Proceedings of the 15th Mediterranean Conference on
Control & Automation, July 27 - 29, 2007, Athens - Greece

T34-004

Evolutionary Path Planning and Navigation of Autonomous Underwater Vehicles

V. Kanakakis and N. Tsourveloudis
Technical University of Crete
Department of Production Engineering and Management
Intelligent Systems and Robotics Laboratory, 73100 Chania, Crete, Greece.
{vkanakakis, nikost}@dpem.tuc.gr

***Abstract*— This paper presents a complete methodology for mission planning and navigation of Autonomous Underwater Vehicles (AUVs) in ocean environment. Path planning near the ocean floor is accomplished via genetic algorithms and B-Splines based on known data of the ocean floor. In addition, collision free navigation is achieved in unknown environments. Prior to vehicle's launch, a genetic algorithm based on ocean floor data and on mission restrictions calculates the optimal path. Once the path is calculated, the vehicle is navigated through the predefined path by a set of fuzzy controllers. A second evolutionary algorithm optimizes the membership functions of these controllers so as the vehicle has the minimum error through its course. Extensive simulations were performed in order to evaluate the methodology and the derived optimized controller.**

I. Introduction

As the technology of Autonomous Underwater Vehicles (AUVs) [1] grows and their applications and relative tasks are more demanding, new control architectures and methodologies are needed to expand their potentials [3]. Referring to AUV navigation one can distinguish three different problems [11], [12]: 1. *Close-to-surface* navigation, where GPS sensors can provide accurate position data. In addition to GPS, a various set of sensors can be used to estimate the vehicle's position and state such as Inertia Measurement Units (IMUs), Doppler sonars, altimeters, inclinometers and magnetic compasses. 2. Navigation in the *mid-depth* zone, where the AUV operates far from the sea surface and far from the sea floor. In that case acoustic sonars and IMUs are the only effective sensors. 3. *Close-to-bottom* navigation, where the needs for precise sensing and autonomous operation are increased. Thus close-to-bottom path design and following as well as precise navigation and effective collision avoidance are today's challenges and objectives of the current research.

The paper describes a complete methodology applicable to close-to-bottom navigation of AUVs, which generalizes techniques recently developed and published in the area of AUV navigation and path planning [5], [6], [9]. Similar, fuzzy modular controllers have been designed and tested with promising results [2], [4], [8], [10], while the current work expands the benefits of this technology.

When the mission of an AUV enables close-to-bottom navigation, then the vehicle should be fitted with a path planner which is able to compute off-line a suitable path given the ocean floor features at the area of operation and restrictions to be applied during the mission. Accordingly the vehicle should be able to follow the predefined path with minimum error based on its sensors. During path following (on-line) the vehicle should be able to avoid obstacles that had not been taken previously into consideration. A variety of sensors can be utilized for vehicle state estimation and obstacle avoidance. Such sensors are altimeters (z-altitude), pressure sensors (z-depth), inclinometers (roll, pitch), magnetic or gyro compass (heading), IMUs (Euler angles, accelerations, yaw rate), sonars (obstacle detection), and various cameras.

The derived optimum path should be in accordance to restrictions that may vary in each case. Typical restrictions are: 1. maximum depth of path, which is inherited by the capabilities of each vehicle. 2. Minimum depth of path which is set by operational requirements, 3. Minimum distance between path points and ocean floor, set by controllability of the vehicle and by safety requirements, and 4. Minimum curvature on each point of the path set by vehicle controllability.

Here we suggest a path planning and navigation methodology which may be applicable to a wide variety of AUVs. The architecture of the proposed method is in Fig. 1. It enables three separate levels of mission planning, optimization and motion control. The first two layers are accomplished off-line prior to the mission, while the third level is responsible for the on-line navigation with collision avoidance and low level motion control of the vehicle.

In the first layer of Fig. 1 a Genetic Algorithm (GA), similar to the one presented in [7], is used to compute the desired path. The GA's chromosomes are the coordinates of the control points of a B-Spline curve [7]. The ocean floor data as well as the mission restrictions form the fitness function of the GA.

