
# Splice Linear Programming Algorithm

Lawrence A. M. Bush
Autonomous Solutions, Inc.†
990 North 8000 West
Mendon, UT 84325
Email: BushL2@alum.mit.edu

Lars Blackmore
Space Exploration Technologies†
1 Rocket Road
Hawthorne, CA 90250
Email: lars.blackmore@spacex.com

Brian C. Williams
Massachusetts Institute of Technology
77 Massachusetts Avenue
Cambridge, MA 02139
Email: williams@csail.mit.edu

***Abstract*—We collaborated with the Monterey Bay Aquarium Research Institute (MBARI) to improve the depth control algorithm used by the Dorado class autonomous underwater vehicle (AUV) in conducting bathymetric surveys and other remote sensing tasks. The algorithm enables better bottom following by planning for the depth profile that follows the desired depth best, while pulling up safely for bathymetry. The algorithm allows the AUV to operate closer to the sea floor and in more variable submarine conditions. Deployment tests demonstrated improved AUV performance in a bathymetrically complex area three miles off shore in Monterey Bay, and highlighted areas where further research and development can enhance AUV operation.**

## I. Introduction

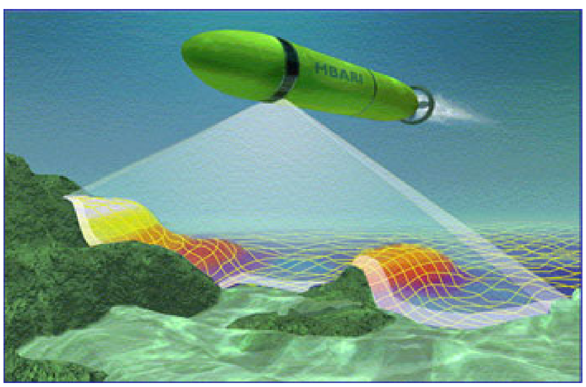


Fig. 1. Dorado class autonomous underwater vehicle conducts scientific underwater missions such as mapping the sea floor shown.

### A. AUV Bathymetry Control

Dorado [7] AUVs record bathymetric data by descending to depth and tracking position above the sea floor via multiple navigation sensors. The AUVs default depth planner projects 100 meters ahead and plots a course straight toward the highest point on the horizon. The AUV tracks its position throughout the mapping mission by maintaining ground-lock, which is effective up to approximately 50 meters above the ocean floor. The AUV often loses ground lock in steep or variable terrain. Aquadynamic perturbations can also prevent the AUV controller from accurately navigating to target depth. When it loses ground-lock, the AUV must default to its inertial navigation system, which offers less accuracy in mapping. Likewise, a target depth closer than 40 meters engenders an unacceptable risk of collision with the ocean floor in steep or variable terrain. We developed a prototype depth control algorithm that uses simulation and spliced linear programming to plan vehicle maneuvers. The modified strategy enhanced the AUVs ability to maintain ground-lock in variable terrain and thus improved its mapping capabilities.

AUV algorithms use general, coarse-scale information about a survey area to plot navigational courses that can then be dynamically reoptimized with close-range input. The projectional component of the algorithm, referred to as the planner, assumes a linearized model of AUV dynamics to plan a course constrained by operational parameters such as pitch and elevator angles. It processes a sea floor transect of minimum, maximum and optimal depth specifications to generate a series of depth waypoints that minimize the average positional deviation from optimal depth according to operational constraints (e.g., with respect to pitch and elevator angles).

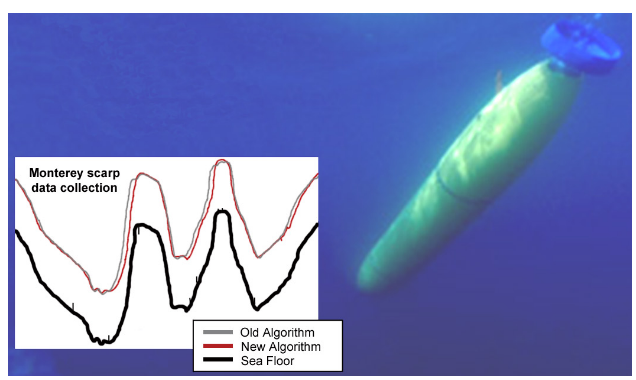


Fig. 2. The inset graph shows the MBARI Dorado class AUV tracking the target depth better when controlled by our new algorithm (red) versus the old algorithm (grey). The black line shows the ocean floor.

† Research performed at the Massachusetts Institute of Technology

978-1-5090-1537-5/16/$31.00 ©2016 IEEE

Our algorithm projects a series of depth waypoints plotted at 5 second intervals over successive, predefined windows ($1,000$ seconds) of the transect. Once populated with waypoints, the window or interval becomes a long term plan that can then be dynamically modified using input from the AUV dynamics controller. (The dynamics controller provides input to the articulated ring-wing and ducted thruster actuators on the AUV.) The algorithm minimizes depth error by penalizing optimal depth deviation. The planner also dampens rapidly changing (chattering) control inputs by penalizing average pitch-rate where necessary. The pitch-rate penalty is a design parameter, which depends on the user-assigned weighting of a smooth control plan.

