Multivalued Versus Univalued Reactive Fuzzy Behavior Systems for Navigation Control of Autonomous Ground Vehicles

Majura F. Selekwa, Emmanuel G. Collins and Joseph Q. Combey Jr.
Department of Mechanical Engineering
Florida A& M University - Florida State University College of Engineering
Tallahasee, Florida, 32310
Email: majura@eng.fsu.edu; ecollins@eng.fsu.edu; combey@eng.fsu.edu

***Abstract*— Applications of autonomous ground vehicles (AGVs) in field operations have expanded from simple transportation tasks to complicated tasks such as military and rescue missions. The complexity in controlling these vehicles increases with the complexity of the tasks that the vehicles are intended for and the environment in which they are to operate. The behavior robotics approach has been adopted as a paradigm for controlling these systems. Due the uncertainty that surrounds the vehicle dynamics and their environments, fuzzy logic control approaches for navigation control have been developed, hence resulting in fuzzy behavior control systems. Two types of behavior structures have been proposed: the univalued and multivalued behaviors. This paper[1] presents a qualitative and quantitative comparison of the structure and performance of these behavior systems. The quantitative performance comparison is performed by using numerical simulation results for the motions of two identical AGVs each controlled by using one of the two types of fuzzy behavior navigation control systems; one vehicle uses a multivalued fuzzy behavior system, and the other uses a univalued fuzzy behavior system. The robots are made to navigate across a closed room with random obstacles.**

I. INTRODUCTION

Behavior based control systems break the control problem into simple sub-problems each of which is controlled by independent simple units called behaviors or reactive behaviors [1]. Reactive behavior systems rely entirely on sensory inputs for determining the control command, while the classic deliberative systems use stored information for modelling the environment and determining the appropriate control command.

The concept of behavior control was initially seen as a special form decentralized switching control in which each behavior is fully autonomous and when allowed it can control the robot on its own without regard to other behaviors. Under that paradigm, the behaviors are designed to be univalued, i.e., each behavior triggers a single control command that best meets the control responsibilities specific to that behavior. Over time, there has been a concerted efforts to make behaviors run cooperatively so that the overall robot reaction is generally an amalgamation of the commands from the individual behaviors through some form of command fusion [2], [3], [4], [5], [6]; the classic subsumption approach which picked only one command from the behavior that had highest priority [7] can also be viewed a special form of command fusion. Over time, this univalued structure has been found to have serious flaws [8]. In particular, by treating behaviours as fully autonomous, this structure tends to cause the robot to be indecisive when the behaviors have conflicting interests with nearly equal importance. This observation led to the introduction of multivalued behavior control systems [8], [9], [10], [11]. The first fuzzy implementation of multivalued fuzzy behavior systems was in [9] in which two fuzzy values were used: “allowed” and “disallowed”. A more complex multivalued fuzzy structure was implemented in [11].

The differences between univalued and multivalued fuzzy systems have never been documented. Furthermore, there are no documented results that compare the performance of univalued and multivalued fuzzy behavior systems. The purpose of this paper is twofold: to describe the qualitative differences between multivalued and univalued fuzzy behaviors systems, and to compare the performance of the two systems.

The paper is divided into five sections. Section II describes the structure of univalued and multivalued fuzzy behavior systems. It goes further to explain the qualitative differences between the two systems. Section III describes two navigation control algorithms: one that uses a univalued behavior system and one that uses a multivalued behavior system. Section IV presents simulation results that compare the two algorithms described in Section III. The paper presents concluding remarks in Section V

II. UNIVALUED AND MULTIVALUED REACTIVE FUZZY BEHAVIOR SYSTEMS

This section describes the general structures of univalued and multivalued fuzzy behavior systems. These structures can apply to a variety of control applications.

---

[1] Prepared through collaborative participation in the Robotics Consortium sponsored by the U. S. Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD 19-01-2-0012. The U. S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation thereon.

A. *Univalued reactive fuzzy behavior systems*

A univalued reactive fuzzy system consists of a finite set of distributed independent behaviors and a system of arbitration and command fusion. Each behavior responds to its stimuli by issuing a single command that is transmitted for command fusion. Figure 1 shows the basic structure of a univalued reactive behavior system, where each of the behaviors uses the environmental information to determine the control command that satisfies its particular objective, e.g., obstacle avoidance, path following, goal seeking, etc. The behaviors are said to be "univalued" because each behavior responds by triggering only one command signal. The command signal for one behavior does not take into consideration the command of another behavior. If the system has several behaviors with conflicting responses, these behaviors compete for the control of the robot, i.e., each behavior seeks to satisfy its own interests. The behavioral conflicts in these systems has been cited as one of the leading causes for robot navigation failures [8], [10], [11].