In the second layer another GA is used to optimize the membership functions of the navigation fuzzy controller. The objective of this controller is to follow the path derived from the first layer. The third layer is used to guide the vehicle through the derived optimum path with the derived genetically optimized controller while it enables collision avoidance in case of unexpected obstacles. In such case the goal-based navigation is altered to a reaction-based one.

The rest of the paper is organized as follows. In section 2 the genetic path planner is described and analyzed. In section 3 the optimization of the motion control module as well as the simplified low level control module are presented. The results shown in section 4 are compared to previously used approaches. Finally, section 5 concludes the paper with remarks and suggestions for future work.

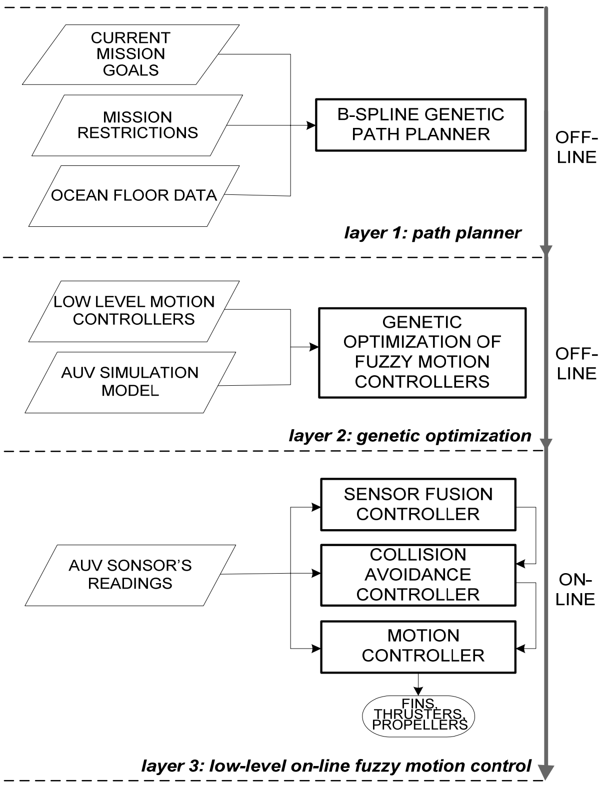

Figure 1. The proposed path planning and navigation architecture.

II. GENETIC PATH PLANNING

The objective of path planning is to define the best trajectory that the AUV should follow in order to accomplish its mission.

In previous path planning approaches [13], [14], [15], the path was commonly represented by way points as a set of line segments. The generated error and the possible instability that may occur because of sudden and consequential direction altering may be one disadvantage of such methods.

In the current work the use of B-Splines for the representation of the AUV's path, leads to smooth curves with minimum curvature that the vehicle can follow during its mission. A B-Spline is a parametric curve based on blending functions defined by its control points [16]. If the number of control points is (n+1) with coordinates (x_o, y_o, z_o),…,(x_n, y_n, z_n) the coordinates of the B-Spline may be written as:

$$x(t) = \sum_{i=0}^{n} x_i \cdot B_{i,K(t)},$$
$$y(t) = \sum_{i=0}^{n} y_i \cdot B_{i,K(t)}, \quad (1)$$
$$z(t) = \sum_{i=0}^{n} z_i \cdot B_{i,K(t)},$$

where $B_{i,\, K(t)}$ represent the blending functions and K the order of the curve witch is associated with the curve's smoothness, with higher values of K corresponding to smoother curves. The sum of the values of the blending functions for any value of the parameter t is always 1. The blending functions are defined recursively in terms of a set of *Knot* values, as explained in [7].

Its start and endpoints, the second point and six free-to-move points define the shape of a B-Spline path. The Cartesian coordinates of these free-to-move points are the chromosomes of the individuals to be optimized by the genetic procedure. The quality of the solution is evaluated by the fitness function, which is calculated for each member of the population. By this way, all members are scaled accordingly to their scores so that members with greater scores have greater possibility to pass their genes to the next generation and to give offsprings through the mutation and crossover operators.