Path construction requires concatenation of the optimized long term plans ($1,000$ second intervals). Given the projectional nature of these plans, however, the optimality/accuracy of the path declines with time as it is plotted at the outermost (future) points of the plan. To reconcile sequential intervals with potentially inaccurate margins, the algorithm discards the outermost 25% of the current interval, stitching only the most immediate 75% to the next interval. This method, referred to as splicing, effectively reconciled final and initial AUV states and thus enhanced the compatibility of sequential planning intervals.

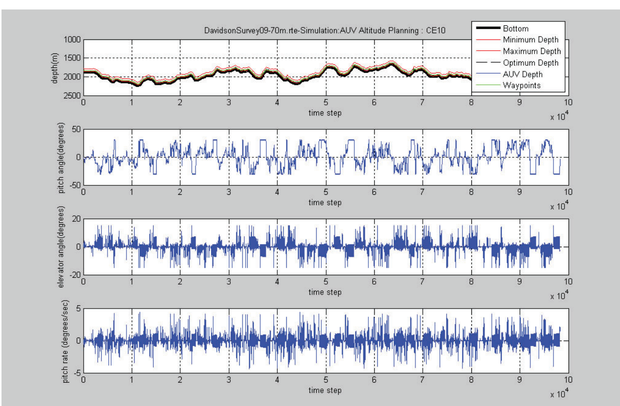


Fig. 3. We ran our spliced linear programming algorithm on Davidson Seamount data, resampled at 5-second intervals. The solver produces a set of target waypoints for the controller to follow. We then simulate results of the entire transect using the AUV's actual, shorter (.2 second) dynamics control interval. The dynamics controller provides input to the articulated ring-wing and ducted thruster actuators. The simulation accepts the 5 second interval target waypoints for the controller to follow. The inputs are duplicated to accommodate the higher frequency. The top graph shows the sea floor, minimum, maximum and optimal depth, the outputted waypoints as well as the simulated AUV depth. The three lower graphs show the pitch angle, elevator angle and pitch-rate. The simulation results demonstrate that our 5 second planning interval is sufficiently short to produce good vehicle guidance.

The algorithm was successfully tested in both simulated and real world environments. Tests allowed us to compare AUV pathways generated by the algorithm described above with paths generated by an earlier generation (pre-existing) algorithm. Our algorithm improved responses to steep inclines along the transect and tracked actual depth more accurately than the previous algorithm. The results shown in Figure 17 aggregate to a 42% mean reduction and a 76% mean-squared reduction in deviations from the target depth over the course of the mission, demonstrating the effectiveness of these control system modifications. The deployed depth planning code is available at [4]. A users guide is also provided [5].

The following sections describe operational parameters and capabilities of the Dorado AUV, algorithm formulation and testing results.

## II. Dorado AUV System Design

The MBARI developed and operated Mapping AUV is a 0.53 meter (21 inch) diameter, $1,500$ pound variable length autonomous vehicle constructed from multiple sections. The minimal configuration includes a nose and tail section. The mapping AUV nose contains batteries and conductivity, temperature, and pressure depth (CTD) sensors. The tail section includes control, communications, navigation, power, and propulsion components. Propulsion is provided by a single articulated propeller, which resides inside a circular duct and serves as the vehicles sole control surface. Buoyancy is provided by blocks of foam.

### A. Mapping

A third midbody section contains a Reson 200 kHz multibeam mapping sonar providing 1 meter resolution bathymetry at 50 meters altitude.

### B. Localization

At the ocean surface, an AUV can acquire precise absolute GPS-based location information. However, in order to navigate complex seafloor environments the AUV uses navigation and attitude data from a Kearfott Seadevil laser-ring-gyro based inertial navigation system (INS). When available, the AUV also integrates data from an acoustic 300 kHz Doppler Velocity Log (DVL), which measures the vehicle velocity over the seafloor. Without DVL or GPS support, location errors accumulate quickly, at multiple kilometers per hour. With continuous DVL seafloor observations, navigation error is 0.05% of distance traveled at a representative speed of 3 knots. Reliable DVL seafloor observations are possible for altitudes less than 130 meters. Consequently, it is very important to maintain close proximity to the seafloor throughout a mission. If follows that a vehicle should also be launched in shallow water in order for the DVL to sense the bottom while GPS is still available.

### C. Processing

The main vehicle computer, housed in a 17 inch glass sphere in the tail section, using an object-oriented software framework, running with the QNX operating system on a PC-104 computer, provides guidance and control functions, data logging, and sensor interfacing. Mapping data is processed using MB-System [6]. Scripts containing waypoint following and safety behaviors are downloaded to the vehicle before the mission and executed during the mission.

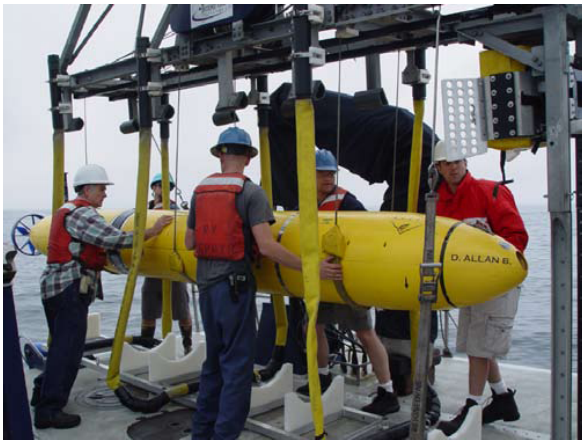


Fig. 4. MBARI Dorado class mapping AUV being prepared for deployment via an over-the-stern J-frame crane [7].

