

## Research Article

## Development of Hardware Simulator to Validate Free-State based Navigation Algorithm of AGV

Harisha S.K, Ramakanth P. Kumar, M. Krishna, S.C. Sharma

Accepted 25 July 2013, Available online 01 August 2013, Vol.3, No.3 (August 2013)

### Abstract

Over the past few decades, the research in Autonomous Guided Vehicle/Autonomous Robot gained an extensive interest; this is due to the desire to replace manual work handling with AGV in dangerous and continuous tasks in industry. AGV's are vehicle type device capable of moving in an environment without human intervention. Autonomous navigation and coordination is associated with the external sensors that capture information from the environment through proximity sensors. As low cost sensors are not reliable while validating the algorithm for fuzzy based navigation in clustered environment, so development of hardware simulator which is useful in validating the free-state based navigation algorithm is discussed here. It's very difficult to validate the fuzzy based algorithm using unreliable sensor, in order to validate free-state based navigation algorithm this simulation model was developed.

**Keywords:** Freeway state, Fuzzy logic, Navigation, Simulator, Clustered environment.

### Introduction

The extensive research work on robotics has led to the greater achievement in this filed. The robot which mimics the human behavior and also the robot which navigates without any human interventions are the major areas of research. Autonomous robots are vehicles type devices capable of moving in an environment by making its own decisions with the predefined rules. Autonomous navigation is associated with the external sensors that capture information from the environment through distance or proximity sensors. The most common sensors are distance sensors (infrared, ultrasonic, laser, etc.) capable of detecting obstacles, walls, pits, targets and other features by measuring their distances. When advanced autonomous robots navigate within indoor environments (building, industries, offices), they have to be empowered with the ability to move through corridors, to follow walls, to avoid obstacles, to turn corners and to reach the target within building.

In efforts to articulate approaches that can handle real world uncertainty, researchers are often faced with the necessity of considering compromise between developing intelligent systems that are difficult to control, or adopting

a many number of assumptions that reduces the models design which are not enough to represent the actual system or the real world. The latter option control laws are typically valid only for systems where we can impose some assumptions. The option that involves complex real systems has been less prevalent due to the lack of analytical methods. Many approaches have been employed in navigation of Autonomous Guided Vehicles using mathematical model. Adaptive navigation is a method using differential equations to reach a pre-defined goal while avoiding obstacles but this analytical method is troublesome for complex behavior and environment.

Current research and application adopts soft computing such as fuzzy logic, genetic algorithm, and neural networks, they have showed the utility and ease of these paradigms for cognitive control of complex systems. In particular, fuzzy logic has proven to be a handy tool for handling real world uncertainty in case of robot exploration. Fuzzy logic does not need the mathematical description how the output functionality depends on the input. It is comparatively easy to adapt a system that deals with many situations without defining an analytical model of environment, by representing relations between inputs and outputs in an if-then rule. Fuzzy logic controllers provide a means of transforming linguistic control strategy based on expert knowledge into an automatic control strategy. It appears to be very useful for handling problems that are too complex to be analyzed by conventional quantitative techniques or when the available sources of information provide qualitative, approximate, or uncertain data. Fuzzy logic is suitable for multi-sensor, multi behavior fusion and integration.

F. A. Author Assistant Professor, Department of Mechanical Engineering, R V College, Engineering, Bangalore

S. B. Author, Professor & H.O.D, ISE Department, R V College of Engineering, Bangalore

T. C. Author is - Director, R&D, CMRTU, R V College of Engineering, Bangalore

F.D- Vice Chancellor, Tumkur University, Tumkur, Bangalore, Karnataka

## Literature Review

Behavior-based approaches have been established as a main alternative to conventional Autonomous Guided Vehicle control in recent year's (Anis Fatmi *et al*,2006; Wei Li *et al*,1994) . These approaches can be implemented and tested independently. The system architecture of behavior based approaches contains three levels i.e. high level, intermediate level and low level. A number of methods(E. Gat *et al*,1994; Wei Li *et al*,1994; M. Gerke *et al*,1992) based on behavior control has emerged since Brooks introduced the approach. There are two main issues in employing behavior control method, behavior module construction and behavior fusion. A behavior module is usually designed to be a reactive system, which maps a sensed situation to an action. Fuzzy logic control was introduced to construct the behaviors by many researchers (M. F. Selekwa *et al*,2003) and also to control the speed in various terrain condition (Kiwon Park *et al*,2007; Laura Ojeda *et al*,2002), as FLC requires no mathematical description and is able to represent human type knowledge on a control plant.