A. The Fitness function

The fitness function is a weighted sum of six terms, each one connected with path constraints. These constraints describe the path feasibility, that is, whether can be followed by an AUV. The fitness function is defined as follows:

$$f = \sum_{i=1}^{6} a_i \cdot f_i, \quad (2)$$

where a_i is an importance index of f_is. The term f_1 penalizes the existence of a non-feasible part of the path. It represents the percentage of the discretized B-Spline curve points (not the control points) that have z-heights greater of those of the bottom and are located under the solid sea floor boundary. By this way curves with smaller segments under the sea floor may give less values of the fitness function and are more probable to pass their genes to the next generations.

Term f_2 is the ratio between the actual length of the B-Spline and the straight distance between the start and the end point, which is the minimum possible. It penalizes the overall length of the path, takes the values of one or greater and is weighted into the sum of the fitness function so that the shortest curves have greater possibilities to give offsprings.

Term f_3 is the percentage of the B-Spline points that are outside the bounding area of mission in the x-y plane. The size of the bounding area is user defined and may represent a mission restriction.

Term f_4 is the percentage of the B-Spline points that are outside the bounding minimum or maximum acceptable depth in the z plane. The depth limitations may as well are path restrictions and are useful when the AUV needs to navigate in specific depth regions or near the ocean floor.

Term f_5 is the percentage of the B-Spline points that are located within a minimum safety distance from the ocean floor, i.e. that have distance from the floor less than a minimum one. The minimum safety distance is user defined and is useful to keep the vehicle away from the dangerous area of sea floor and its features.

Term f_6 is the percentage of the B-Spline points that have a local curvature radius of less than a minimum user defined value. The radius is calculated for all the separate points and their adjacent ones and if it is less than the minimum then the curve is penalized. This term penalizes the curves that have good scores but form paths that the AUV will have difficulty to follow and thus big overshoot errors will occur during the execution. The minimum path radius that the AUV can follow is user defined accordingly to the AUV's specifications for maneuverability such us minimum turn diameter.

B. The Genetic Algorithm Procedure

The above described fitness function is calculated for each one of the individuals and each individual's performance is evaluated by it's score. Accordingly the satisfaction of the best fitness-stopping criterion is checked, i.e. the algorithm is stopped when one individual's performance is better than the best expected in order to save computational time. If this is not the case, the fitness function is evaluated for all the individuals of the population.

If a generation has reached a predefined value the procedure is stopped and the solution is given by the next individual. If not, crossover and mutation operators are applied to the individuals who are combined to give off springs that will form the next generation. With the crossover operator two randomly selected individuals are divided in a random gene and the first part of the first individual is connected to the second part of the second individual and vise-versa. By this way information is exchanged between two possible solutions of the problem. The mutation operator alters a randomly selected gene of an individual with a randomly selected value from the constrained space of genes as defined in the beginning of the process. The mutation operator introduces variability to the possible solutions, explores the possible solution space, and refines the solutions. The individuals with better scores (smaller fitness values) are more possible to participate to the reproduction and to pass their genes to the new generation. This procedure imitates the natural procedure of biological reproduction. After the formation of the new population a new circle starts.

It should be noted that since ocean environment is in a great extent unpredictable, the use of path planning alone is insufficient to ensure collision free navigation and collision avoidance methods are need to be incorporated during the execution of the mission [12].

III. Optimization of Low-Level Motion Control

After the calculation of the optimum path, the next step at the layered control architecture (Fig. 1) is the optimization of the motion control module. The procedure is designed to ensure that the AUV will have the minimum position error. The genetic optimization of the motion control module depends on reliable AUV modeling, and exact motion parameters identification. When position error increases or when an obstacle appears (moving obstacle or in areas with bottom data of poor accuracy) the collision avoidance module will navigate the AUV with reaction based navigation in collision free areas.