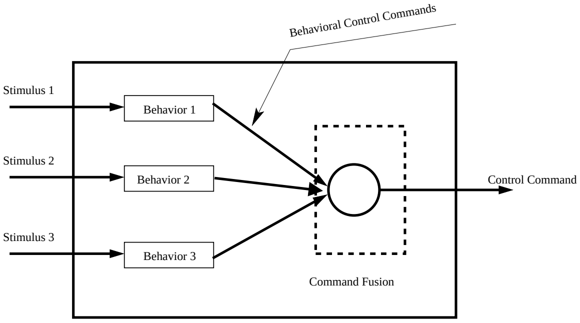

Fig. 1. The General Structure of the Univalued Fuzzy Behavioral Control System

B. *Multivalued reactive fuzzy behavior systems*

A multivalued behavior system, shown in Figure 2, also consists of a finite set of parallel running behaviors, and a centralized control command unit. A set of possible control commands is kept by this command unit and is also known by each behavior. The behaviors respond to their respective stimuli by expressing their preferences to the available set of command alternatives. The central control command unit augments the preferences that each command receives from the behaviors and selects the command that gets the highest score. The behaviors are treated as advisors to the central command unit; they together form an advisory block. Since each behavior has to express it relative preference to each of the available command alternatives it responds by firing multiple signals, one corresponding to each of the available command alternatives, i.e., the behavior is multivalued.

C. *Qualitative differences*

Although both the univalued and the multivalued fuzzy behavior systems are behavior based systems that use self contained behaviors, these systems have distinct differences. This section gives a brief description of the qualitative differences between these systems.

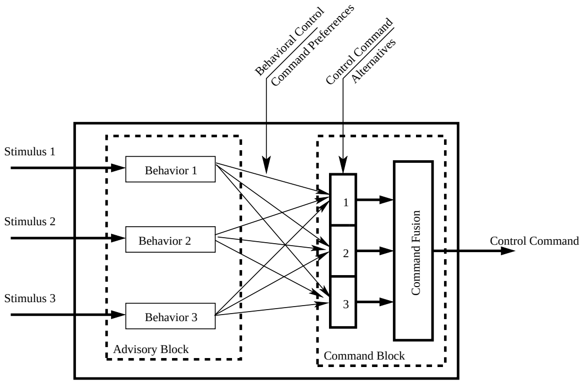

Fig. 2. The General Structure of a Multivalued Fuzzy Behavioral Control System.

The design of the behaviors in the univalued system allows them to directly influence the final control command, i.e., the response of the individual behavior is itself a control command that can drive the robot. On the other hand, the multivalued behaviors don't have that capability. The control commands are issued by the command block. Because of this structure, it is seen that the decision process passes through two steps a the univalued system and three steps for a multivalued system.

The behavior fusion process under a univalued system tends to favor the high priority behaviors such that the information communicated by the low priority behaviors tend to be ignored. Hence, there is an inherent loss of information under a univalued behavior system. On the other hand, since the fusion process for the multivalued system picks the command that best fits the interests of all behaviors, this system tends to preserve most of the information required for command generation.

Due to its structure, the univalued system is computationally simple, whereas the multivalued system is relatively complex requiring many more fuzzy rules in the command block. For example, if there are two inputs to each behavior that responds by expressing the relative suitability of each command using three fuzzy sets; then a multivalued fuzzy behavior system will have a total $3 \times 3^2 + 3^5 = 370$ rules, while the corresponding univalued behavior system would no more than $4 \times 3^2 = 36$ rules. The difference in the number of rules for the two systems is very significant.

III. TYPICAL NAVIGATION CONTROL APPROACHES

The robot navigation control problem consists of several objectives. For low speed motion a minimum of two objectives must be satisfied: (a) the ability to safely reach the goal, and (b) the ability to safely avoid hitting obstacles on the way. This section describes two algorithms for navigation control that are based on fuzzy behavior systems. One of the algorithms is based on a univalued fuzzy behavior system, and the other is based on a multivalued fuzzy behavior system.