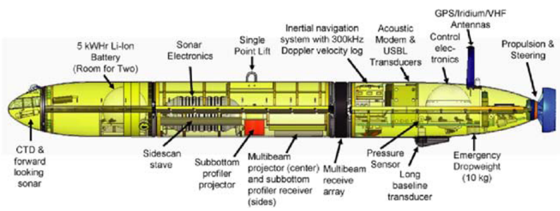


Fig. 5. Dorado class mapping AUV system component diagram [7].

## III. Existing Algorithm

MBARI has a multi-beam side-scan sonar for bathymetric mapping mounted on a Bluefin AUV. The vehicle is currently capable of making pre-scripted 8-hour missions. The missions are planned using MB-System Software. Two of the steps in the planning process involve laying out the path and computing a depth profile. In the first step, the user indicates the area to map. The software lays out a lawnmower pattern with a couple of crossings. The crossings are done so that the resulting data can be associated more accurately. Once the path is constructed, it is output as a set of major and minor waypoints. The major waypoints are all that is necessary for defining the path. However, the minor waypoints are output in order to allow for depth planning. Each waypoint consists of a latitude, longitude and topology (seafloor depth). the current depth planner makes a very approximate plan which includes a look-ahead of about 100 meters. From a given position, the depth planner looks ahead 100 meters, and determines the highest point in that look-ahead. It then creates a linear path (in depth) between the current point (50 meter altitude) and the next highest point (at a 50 meter altitude). This linear path essentially travels along the local bathymetric peaks. So, if the bathymetry is very flat, it will maintain a 50-meter altitude quite closely, but if the bathymetry is very hilly, it will often be well above the 50 meter altitude. This is a very approximate planning method. Therefore, it must be very conservative in order to be safe.

There is a clear need for a more accurate depth plan for which the AUV is still dynamically capable of executing.

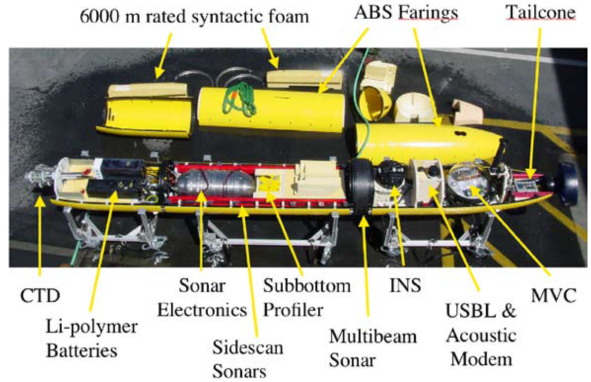


Fig. 6. Dorado class mapping AUV interior [7].

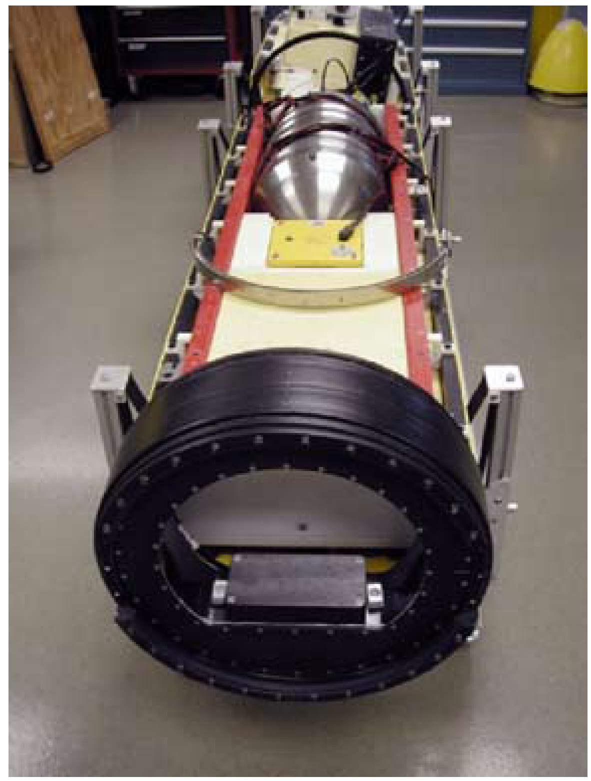

Fig. 7. Dorado class mapping AUV interior [7].

## IV. Algorithm Summary

We have created a prototype dynamics-based optimized depth waypoint planner for the MBARI AUV. This planner takes as its input:

1) A linearized model of the AUV dynamics.
2) Constraints on the AUVs performance, such as pitch angle and elevator angle constraints.
3) A profile of the sea floor along a transect.
4) Specifications of the minimum, maximum and optimal altitudes above the sea floor for the AUV.

Fig. 8. Kearfott Seadevil inertial navigation system.[7]

The planner provides a series of depth waypoints along the transect that minimize the average deviation of the AUV from its optimal altitude (altitude error), while satisying hard constraints (such as pitch angle). In practice, as well as penalizing deviation from the optimal altitude, we also penalize average pitch-rate magnitude. Without this, the optimal planner will use quickly changing (chattering) control inputs to achieve as little altitude error as possible. How much to penalize the pitch-rate is a design parameter; this should be chosen depending on the importance of a smooth control plan.