The control approach we use here is described in [8]. It provides collision avoidance by a sensor fusion module where the information provided by the vehicle's sensors is evaluated and collision possibilities at the four cardinal directions are computed. The low level motion control is provided by three fuzzy controllers i.e. the *pitch*, *depth* and *yaw* controllers. The pitch and depth controller are controlling the fins and thrusters for pitch angle adjustment and vertical positioning of the AUV in the vertical x-z plane. The yaw controller is controlling the fins and thrusters for adjustment of the orientation of the vehicle in the horizontal x-y plane. The rulebase of these controllers are of the Mamdani type and they all have similar structure [8]. The membership functions of the pitch, depth and yaw controllers have been redesigned and the membership functions are modified to be z-shaped, triangular and s-shaped.

The individuals that are optimized through the genetic process consist of genes that correspond to the left, center and right values of the membership functions. For a total of 3+3+2=8 inputs and 3 outputs for the pitch, depth and yaw controllers, the total number of membership functions required is 21 (5 z-mfs, 5 s-mfs, and 11 trimfs) thus all individuals should consist of 53 genes. This number leads to great populations and increased computational time. In order to minimize both individual size and computational time we assume symmetric membership functions for all fuzzy sets involved. By using symmetric membership the described gene assignment the individuals to be evolved consist now of 33 chromosomes. The assumption of symmetric membership functions is totally realistic as to navigate an AUV in a uniformly random underwater environment, symmetric membership functions are required so that left-right motion in the horizontal plane and up-down motion in the vertical plane are treated in the same way by the controllers. Furthermore, all required motions and turns will have the same contribution to the evolution of the fittest membership functions, regardless of vehicle's direction.

A. The Fitness function

The fitness function is used to score the quality of the controllers described by each individual's genes through the genetic process. The score is to be minimized through the process as the population evolves towards better generations. In order to optimize the 33 membership functions of the three motion controllers a fitness function has been written in MATLAB®. The fitness function has inputs the individuals and the values of their chromosomes. At the first step the chromosomes are assigned to left, center and right arrays with size 33x1 in order to construct 33 z-shaped, triangular and s-shaped membership functions. At the next step various checks and corrections are performed in order to ensure that the values of left, center, right arrays represent proper and feasible membership functions. This is necessary as the genes of the individuals are created through random processes, such as, crossover and mutation and thus can inherit inconsistencies that have no meaning to membership functions.

After all possible corrections and penalty setting, the chromosomes represent feasible membership functions and the movement of the AUV through the path is simulated. A simulated path is generated and during the AUV's mission the average position error is calculated, as described in the next paragraphs.

The optimum B-Spline path, which was created during the path planning layer, is now discretized in *n* points. The Cartesian coordinates of each of these points is linked to the AUV motion model and the simulation starts with the AUV trying to reach the target points in sequence. Each target point is considered to have been reached when the

AUV reaches it in a predefined distance. This predefined radius defines a circle noted as circle of acceptance. When the AUV has reached this circle the target point shifts to the next target point. At each simulation step the perpendicular distance of the position of the AUV to the line connecting the transition and target points is calculated. This distance is summed to the total error which is then divided by the number of simulation steps to give the average position error of the AUV. The above described procedure is shown in Fig. 2.

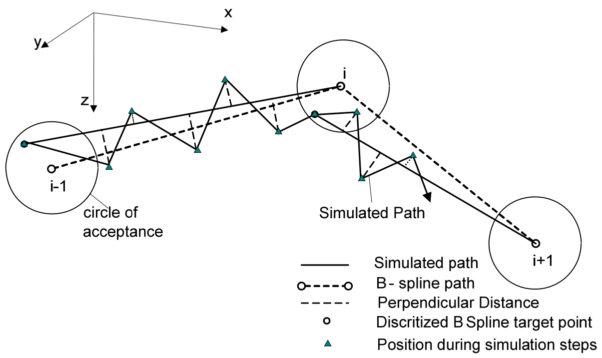

Figure 2. Path following and position error count during simulation.

When the perpendicular distance in each step exceeds a predefined threshold, it is assumed that the controllers cannot control the AUV sufficiently and the simulation stops after the individual is penalized.

IV. EXPERIMENTAL RESULTS

The proposed three-layered control architecture has been tested using a simulation environment. Genetic procedure calculations were programmed using the MATLAB® Genetic Algorithm Tool. A previously developed Phoenix AUV [18] motion model is used in the simulations for the optimization of the pitch, depth and yaw controllers.