A. *Univalued fuzzy behavior navigation control system*

Many types of navigation control algorithms have been proposed based on a univalued fuzzy behavior structure. A typical example is the *Adaptive Hierarchicy of Fuzzy Behaviors* [12], [4]. This system, which will be referred to by this paper as the *Univalued Algorithm*, breaks the control problem into a hierarchy of subproblems in which high level subproblems are dependent on low level subproblems. The individual subproblems in each level of the hierarchy are controlled by univalued behaviors. Behaviors at the bottom of the hierarchy, also known as primitive behaviors, are self-contained and designed to independently accomplish specific individual control objectives.

Figure 3 shows how this algorithm decomposes the Goal Directed Navigation control problem. Two subproblems are defined: the goal-seeking and the route-following. The behavior for the route following subproblem assumes that there is a predefined route that has to be followed. Since there was no predefined route for all experiments reported in this paper, this behavior was not included and the discussion that follows is based on the goal-seek behavior only. The goal-seek behavior is broken into the obstacle avoidance behavior and the go-to-xy behavior, which actually directs the robot to a particular point as directed by requirements of reaching the goal. The goal-seeking behavior receives the commands from the go-to-xy and avoid-collision behaviors and assigns some degree of applicability (DOA), which is a real number $\alpha \in [0, 1]$, to each of these commands. The degree of applicability expresses the importance of the commands for each behavior depending on the existing environment. After assigning the DOA, the commands are fused by using a fuzzy t-conorm operation [13] and the robot control commands are issued. The work in [12], [4] is based on a low level control of a differentially steered vehicle in which the control commands are the left and right wheel velocities (V_{wl} and V_{wr}). These wheel velocities determine both the heading direction (θ) and the speed (v) of the robot [14], [15]. It can be seen that the goal-seek behavior is a unit that intelligently fuses the commands from its primitive behaviors. Details of this algorithm can be found in [12], [4].

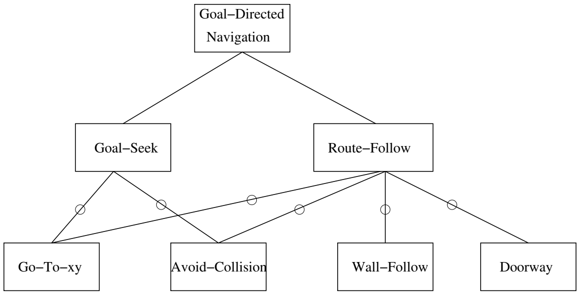

Fig. 3. Hierarchical Organization of the *Univalued Algorithm*.

The structure of the navigation control system that implements this algorithm is shown in Figure 4. It has two main parts, the primitive behavior block and the composite behavior block. The primitive behavior block consists of a set of independent, concurrently acting, primitive behaviors that respond to sensor data inputs. The composite behavior block is responsible for arbitration and fusion of the commands from the primitive behavior block. As can be seen, this structure is exactly that of the univalued fuzzy behavior system described in Section II.

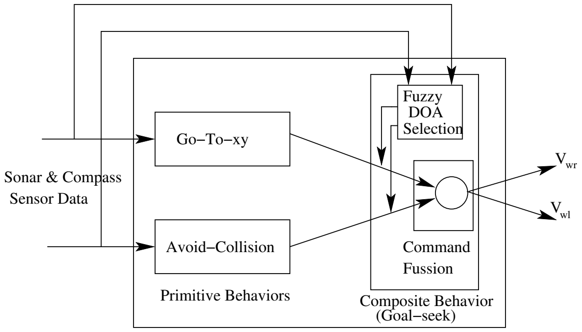

Fig. 4. Structure of the Navigation Control System That Implements the *Univalued Algorithm*.

B. Multivalued fuzzy behavior navigation control system

Navigation control algorithms that use multivalued fuzzy behaviors are described in [9] and [11]. The experiments reported here were performed using the system of [11]. This section gives a brief description of this algorithm, which will be referred to here as the *Multivalued Algorithm.*

The multivalued algorithm in [11] breaks the navigation control problem into two parallel control activities: the heading control and the speed control. The heading control activity controls the heading direction while the speed control activity controls the speed of the robot. Each control activity is controlled by one centralized multivalued fuzzy behavior whose structure was described in Section II. The control command for the heading control activity is $\Delta\theta$ where θ is the steering angle; and the control command for the speed control is Δv where v is the robot linear speed.