Behavior arbitration scheme introduced by Saffiotti *et al*., uses FLC, which allows one behavior at a time. This has a limitation of working in clustered environment where the Autonomous Guided Vehicle has to face more than one behavior at a time. There are various schemes developed for behavior integration, some of them are, e.g., based on Brooks sub-assumption architecture using a switching type of behavior arbitration. This method employs a priority scheme wherein the recommendation of only one behavior with the highest level is selected, while the other behaviors with lower priorities are ignored always. This type of approach leads to undesired performance in the following situation. Autonomous Guided Vehicle faces an obstacle situated directly in front of it. So the controller is decided to avoid obstacle behavior. The Autonomous Guided Vehicle has two options either to turn right or left, when controller executes top level behavior it takes turn either left or right, but there may be a situation that the target may exist opposite to the selected turn its decision hampers the progress of the seek-goal behavior.

Another technique (S.Parasuraman *et al*,2005) focuses on adding the inputs of each behavior using predetermined weighting factors. This leads to direct dispute in execution when multiple behaviors give adverse commands. To deal with these limitations, other strategies have employed fusion methodologies in which each behavior is allowed to provide the final output based on the situational context (J. Rosenblatt *et al*,1995;) . Saffiotti uses context-dependent blending, in which the current situation is used to decide between behaviors using fuzzy logic. Tunstel *et al* . is similar to , in that, adaptive hierarchies of multiple fuzzy behaviors are combined using the concept of degree of applicability in particular context. In this approach, certain behaviors are allowed to influence the overall behavior as required by the current situation and goal.

The behavior fusion method employed by implemented the approaches used in multi behavior

coordination. The differences in improved method are the behavior arbitration process, which allows one behavior at a time to be active. In (Bin Lei, Wenfeng *et al*,2007) multi agent or multi valued behavior method, independent behaviors are executed in a concurrent fashion, and depending on the situational context, outputs are blended together. Each behavior is assigned a weighting factor, and these factors are adjusted dynamically according the weight rules. The weighting factors determine the degree of influence of each behavior on the final motion command. In our method little contribution is added to the present state of knowledge, we adopted fuzzy based sensor perception grouping to make the fuzzy multiplexing simple when Autonomous Guided Vehicle is added with more number of behavior features.

### Behavior Control Concept Using LDR Sensor

Idea behind this concept generation is to provide easy methodology to develop low cost Autonomous Guided Vehicle for research as well as for learning purpose. This can easily adopted for executing any features or behavior of robots. Individual behavior listed above are tested by using low cost conceptualizations, as LDR is a low cost sensor and this is not much reliable because of its variation or sensitiveness to the light. As the LDR values with respect the place as well as time changes, (this is due to high sensitiveness to the light) it's very difficult to tune the LDR. So in this experimentation light emitting obstacle is considered (battery as source).

For all the feature execution we considered lighted torch as obstacles, target, pit, slope, surface texture (different intensity), Wall etc.

#### Goal Seeking Behavior (GSB)/Way Point Steering (WPS)

Eight LDR sensors are mounted on top of the vehicle for goal seeking/ way point steering purpose. While running experiment lighted torch beam is made to fall on different LDR sensor to check the way point steering, way points are differentiated by using different intensity light beam.

#### Obstacle Avoidance Behavior (OAB)

Eight LDR sensors are mounted on Front, right, left and back of the vehicle for obstacle avoidance, while running experiment lighted torch beam is made to fall on different LDR sensors, here LDR sensors treats light beam as obstacle. Algorithm execution is checked for single and multi-obstacle context.

#### PIT Avoidance Behavior (PAB))

Six extended LDR sensors are used for pit recognition and avoidance, sensors are tuned for two values one without light, that treated as surface and other value is tuned to the light beam that acts as pit. The experiment is conducted and validated for front, left and right side pit.

#### Localization Based Behavior (LBB) / Position Based

**Behavior (PBB)**

Eight LDR sensors are mounted on top of the vehicle for localization purpose, while running experiment lighted torch beam is made to fall on different LDR sensor to check the localization with respect to its own place. Localization sensors shares same waypoint and goal point sensors.

Low cost fuzzy based Autonomous Guided Vehicle uses LDR sensors but these sensors are sensitive for light. It is not possible to execute each behavior by the general concept using LDR sensors. This paper gives new strategy to check and validate the behavior using low cost LDR sensors. Torch is used as a light emitting type obstacle and also used for target steering source for different set of sensors.

Three groups of sensors are used in this work; target and localization sensor attracts the light source and moves towards it. Obstacle and pit avoidance behavior based group sensors repulse the light source and move away from the source and treats as obstacles.

Obstacle avoidance (repulsion to the light source), target steering (attraction to the light source), pit avoidance (downward directed sensors repulsion to the torch light), edge following (rotating robot starts following the light either on right or left which adopts the concept of wall or edge following behavior), slope riding, terrain based speed controller and combined behaviors are done using low cost sensors.