A. Path Planning

As previously mentioned, the path-planning layer calculates the optimal path in B-Spline form. The first, the second and the end points of the B-Spline path are fixed while the rest of the path is defined by six free-to-move points. The maximum and minimum depths are defined as: max(z) = 100 ft, min(z) = 85 ft. Each individual has 18 genes, and the population size was set to 30. The crossover possibility was set to 0.9, the elite count was set to 1 and the maximum number of generations is 100.

Fig. 3 is a typical example of the derived paths. It shows the produced B-Spline path through the ocean floor. By observing Fig. 3 one could conclude that the path generator produces smooth and feasible B-Spline paths. This path is used to train the membership functions of the low level motion control module as previously described.

B. Optimization of Navigation Controllers

The optimization of the membership functions of the low-level motion controller is performed with MATLAB's Genetic Algorithm toolbox. Population size is set to be 10. The relatively small (with respect to the individual size) population size is chosen in order to reduce computational time. The dimension space of possible solutions is scattered through the mutation operator which has a possibility of 0.2 and the possibility of crossover is 0.6.

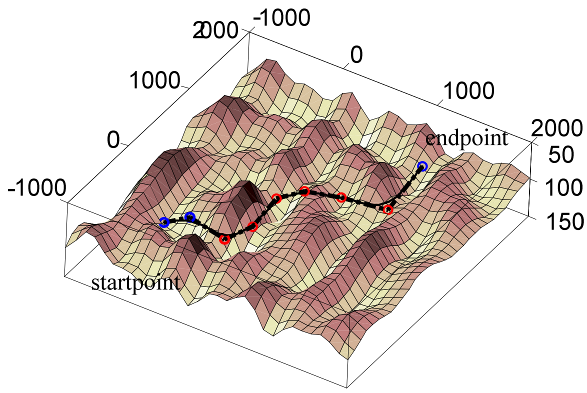

Figure 3. Optimized B-Spline path

The possibility of a chromosome to be mutated within an individual is 0.3. The best individual of each generation survives to the following generation. The scaling of the individuals and respectively their expectation to pass their genes to the new generations is set to be proportional to their scores. The maximum number of generation is set to be 80.

C. Performance evaluation - Comparisons

In order to evaluate the performance of the genetically evolved fuzzy controllers, we performed various comparisons between them and the manually tuned fuzzy controllers presented in [8]. Also the evolved controllers were compared to a set of simplified fuzzy controllers with z-mf, trimf and s-mf membership functions not manually tuned or evolved. All simulation runs are carried out with MATLAB® and SIMULINK®. During each simulation run, the actual position and the position error of the AUV are stored. After the simulation is terminated (when the AUV reaches it's target), the maximum error, the mean error, the time needed and the overall length of mission are calculated.

Three test cases are presented. In the first test case the AUV is moving along the predefined B-Spline path using three different sets of fuzzy controllers; the manually tuned, the simplified (without tuning) and the genetically evolved one. In the second test case similar comparisons are made under the presence of lateral ocean current of various velocities. In the third test case the AUV is simulated to move along a different B-Spline path than the one used to train the genetically evolved controller. The three test cases are presented in the following paragraphs.

Test case 1: In this test case the AUV has to follow the path presented in Fig. 3. Ocean currents are not present. Results are shown in Table I, whereas the actual paths are shown in Fig. 4. From Table I it can be noticed that the not tuned (w/o tuning) fuzzy controllers have the worse performance among the three tested; the manually tuned controller and the genetically evolved one have almost the same average and maximum position errors (with the genetic tuned been slightly worse on the average error). The advantage of the genetically evolved controller is that with this setting the AUV needs much less time to complete the mission.