The overall system has five behaviors:(1) goal-seeking, (2) obstacle avoidance, (3) left edge tracking, (4) right edge tracking, and (5) overturning avoidance. The heading control uses four behaviors while the speed control uses two behaviors. The obstacle avoidance behavior runs in both the heading control activity and the speed control activity, and the overturning avoidance behavior runs on the speed control activity only; the remaining behaviors are for the heading control only. Each of these behaviors uses sensory information to determine its course of action. Structurally, the speed control system and the heading control system are very similar; the description that follows will be limited to the heading control system only with the understanding that it applies also to the speed control.

To describe how the system works, it is necessary to know the fuzzy sets that represent the possible command alternatives. For the heading control activity, the universe of discourse for the control command $\Delta\theta$ is fuzzified into five symmetric fuzzy sets known here as the output fuzzy sets. Two of these output fuzzy sets turn the robot to the right and the other two sets turn the robot to the left; the middle set maintains the course of the vehicle. Figure 5 shows the output fuzzy sets for a universe of discourse in the range from -0.6 to 0.6 radians.

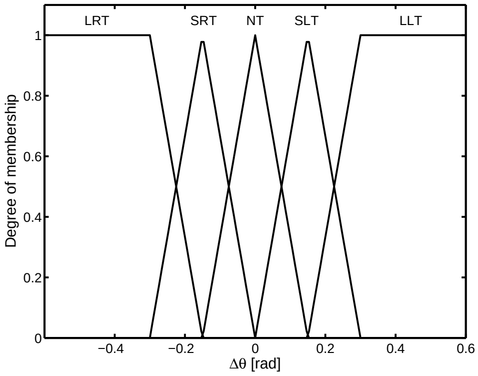

Fig. 5. Output Fuzzy Sets for the *Multivalued Algorithm*

The individual behaviors in the system respond to their sensory inputs by expressing the relative suitability α of each of the output fuzzy sets in controlling the robot according to the behavior's objective. Since there are five sets in the output, each behavior responds by triggering five outputs that express the relative suitability as fuzzy sets in the range [0,1]. The scores for each of the output fuzzy sets are combined by a fuzzy intersection operation. The control command for the system is derived from the output fuzzy set that receives the highest combined score from all the behaviors. Details of this algorithm are given in [11].

Figure 6 shows the structure of the heading control system that uses multivalued behaviors. In its advisory block there are four behaviors. The command block has the list of command alternatives, i.e., the Large Left Turn (LLT), Left Turn (LT), No Turn (NT), Right Turn (RT), and Large Right Turn (LRT). It receives votes from the individual behaviors α on the suitability of each of these command alternatives. The scores for each command are combined and the resultants are compared. The command that gets the highest score is implemented.

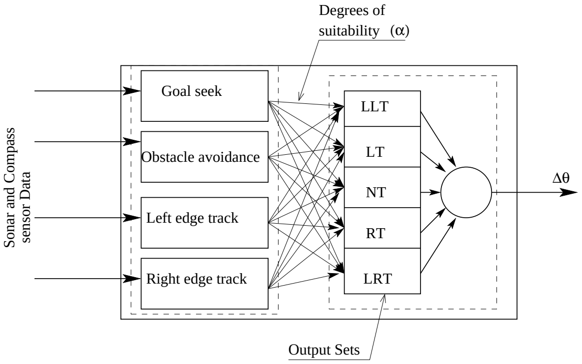

Fig. 6. Structure of the Heading Control System of the *Multivalued Algorithm*

IV. PERFORMANCE IN AN OFFICE APARTMENT

Numerical simulations were carried out to compare the performance of the navigation control systems based on univalued fuzzy behaviors and multivalued fuzzy behaviors in an office apartment. The robot was made to move from different locations at different orientations to goals placed in different locations. The performance of each control system was measured in terms of the distance covered by the robot and the curvature of the path followed in moving from one point to another. This section presents the results of this simulation. It starts with a description of the performance metrics.

A. Performance metrics

It is assumed that the best path is a straight line, i.e., with minimum path length P_L and zero curvature. Recall that the path length for the curve

$$y = f(x) \tag{1}$$

in the x-y plane between the points $(a, f(a))$ and $(b, f(b))$ is given by

$$P_L \triangleq \int_a^b \sqrt{1 + (f'(x))^2}dx. \tag{2}$$

The radius of curvature at any point $(x_i, f(x_i))$ along this curve is given by

$$\kappa(x_i) = \frac{f''(x_i)}{\left[1 + (f'(x_i))^2\right]^{\frac{3}{2}}}. \tag{3}$$

The bending energy associated with vehicle motion from point $(a, f(a))$ to $(b, f(b))$ along the path defined by (1) is related to the radius of curvature and can be computed as

$$B_E \triangleq \int_a^b \kappa^2(x)dx. \tag{4}$$

Less bending energy indicates that the motion is smooth while higher values of bending energy are associated with rough and erratic motions.