Planning is carried out with respect to discrete-time AUV dynamics. A full plan takes place over several hours, whereas the time constants of the AUV dynamics are on the order of seconds. Generating a full depth waypoint plan therefore involves optimization over a very large number of decision variables. To make this problem tractable, we do the following: 1) For waypoint planning, we use a time interval of 5s in the AUV dynamics. This is chosen to be as large as possible while still small enough to capture the AUV dynamics adequately. After a plan has been generated, more detailed simulation should be carried out to validate the plan, as discussed in Section III. 2) We plan depth waypoints for 1000s segments and splice the solutions together to give a full plan. The splicing is carried out to ensure a smooth transition between segments, and to ensure that one segment is not planned in such a way that planning the next segment is infeasible.

### A. Approach

The AUV collects bathymetric data using a multibeam sidescan sonar sensor and navigates using a gyro-based inertial navigation system integrated with a Doppler velocity sonar. The AUV deploys from a ship, descends in a spiral path and navigates along the ocean floor using GPS.

As it approaches the ocean floor, the AUV tracks its position via multiple navigation sonars, one of which monitors the relative location of the ocean floor for accurate positioning. This sensor, the Doppler velocity sonar, must maintain ground-lock in order to provide reliable velocity estimates. When ground-lock is lost, the AUV defaults to a less accurate inertial navigation system.

## V. Steps

Briefly, the AUV produces a bathymetric map by descending to bathyal depths and by tracking its position above the ocean floor via multiple navigation sensors. As long as the sensors allow the AUV to maintain ground- lock, which occurs at about 50 meters above the ocean floor, the AUV can accurately track its position. When it loses ground-lock, the AUV must default to its inertial navigation system, which compromises accuracy. We implemented a spliced linear programming algorithm that plans vehicle maneuvers based on more accurate information concerning the vehicle's depth and surroundings. The modified strategy enhanced the AUV's ability to maintain ground-lock in variable terrain and thus improved its mapping capabilities.

## VI. Implementation of Our Mission Planning Algorithm

We created a prototype dynamics-based optimized depth waypoint planner for the MBARI AUV. This planner takes: 1) a linearized model of the AUV dynamics, 2) constraints on the AUVs performance, such as pitch angle and elevator angle constraints, 3) a profile of the sea floor along a transect, 4) and minimum, maximum and optimal depth specifications relative to the sea floor.

### A. Dynamics

A dynamical model for the exchangeable mid-body articulated ring-wing partially ducted thruster AUV were developed in [18]. The AUV hull shape was derived from the Series $58$ Model $4154$ Gertler polynomial [10]. We determined the system matrix, control matrix, output matrix and feed-forward matrix from [18]. State variables include depth, depth rate, pitch rate, depth, and pitch angle. The nearly symmetrical AUV is controlled entirely through the double gimballed ring-wing partially ducted thruster. Two stepper motor produce rudder and elevator control up to $\pm 15°$. The output vector includes the state variables plus elevator angle. We adopted a nominal close to straight and level trajectory assumption so that we can use the linearized dynamical model provided in [18].

### B. Linear Program

The planner provides a series of depth waypoints along the transect that minimize the average positional deviation from the AUV's optimal depth, while adhering to hard constraints such as pitch angle. In addition to penalizing optimal depth deviation, we also penalize average pitch rate magnitude, which dampens the control inputs. Without the pitch rate penalty, the optimal planner will use quickly changing ('chattering') control inputs to minimize depth error. The pitch rate penalty is a design parameter, which depends on the relative importance of a smooth control plan.

Planning is carried out with respect to discrete-time AUV dynamics. A full plan takes place over several hours, whereas the time constants of the AUV dynamics are on the order of seconds. Generating a full depth waypoint plan therefore involves optimization over a very large number of decision variables. The following specifications make this problem tractable:

1) For waypoint planning, we use a 5-second time interval. We chose this interval to be as large as possible while still small enough to capture the AUV dynamics adequately. After a plan has been generated, a more detailed simulation with a finer time resolution is carried out to validate the plan.
2) We plan depth waypoints for 1,000-second segments and splice the solutions together to give a full plan. The splicing ensures a smooth transition between segments, and ensures that one segment is not planned in such a way that planning the next segment is infeasible.

### C. Linear Program Setup

We minimize $f'x$ subject to $Ax <= b$, where $A$ is a composite of dynamical limitations (e.g. elevator and pitch limits) and mission constraints (e.g. minimum & maximum depths).

Due to the large number of waypoints, we must solve the problem in parts. We must constrain a subset to start in the same dynamic state that the previous waypoint set finished in, less a small lookahead in terms of waypoints to make sure that the AUV remains in a safe state.

We also include the equality constraints $A_{eq} \leq b_{eq}$ where $A_{eq}$ represents the system dynamical constraints, and $b_{eq}$ represents the initial state (pitch angle, pitch rate, etc).

$$\begin{array}{c} min \quad f'x \\ s.t. \quad Ax \leq b \\ A_{eq} = b_{eq} \end{array}$$

$f$ represents control effort penalty (e.g. pitch rate) and decision variable $x$ is a concatenation of four temporal sequential variables.

$$x = \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ ... \\ x_T \\ u_0 \\ u_2 \\ ... \\ u_T - 1 \\ d_0 \\ d_1 \\ ... \\ d_T \\ f_1 \\ f_2 \\ ... \\ f_T \end{pmatrix}$$

where $x_t$ is a column vector describing the state at time $t$, $u_t$ is a column vector describing the control input at time $t$ (a depth waypoint), $d_t$ is the magnitude of the depth error at time $t$, and $f_t$ is the magnitude of the pitch rate at time $t$.