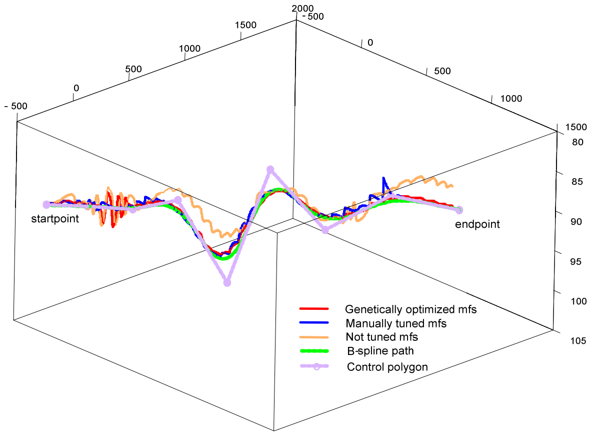

Figure 4. Test case1: Path lines achieved from various fuzzy controllers – No ocean current present - Distances in ft.

TABLE I.
PERFORMANCE EVALUATION OF CONTROLLERS FOR THE TEST CASE 1.

	w/o tuning	Genetically tuned	Manually tuned
Max error (ft)	11.2498	8.1033	8.1408
Average error	2.4335	1.3298	1.2812
Time (sec)	1768.4	1710.2	1815.4
Path Length (ft)	3083.566	3078.922	3085.370

Test case 2: In this case a lateral ocean current is taken into consideration. The AUV is moving along the predefined B-Spline path using the three different fuzzy controllers like in the test case 1. Certain performance measures are shown in Table II for ocean current speeds of 0.1, 0.2 and 0.3 ft/sec. The ocean current is assumed to have time-steady velocity in the x-y plane. The direction of its velocity is chosen to be vertical to the line connecting the start and end points of the path. The AUV motion lines for all controllers under the presence of 0.3 ft/sec ocean current are shown in Fig. 5.

TABLE II.
PERFORMANCE EVALUATION OF CONTROLLERS WITH LATERAL OCEAN CURRENTS

			w/o tuning	Genetically tuned	Manually tuned
Max error (ft)	Current velocity (ft/sec)	0.1	9.6775	8.4075	8.3006
		0.2	8.9036	9.5211	8.7098
		0.3	8.3877	11.3468	10.1037
Mean error (ft)	Current velocity (ft/sec)	0.1	2.3000	1.4645	1.3184
		0.2	2.2722	1.6798	1.5496
		0.3	2.4068	1.9257	1.8010
Time (sec)	Current velocity (ft/sec)	0.1	1760.6	1769.6	1794
		0.2	1766.4	1783.6	1795
		0.3	1782.4	1812.6	1818
Path Length (ft)	Current velocity (ft/sec)	0.1	3083.431	3081.576	3083.365
		0.2	3084.576	3085.644	3085.706
		0.3	3086.571	3091.678	3090.963

From Table II it may be noticed that the genetically evolved controller has slightly worse performance, in terms of mean error, than the manually tuned one in almost all cases with lateral ocean current.

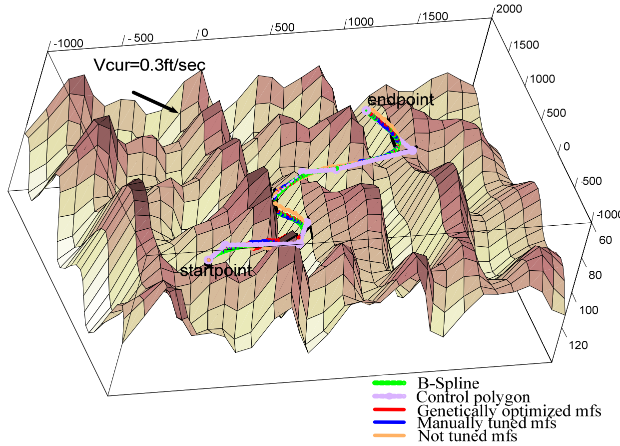

Figure 5. Test case 2: Path lines achieved from various fuzzy controllers – Lateral ocean current 0.3 ft/sec - Distances in ft.

Test case 3: The AUV follows a path line other than the one used in the genetic process of evolving membership functions for the fuzzy motion controller of the motion control module.