B. Observed results

The sample performance results for the two methods are shown in Table I, which summarizes the performance for 18 scenarios. The path length and the bending energy were determined by approximating (2) and (4) using the methods of [16]. There are some scenarios in which the algorithms failed, i.e., the robot was unable to reach the goal after getting stuck at some point. Performances for scenarios when both algorithms were successful are summarized in Figures 7 and 8. It is clear from these results that the multivalued behavior system had better performance by leading the robot along a shorter path with smaller bending energy. Figures 9 and 10 show the typical performance for the two algorithms (in successful scenarios) as they lead the robot from point (140,110) to point (15,10). The multivalued system tends to follow a more direct path while the univalued system tends to make many "guesses" along the way, hence following a "wavy" path. The tendency of the univalued system to follow "wavy" paths has also tended to lead the robot into deadlocks more often than saving it out of them. Figure 11 show one typical scenario when the robot was led by the univalued system to a deadlock while

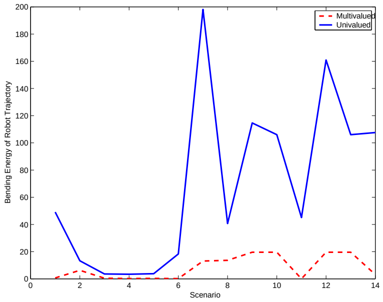

Fig. 7. Comparison of the Total Bending Energy Between the Univalued and the Multivalued Fuzzy System.

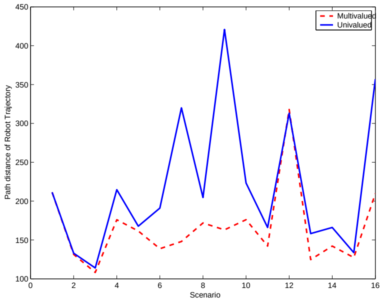

Fig. 8. Comparison Path Lengths Between the Univalued and the Multivalued Fuzzy System.

the multivalued system led it swiftly to its goal as shown in Figure 12.

On the other hand, the tendency of the multivalued system to follow more straight paths also can lead the robot to deadlocks as shown in Figure 13 whereas the path waviness of the univalued system was instrumental in avoiding that deadlock as shown in Figure 14. However, the frequency of failure for the multivalued system was less than that of the univalued system.

It was observed that all cases of failure occurred when the robot, the goal, or both were confined in enclosures and there was no direct pathway (or doorway) between the robot and the goal. Either the robot became trapped in the enclosure or the goal was located in some enclosure for which there was no direct access from the initial position of the robot. These failures are very common for almost all robotic systems that are only controlled by purely reactive behaviors. Figures 15 and 16 show the typical cases of failure when the robot started from an enclosure and the goal is located in an adjacent enclosure; there is no adjacent pathway between the two enclosures. For the scenario in these figures, the robot starts from an enclosed position (180, 80) with an initial orientation of $\frac{\pi}{2}$ radians, and the goal is at an enclosed position (140,15). For both systems, the failures can be avoided by using hybrid structure in which the reactive fuzzy behavior system is complemented by a deliberative system. The deliberative system should be able to provide memory and cognitive capabilities that enable the robot detect and avoid deadlocks.

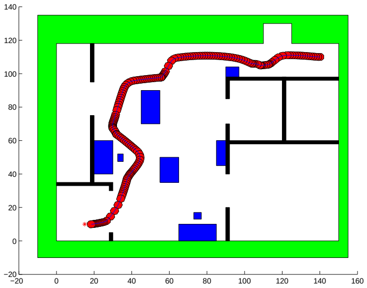

Fig. 9. Example of the Tendency of a Univalued System to Follow a Non-Straight Path.

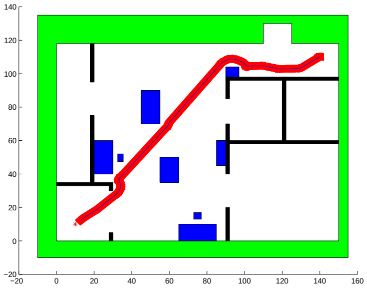

Fig. 10. Example of the Tendency of a Multivalued System to Follow a Straight Path.