To investigate the effectiveness of the proposed approach, the light dependent sensor based robot is tested is shown in Fig.11 to Fig.14 The robot is placed at the starting point and a torch lights are positioned at different places to check the behavior of robot for obstructions and target steering. The separate sets of sensors are used to get the different behavior actions.

As a comparison, a non-fuzzy (without range and fuzzy values) robot is developed and tested on the same table to evaluate the performance of the fuzzy logic controlled robot. From the experimental results, it is concluded that the fuzzy robot is more efficient in detecting and steering towards the light source without colliding with the walls within a shorter period. On the other hand, although the non-fuzzy robot finally successful to track the light source, it takes a relatively longer time to accomplish the task as it tends to rotates around the table and deviates from the path after sometimes. The efficiency of the fuzzy logic based robot can be used to demonstrate the concepts, real world applications and also this concept is helpful to motivate the students to work in the field of autonomous robot.

**Free State-Based Control Algorithm**

The FSBCA algorithm is developed for complex environment where robot comes across multiple behaviors at a time. The inputs from different sensor groups (Group-A, Group-B and Group-C) are received by micro controller in parallel and it takes action depending on the situation, this algorithm is suitable for crowded

environment. Free State based control algorithm always looks for the free space rather than identifying obstacle. If in case it contains many Free states it takes path which is nearer to the way point or a goal point.

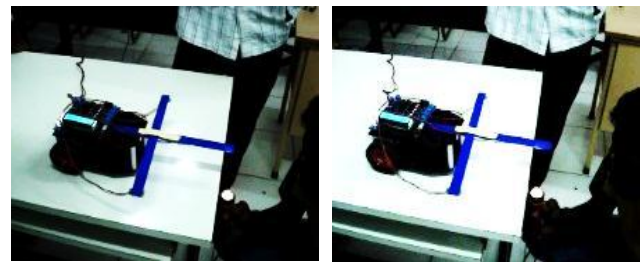


Fig. 11 Concept of PAB



Fig. 12 Concept of TSB

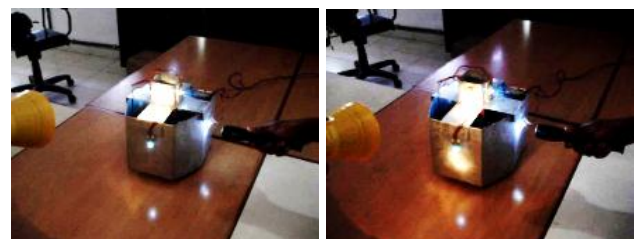


Fig. 13 Concept of TSB & OAB

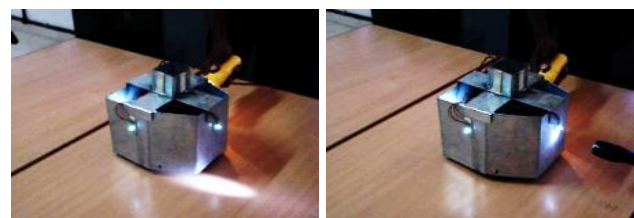


Fig. 14 Concept of OAB

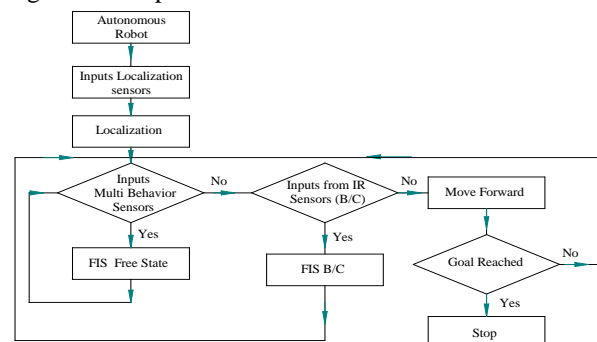


Fig. 1 Flow chart- Free State based speed control algorithm

### Fuzzy Inference System of Free-State Based Navigation

Fuzzy inference engine of Free State contain sensor readings and way point position as inputs and speed of right and left wheel as outputs for successful navigation. Free State navigation contains Group-A Group-B and Group-C sensors for coordinating all behaviors in clustered environment.

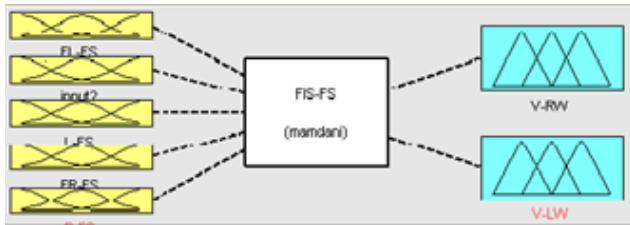


Fig. 2 Fuzzy inference engine for Free State

Membership Function: Member ship function of infrared extended sensors shown in Fig.3 represented with F-FS, FR-FS, FL-FS, R-FS, and L-FS is represented with two linguistic variables [Free (F), Not Free (NF)].