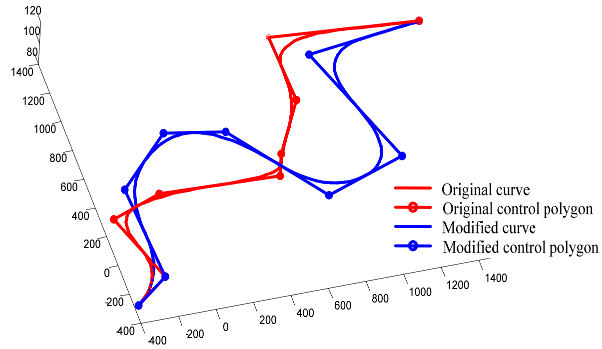

Figure 6. Original and modified B-Spline paths.

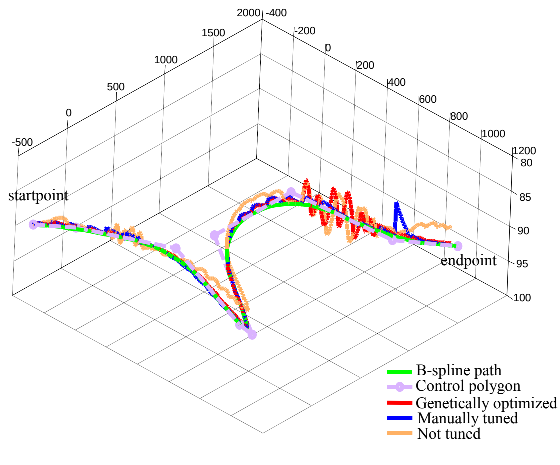

Figure 7. Test case 3: Performance evaluation of controllers – Path following along a modified B-Spline

As previously explained, the B-Spline path is determined via three fixed points, i.e. the start point, the second point and the end point, and six free-to-move control points which are genetically optimized so as to produce a collision free B-Spline path. An altered B-Spline is created by manually changing the six control points and holding the first, second and end points the same. As a result the new B-Spline is not related to the ocean floor and is used only to evaluate the robustness of the fuzzy controllers. The genetically determined and the altered B-Splines with their control polygons are shown in Fig. 6. The maximum and the average error as well as the path length and the time needed to complete the path are presented in Table III, while the actual paths of each controller are presented in Fig. 7.

TABLE III.
PERFORMANCE EVALUATION OF CONTROLLERS FOR THE TEST CASE 3.

	w/o tuning	Genetically tuned	Manually tuned
Max error (ft)	7.9124	8.3458	10.2202
Average error	2.7308	1.7287	1.60
Time (sec)	2035.4	2044.8	2075
Path Length (ft)	3636.602	3634.678	3641.308

From Table III it can be noticed that the genetically evolved fuzzy controller has better performance than the manually tuned one in all aspects except on the average error. This means that it has greater average overshoot error though it is able to maintain vehicle's stability.

V. CONCLUSION

We have presented a layered control architecture applicable to AUVs and especially useful for near to ocean floor missions. Accurate navigation is achieved through optimum path planning utilizing B-Splines and genetic algorithms, and evolution of the membership functions of the motion fuzzy control module to fit the defined optimum path. In order to apply the suggested methodology accurate data for vehicle's characteristics and hydrodynamic coefficients are needed.

With the proposed control architecture an AUV will be able to navigate closer to obstacles not in segmented line paths but in paths with smooth form. The genetic algorithms have proven to be a robust methodology in finding solutions in path planning and membership function tuning. The path planner takes into consideration the AUV's maneuvering characteristics to design paths that the AUV will be able to follow. Extensive testing of the derived controllers has been presented, also under the presence of lateral ocean current and under modified paths to simulate the real case of AUVs operation in ocean environments. The methodology has proven to give controllers that navigate the vehicle with greater accuracy for a given path, compared to controllers that are manually/empirically tuned. The benefits of the derived methodology beside the derived accuracy, is the uniform design of the control module for all AUV types and the smaller overall time needed for this design. Another advantage is that the genetically evolved fuzzy motion controllers navigate the AUV in paths with smaller length and in less time, which leads to greater battery efficiency and operation autonomy. In the future, this will be further investigated in realistic conditions.