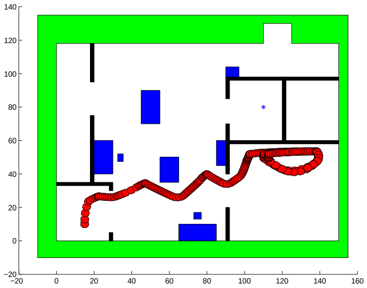

Fig. 11. Example of Deadlock in Univalued Behavior System Due to Waviness in the Path.

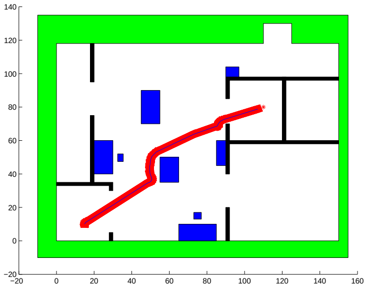

Fig. 12. Example of How a Straight Path Due to a Multivalued Behavior System Avoids Deadlocks.

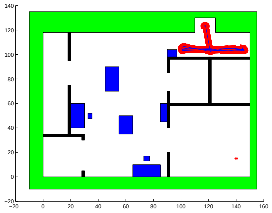

Fig. 13. Example of How a Straight Path Can Also Lead to Deadlocks for the Multivalued Behavior System.

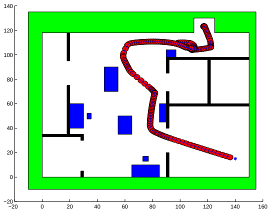

Fig. 14. Example of How Waviness of the Path Due to Univalued System Can Sometimes Help Avoid Deadlocks.

Scenario #	Path Length		Bending Enegry	
	Multivalued	Univalued	Multivalued	Univalued
1	162.0	211.4	0.641001	49.109226
2	131.4	132.9	6.295439	13.282470
3	108.2	114.0	0.472800	3.566526
4	176.0	214.7	0.1981985	3.396495
5	164.1	202.9	4.4205982	73.37135
6	138.7	191.0	3988.907694	25.255438
7	148.0	320.1	0.171330926	18.275666
8	157.4	fail	0.19707809	fail
9	132.5	fail	0.302545029	fail
10	121.4	fail	0.89611399	fail
11	fail	fail	fail	fail
12	fail	249.8	fail	956.762022
13	171.9	204.7	13.074766	198.152029
14	163.0	420.9	13.611470	40.7023553
15	176.1	223.3	19.555949	114.642486
16	142.1	166.1	19.555949	106.025333
17*	123.4	164.4	6.1230179	22.7568200
18	124.7	158.2	19.555949	160.908137
19	123.0	fail	0.3084862	fail
20	139.9	140.3	45.045834	20.528188
21	130.5	fail	18.853825	fail
22	fail	fail	fail	fail
23	127.4	133.7	3.1118613	107.559607
24	171.8	326.5	16.461359	27.3784875
25	209.1	357.0	1.7173921	55032.8194

TABLE I

SIMULATION PERFORMANCE RESULTS.

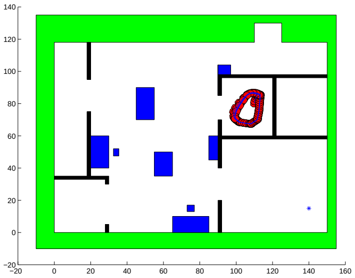

Fig. 15. Example of a Univalued Behavior Failure Due to Robot and Goal Being Enclosed Without Adjacent Pathways.

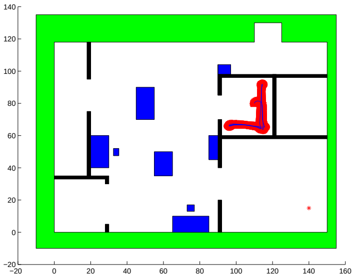

Fig. 16. Example of a Multivalued Behavior Failure Due to Robot and Goal Being Enclosed Without Adjacent Pathways.