### D. Splicing Method

Once we solve for the first interval, we used the final AUV state as the initial AUV state for the next segment. Unfortunately, the first interval was optimized without regard for compatibility with the second interval, potentially leaving the AUV in a difficult final state. In order to accommodate the subsequent interval we used only 75% of the planning interval, discarding the latter 25% and beginning the next interval according to this modified final AUV state. We refer to this procedure as splicing. We reach end the interval in a safe state because we have planned through the potential risks that state. We also re-plan part of our transect in this procedure but in an efficient and effective manner. The output is shown in Figure 10.

## VII. Verification & Validation

### A. Simulation

Figures 10 through 13 show simulated test results from the Davidson 2007 Survey. We developed, simulated and tested our algorithm using the Davidson Survey data. Figure 9 shows the location of the survey, off the coast of California near Big Sur.

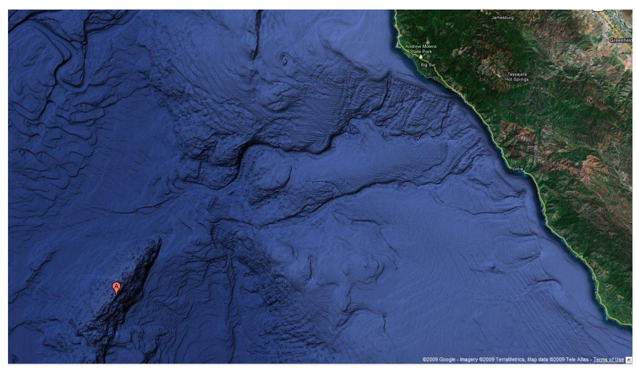

Fig. 9. The red marker shows the location of the 2007 Davidson Seamount survey. Davidson Seamount lies west off the coast of California near Big Sur. We used the survey data to develop and test our algorithm in simulation.

We ran our spliced linear programming algorithm on the Davidson Seamount data. The data was first resampled at 5 second intervals. We then ran our linear programming solver on the first 1,000-second interval. The solver takes as input the initial AUV state, the desired depth, depth constraints, linearized vehicle dynamics and the respective vehicle dynamics controller. The solver produces a set of target waypoints for the controller to follow.

*a) Simulation Results:* After running our planning algorithm on the 5-second (sampling interval) data, we simulated the entire transect using the shorter (.2 second) interval used by the actual dynamics controller. The dynamics controller provides the input to the articulated ring-wing and ducted thruster actuators. The simulation accepts the 5-second interval target waypoints for the controller to follow. The inputs

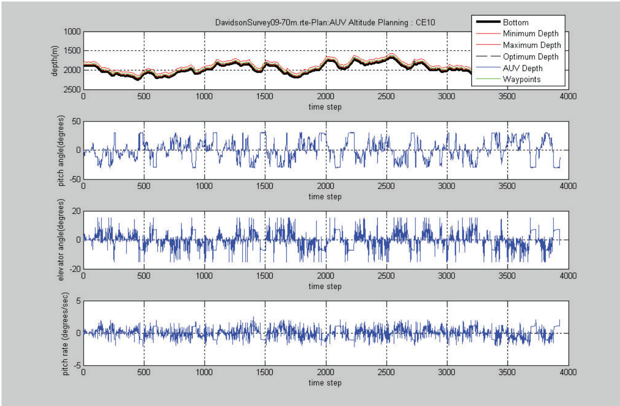


Fig. 10. We ran our spliced linear programming algorithm on Davidson Seamount data, resampled at 5-second intervals. The solver produces a set of target waypoints for the controller to follow. The top graph shows target waypoints, sea floor depth, minimum, maximum and optimal depth as well as simulated AUV depth. The three lower graphs show the pitch angle, elevator angle and pitch-rate respectively.

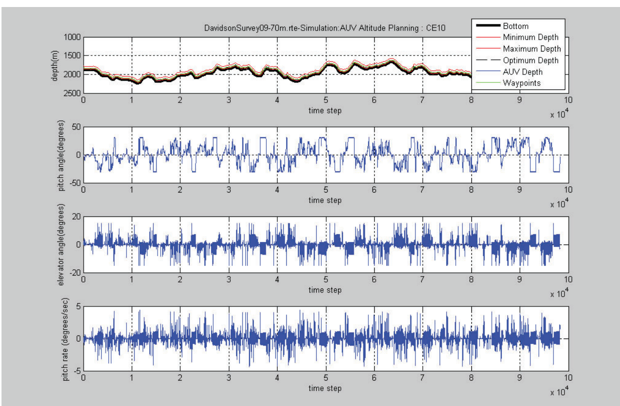


Fig. 11. Simulation results of the entire transect using the AUVs actual, shorter (.2 second) dynamics controller. The dynamics controller provides input to the articulated ring-wing and ducted thruster actuators. The simulation accepts the 5 second interval target waypoints for the controller to follow. The inputs are duplicated to accommodate the higher frequency. The top graph shows the sea floor, minimum, maximum and optimal depth, the outputted waypoints as well as the simulated AUV depth. The three other graphs show the pitch angle, elevator angle and pitch-rate. The simulation results demonstrate that our 5 second planning interval is sufficiently short to produce good vehicle guidance.