REFERENCES

[1] J. Yuh, (ed.), *Underwater Robotic Vehicles: Design and Control*, TSI Press, 1995.

[2] N. Kato, "Applications of fuzzy algorithm to guidance and control of underwater vehicles," in *Underwater Robotic Vehicles*: *Design and Control*, J. Yuh (ed.), TSI Press, 1995.

[3] K. P. Valavanis, D. Gracanin, M. Matijasevic, R. Kolluru, and G. A. Demetriou, "Control architectures for autonomous Underwater vehicles," *IEEE Contr. Syst. Mag.*, vol. 17, pp. 48-64, 1997.

[4] N. Tsourveloudis, D. Gracanin, and K. P. Valavanis, "Design and testing of navigation algorithm for shallow water autonomous underwater vehicle," in *Proc. of IEEE OCEANS '98*, Nice, France, 1998.

[5] A. J. Healey and D. Lienard, "Multivariable sliding mode control for autonomous diving and steering of unmanned underwater vehicles," *IEEE J. Ocean Eng.*, vol. 18, no. 3, pp. 327–339, 1993.

[6] A. J. Healey and D. B. Marco, "Slow speed flight control of autonomous underwater vehicles: Experimental results with the NPS AUV II," in *Proc. of 2nd Inter. Offshore and Polar Engineering Conf.*, San Francisco, CA, pp. 523–532, 1992.

[7] I. K. Nikolos, K. P. Valavanis, N. C. Tsourveloudis, and A. N. Kostaras, "Evolutionary Algorithm Based Offline/Online Path Planner for UAV Navigation," *IEEE Trans. Syst., Man and Cybernetics, part B*, vol. 33, no. 6, pp. 898-912, 2003.

[8] V. Kanakakis, K. P. Valavanis, and N. C. Tsourveloudis, "Fuzzy-Logic Based Navigation of Underwater Vehicles", *J. Intel. Robot. Syst.*, vol. 40, pp. 45–88, 2004.

[9] T. I. Fossen and S. I. Sagatun, "Adaptive control of nonlinear systems: A case study of underwater robotic systems," *J. Robot. Syst.*, vol. 8, pp. 393–412, 1991.

[10] P. DeBitetto, "Fuzzy logic for depth control of unmanned undersea vehicles," *IEEE J. Oceanic Eng*., vol. 20, no. 3, pp. 242–248, 1995.

[11] J. C. Kinsey, R. M. Eustice, and L. L. Whitcomb, "A Survey of Underwater Vehicle Navigation: Recent Advances and New Challenges", In *Proc. of IFAC Conf. of Manoeuvring and Control of Marine Craft*, Lisbon, Portugal, 2006.

[12] J. J. Leonard, A. A Bennett, C. M. Smith, H. J. S. Feder, "Autonomous Underwater Vehicle Navigation," MIT Marine Robotics Laboratory, Memorandum 98-1, Cambridge, MA, USA, 1998.

[13] R. Smierzchalski, "Evolutionary trajectory planning of ships in navigation traffic areas," *J. Marine Sci. Tech..*, vol. 4, pp. 1–6, 1999.

[14] R. Smierzchalski and Z. Michalewicz, "Modeling of ship trajectory in collision situations by an evolutionary algorithm," *IEEE Trans. Comput*., vol. 4, pp. 227–241, Sept. 2000.

[15] W. Naeem, R. Sutton, S. M. Ahmad and R. S. Burns, "A Review of Guidance Laws Applicable to Unmanned Underwater Vehicles," *J. of Navigation*., vol. 56, pp. 15–29, 2003.

[16] L. Piegl and W. Tiller, *The Nurbs Book*, 2nd ed: Springer, 1997.

[17] Fossen, T., *Guidance and Control of Ocean Vehicles*, Wiley, New York, 1994.

[18] Brutzman, D. P., "A virtual world for an autonomous underwater vehicle," *PhD thesis*, Naval Postgraduate School, Monterey, CA, USA, 1994.