V. CONCLUSIONS

Fuzzy behavior control systems are emerging as reliable tools for control autonomous ground vehicles; several algorithms have been designed to implement such systems. This paper has discussed the main differences between the univalued fuzzy behavior system and the multivalued fuzzy behavior system, and presented simulation results that compare the performance of two systems. It is seen that the multivalued system performs better in guiding the robot from one position to the goal by following a more straight path. On the other hand, by its structure the multivalued system is more computationally intensive than the univalued system. Both the univalued and multivalued behavior systems are susceptible to failure when the robot and the goal are confined in separate rooms that have no adjacent pathways. This failure can be avoided by using a hybrid system that uses the fuzzy behavior system along with deliberative control which enbales the robot to detect and hence avoid deadlocks.

DISCLAIMER

The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U. S. Government.

REFERENCES

[1] R. C. Arkin, *Behavior-Based Robotics*. Cambridge Massachusetts: MIT Press, 1998.

[2] A. Saffiotti, K. Konolige, and E. H. Ruspini, “Blending Reactivity and Goal Dorectedness in a Fuzzy Controller.” in *Proc. 2nd IEEE Conf. on Fuzzy Systems, San Fransisco, California*, June, 1993, pp. 134–139.

[3] F. Pin and Y. Watanabe, “Navigation of Mobile Robots using a Fuzzy Behaviorist Approach and Custom-Designed Inferencing Boards,” *Robotica*, vol. 12, no. 6, pp. 491–504, 1994.

[4] E. W. Tunstel, Jr., “Mobile Robot Autonomy via Hierarchical Fuzzy Behavior Control,” in *Proceedings of the 6th Intl. Symp. on Robotics & Manuf.; WAC’96, Montpellier, France*, May 1996, pp. 837–842.

[5] N. E. Hodge and M. B. Trabia, “Steering Fuzzy Logic Controller for an Autonomous Vehicle,” in *Proc. of IEEE Intl. Conf. on Robotics and Automation,Detroit Michigan*, vol. 3, May 1999, pp. 2482–2488.

[6] E. Aguirre and A. Gonzalez, “Fuzzy Behaviors For Mobile Robot Navigation. Design, Coordination and Fusion,” *International Journal of Approximate Reasoning*, vol. 25, pp. 225–289, 2000.

[7] R. Brooks, “A Robust Layered Control System for a Mobile Robot,” *IEEE Journal of Robotics and Automation*, pp. 14–23, April, 1986.

[8] J. R. D. W. Payton and D. M. Keirsey, “Plan Guided Reaction,” *In IEEE Transactions on Systems Man and Cybernetics*, vol. 20, no. 6, pp. 1370–1382, 1990.

[9] J. Yen and N. Pfluger, “Fuzzy Logic Extension to Payton and Rosenblatt’s Command Fusion Method for Mobile Robot Navigation,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 25, no. 6, pp. 971–978, 1995.

[10] J. Rosenblatt, “DAMN: Distributed Architecture for Mobile Navigation,” Ph.D. Thesis, Carnegie Mellon University Robotics Institute, Pittsburgh, PA, 1997.

[11] M. F. Selekwa and E. G. Collins, “A Centralized Fuzzy Behavior Control for Robot Navigation,” in *Proc. IEEE International Symposium on Intelligent Control, Houston Texas*, October 2003, pp. 602–607.

[12] E. Tunstel and M. Jamshidi, “Fuzzy Logic and Behavior Control Strategy for Autonomous Mobile Robot Mapping.” in *Proc. IEEE Conf. on Fuzzy Systems, Orlando Florida*, June, 1994, pp. 514–517.

[13] D. Driankov, H. Hellendoorn, and M. Reinfrank, *An Introduction to Fuzzy Control*. Berlin: Springer-Verlag, 1993.

[14] P. F. Muir and C. P. Neuman, “Kinematic modeling of wheeled mobile robots,” The Robotics Institute, Carnegie-Mellon University Pittsburgh, Pennsylvania 15213, Research Report CMU-RI-TR-86-12, June 1986.

[15] P. Goel, S. I. Roumeliotis, and G. S. Sukhatme, “Robust Localization Using Relative and Absolute Position Estimates,” in *Proc. IEEE International Conf.e on Intelligent Robots and Systems, Kyongju, South Korea*, October,1999, pp. 1134–1140.

[16] Center for Information & Multimedia Studies Kyoto University Japan, “Feature Extraction,” Kyoto University, Japan, `http://www.imel1.kuis.kyoto-u.ac.jp/ education/ dip/ feature/ node1.html`, Online Lecture Notes on Image Processing.