are duplicated to accommodate the higher frequency. The simulated results are shown in Figure 11. The AUV initially follows a path between the minimum and maximum altitude, but then navigates along the optimal depth, except when pitch angle constraints make this impossible. Overall, the simulation results demonstrate that our 5-second planning planning interval is sufficiently short to produce good vehicle guidance and the algorithm generates a path that makes full use of the pitch angle allowance in order to minimize the depth error. Figure B-5 shows the main simulation results in a separate larger format, with the sea floor, minimum, maximum and optimal depth, the outputted waypoints, and the simulated AUV depth. Figure 13 zooms enlarges a segment of the simulation results

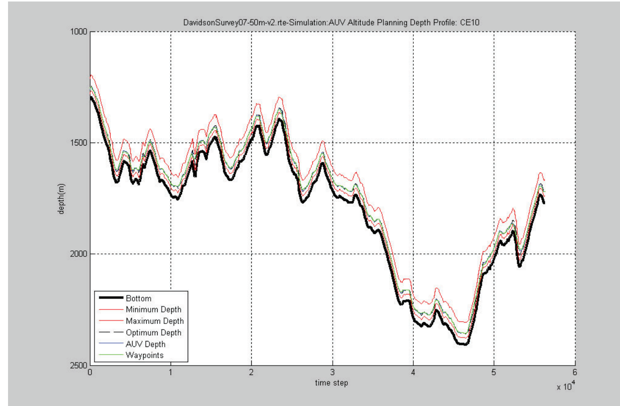


Fig. 12. Larger-scale simulation results. The graph shows the sea floor, minimum, maximum and optimal depth, as well as the simulated waypoints and AUV depth.

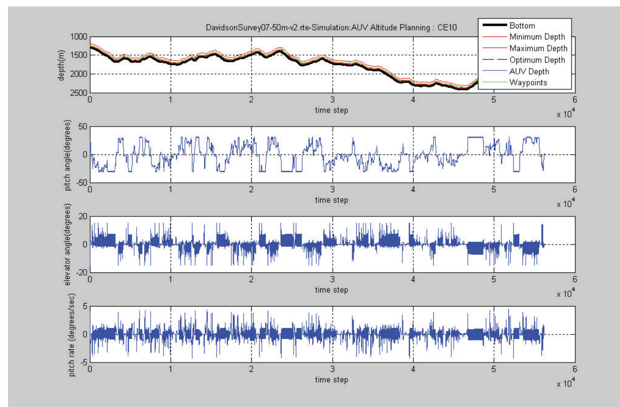


Fig. 13. The graph zooms in on a small portion of the simulation results in order to show algorithm performance along a difficult bathymetric inflection. The red lines show the minimum and maxium depths. The dotted line shows the target depth. The blue line shows the simulated AUV depth. At the difficult inflection point, our algorithm induces the AUV to change altitude in anticipation of the bathymetry, so that it does not grossly overshoot the target depth while still observing the mini-mum depth constraint.

at a steep bathymetric inflection. At this feature, our algorithm guides the AUV to change its depth in anticipation of the bathymetry, so as to avoid grossly overshooting the target depth while maintaining its minimum depth constraint.

## VIII. Field Test Results

After performing the simulation experiments, we then conducted a field test using our algorithm to plan the waypoint based control inputs for the AUV mission and comparing it to results from the cruder algorithm for the same area. The test was conducted at a bathymetrically difficult region three miles off shore in Monterey Bay (see Figures 14 and 15). The blue region indicates deeper conditions whereas orange indicates shallower conditions. The UAV made three forays into deep water along a relatively challenging path for the algorithm and AUV to navigate.

Figure 16 shows the results of our algorithm for the AUV mission. The black line shows the ocean floor. The red line

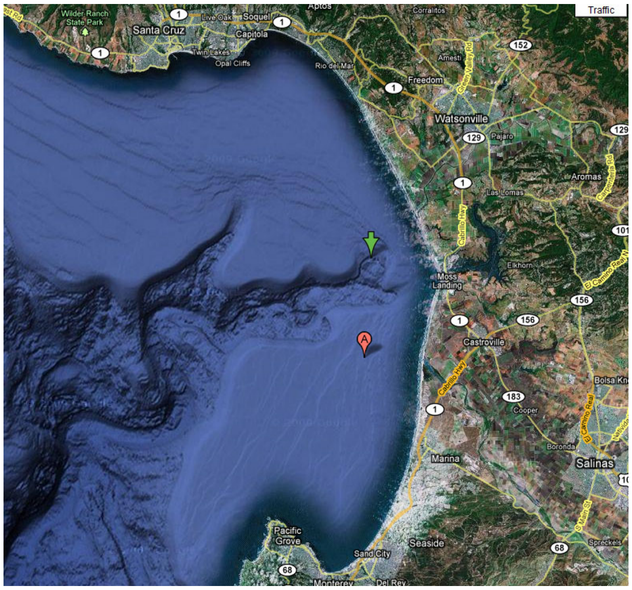


Fig. 14. We conducted a field test using our algorithm. Our spliced linear programming depth control algorithm has been successfully deployed at a bathymetrically difficult region three miles off shore in Monterey Bay. This figure shows the geographic location of our field test.

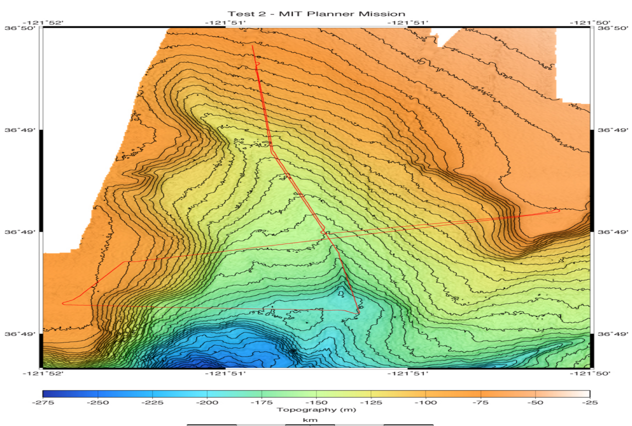


Fig. 15. The path of the mission is shown. The transect was designed to be challenging for the algorithm and AUV to navigate. The blue region indicates deep water and the orange region indicates shallow water. The mission starts in shallow water, at the top of the figure, then goes down, left, right, left and back up. The result is three forays into deep water.

shows the AUV depth under the control of the former depth planning algorithm. The green line shows the actual AUV depth while being controlled by our sliced linear programming algorithm. During steep inclines, the old algorithm anticipated the depth change and turned up too early. Our algorithm induces the AUV truncate these responses allowing the AUV to track the target depth more closely (see Fig. B-10).

## IX. Contribution

Our spliced linear programming depth control algorithm has been successfully deployed at a bathymetrically difficult region three miles off shore in Monterey Bay (see Figures 14 and 15). The algorithm improves the quality of scientific

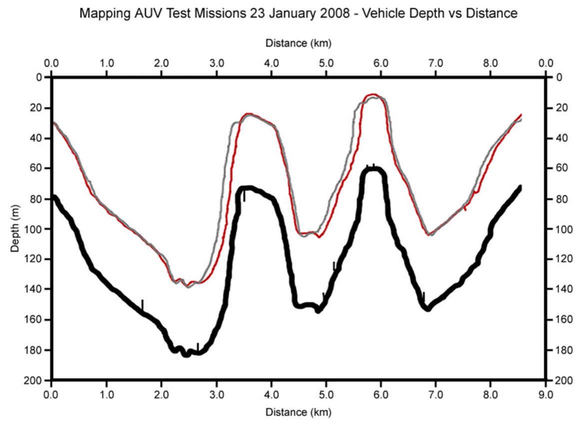


Fig. 16. Our algorithm improves the quality of scientific measurements by allowing the underwater vehicle to approach closer to the sea floor. The graph shows the field test results comparing our algorithm to the previously used algorithm. The black line shows the ocean floor. The red line shows the actual path followed when controlled by our new algorithm. The green line shows the actual altitude under the old algorithm. The results show that the AUV tracked the target depth better when controlled by our new algorithm.

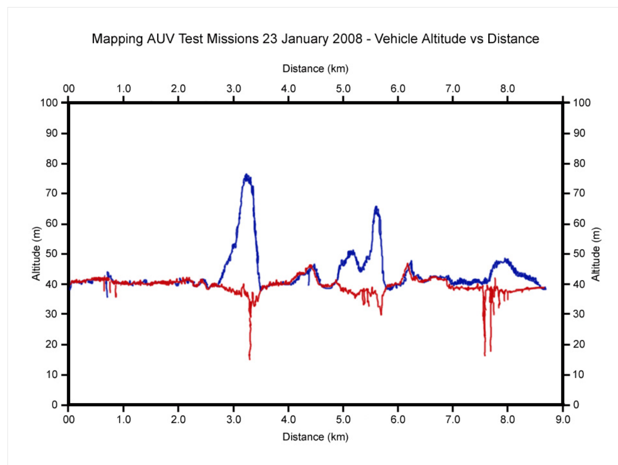


Fig. 17. Our algorithm was more successful than the previously existing algorithm at maintaining the proper altitude above the sea floor. This graph shows the deviation from the target altitude for our new algorithm (in red) and for the old algorithm (in green). This graph makes it easy to see the deviation from the target altitude. The results show that the AUV tracked the target depth better when controlled by our new algorithm.

measurements by allowing the underwater vehicle to approach closer to the sea floor as shown in Figure 16. Figure 17 shows that our algorithm is more successful than the previously existing algorithm at maintaining the proper altitude above the sea floor.

## Acknowledgment

The authors would like to thank Kanna Rajan for inviting us to the MBARI research facility, David Caress for allowing us to work on his AUV platform and for conducting the field test, and Rob McEwen for helping us with the vehicle dynamics.

# REFERENCES


[1] Beckman, R., Martinez, A., & Bourgeois, B. (2000). AUV positioning using bathymetry matching. In OCEANS 2000 MTS/IEEE Conference and Exhibition (Vol. 3, pp. 2123-2127). IEEE.

[2] Bellingham, J.G., K. Streitlien, J. Overland, S. Rajan, P. Stein, J. Stannard, W. Kirkwood, and D. Yoerger. 2000. An arctic basin observational capability using AUVs. Oceanography 13:4-70.

[3] Bellingham, J. G., & Willcox, J. S. (1996, June). Optimizing AUV oceanographic surveys. In Autonomous Underwater Vehicle Technology, 1996. AUV'96., Proceedings of the 1996 Symposium on (pp. 391-398). IEEE.

[4] L. Blackmore, L. Bush, *MBARI Dorado AUV Depth Planner Code Distribution.* http://people.csail.mit.edu/bushl2/transfer/to_dave_caress/.

[5] L. Blackmore, L. Bush, *Depth Planner User's Guide.* http://people.csail.mit.edu/bushl2/transfer/to_dave_caress/Depth Planner User's Guide c.pdf.

[6] Caress, D. W., & Chayes, D. N. (2008). MB-System: Open source software for the processing and display of swath mapping sonar data. http://www.mbari.org/data/mbsystem.

[7] Caress, D. W., Thomas, H., Kirkwood, W. J., McEwen, R., Henthorn, R., Clague, D. A., Charles K. Paull, Jenny Paduan, and Maier, K. L. (2008). High-resolution multibeam, sidescan, and subbottom surveys using the MBARI AUV D. Allan B. Marine habitat mapping technology for Alaska, 47-69.

[8] Curtin, Thomas B., James G. Bellingham, Josko Catipovic, and Doug Webb. "Autonomous oceanographic sampling networks." Oceanography 6, no. 3 (1993): 86-94.

[9] Henthorn, R., Caress, D. W., Thomas, H., McEwen, R., Kirkwood, W. J., Paull, C. K., & Keaten, R. (2006, September). High-resolution multibeam and subbottom surveys of submarine canyons, deep-sea fan channels, and gas seeps using the MBARI mapping AUV. In OCEANS 2006 (pp. 1-6). IEEE.

[10] Gertler, M. (1950). Resistance experiments on a systematic series of streamlined bodies of revolution-for application to the design of high-speed submarines. DAVID TAYLOR MODEL BASIN WASHINGTON DC.

[11] Haley, P. J., Lermusiaux, P. F. J., Robinson, A. R., Leslie, W. G., Logoutov, O., Cossarini, G., ... & Bellingham, J. (2009). Forecasting and reanalysis in the Monterey Bay/California current region for the autonomous ocean sampling network-II experiment. Deep Sea Research Part II: Topical Studies in Oceanography, 56(3), 127-148.

[12] Kirkwood, W. J., Caress, D. W., Thomas, H., Sibenac, M., McEwen, R., Shane, F., ... & McGill, P. (2004, November). Mapping payload development for MBARI's Dorado-class AUVs. In OCEANS'04. MTTS/IEEE TECHNO-OCEAN'04 (Vol. 3, pp. 1580-1585). IEEE.

[13] Kirkwood, W., Caress, D. W., Thomas, H., McEwen, R., Shane, F., Henthorn, R., & McGill, P. (2005). Results from MBARI's integrated mapping system. In Proceedings of OCEANS 2005 MTS/IEEE (pp. 563-570). IEEE.

[14] Kirkwood, W. J. 2007. Development of the DORADO mapping vehicle for multibeam, subbottom, and sidescan science missions. J. Field Robot. 24:487-495. http://dx.doi.org/10.1002/rob. v24:6. (Accessed April 2008.)

[15] Kirkwood, W.J., D.W. Caress, H. Thomas, M. Sibenac, R. McEwen, F. Shane, R. Henthorn, and P. McGill. 2004. Mapping payload development for MBARIs Dorado-class AUVs. Marine Technology Society/IEEE Oceans 2004, Conference Proceedings.

[16] Kirkwood, W.J, D.W. Caress, H. Thomas, M. Sibenac, R. McEwen, F. Shane, R. Henthorn, and P. McGill. 2005. Results from MBARIs Integrated Mapping System. Marine Technology Society/IEEE Oceans 2005, Conference Proceedings.

[17] Kirkwood, W.J., D. Gashler, H. Thomas, T.C. OReilly, R. McEwen, N. Tervalon, F. Shane, D. Au, M. Sigenac, and J.G. Bellingham. 2001. Development of a long endurance autonomous underwater vehicle for ocean science exploration. Marine Technology Society/IEEE Oceans 2001, Conference Proceedings.

[18] McEwen, R., & Streitlien, K. (2001). Modeling and control of a variable-length AUV. Proceedings of the 12th UUST.

[19] McGann, C., Py, F., Rajan, K., Thomas, H., Henthorn, R., & McEwen, R. (2008, May). A deliberative architecture for AUV control. In Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on (pp. 1049-1054). IEEE.

[20] Whitcomb, L., Yoerger, D., & Singh, H. (1999). Advances in Doppler-based navigation of underwater robotic vehicles. In Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on (Vol. 1, pp. 399-406). IEEE.

[21] Yoerger, D. R., Bradley, A. M., Walden, B. B., Cormier, M. H., & Ryan, W. B. (2000). Fine-scale seafloor survey in rugged deep-ocean terrain with an autonomous robot. In Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on (Vol. 2, pp. 1787-1792). IEEE.

[22] J. Yuh, *Design and control of autonomous underwater robots: A survey.* Autonomous Robots, 2000.

[23] Zhang, Y., Bellingham, J. G., Davis, R. E., & Chao, Y. (2005, December). Optimizing Autonomous Underwater Vehicles' Survey for Reconstruction of an Ocean Field that Varies in Space and Time. In AGU Fall Meeting Abstracts (Vol. 1, p. 1178).