Inputs from sensor to microcontroller

F-FS, FR-FS, FL-FS, R-FS, L-FS = [Free (F), Not Free (NF)]

Membership function. (Both IR (8+1) and IRE (6) sensors

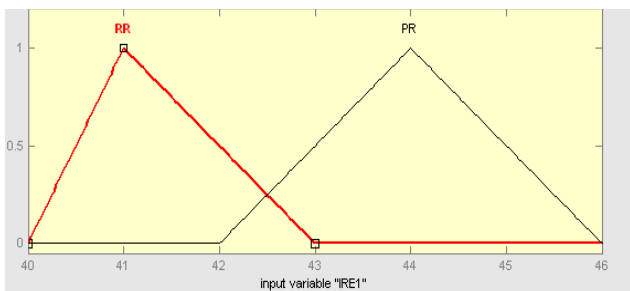


Fig. 3 Member ship function for input IRE sensor

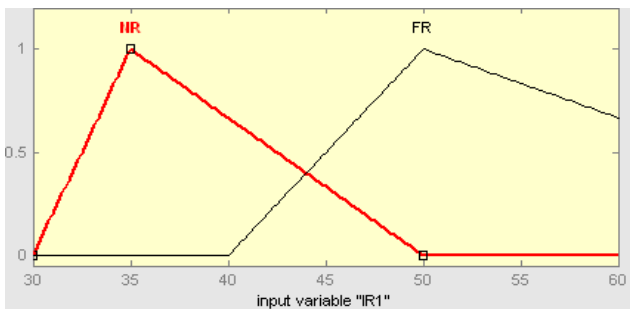


Fig. 4 Member ship function for input IR sensor

Combination of GR-B and GR-C sensors inputs are considered to judge the Free State group sensors (L-FS, FL-FS, F-FS, FR-FS and R-FS).

In Free State logic - MIN (GR-C) sensors and MAX (GR-B) sensors considered to indicate the obstruction, reverse of that indicates Free State condition.

Member ship function of center sensors (CE1-CE8) represented in Fig. 5 with five linguistic variables [Front (F), Front Right (FR), Front Left (FL), Right (R), Left (L)]. These sensors are grouped in GR-A, which is both for localization and waypoint steering.

Inputs from Center Sensors (CS): Group-A sensors

WP = [Front (F), Front Right (FR), Front Left (FL), Right (R), Left (L)]

Membership function:

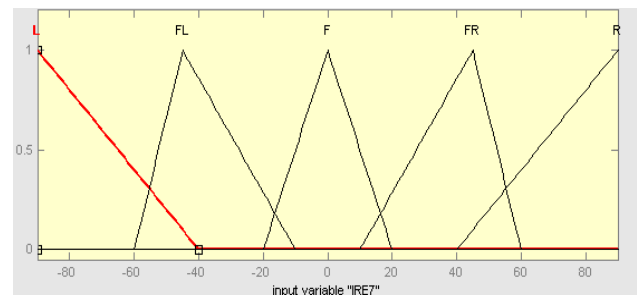


Fig. 5 Member ship function for input LDR sensors

Member ship functions of the two outputs are Right wheel speed and left wheel speed which is represented with two linguistic variables [Stop (S), Turn (T)] shown in fig. 6. The target steering is based on differential drive mechanism.

Outputs to differential drive motors

Right Wheel (RW) = [Stop (S), Turn (T)]

Left Wheel (LW) = [Stop (S), Turn (T)]

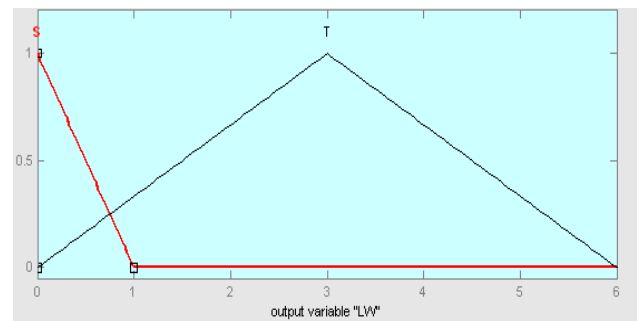


Fig. 6 Member ship function for output wheels

Rules are defined by using inputs of 8 IR (obstacle detection), 6 IR (pit detection) and 8 LDR (waypoint) sensors. These rules are executed using fuzzy toolbox then it is converted in to logical form for the program. Fig.7 shows the rule viewer for GR-FS sensors, the surface viewer graph is shown in result. Combination of GR-B and GR-C sensors inputs are considered to judge the Free State group sensors (L-FS, FL-FS, F-FS, FR-FS and R-FS).

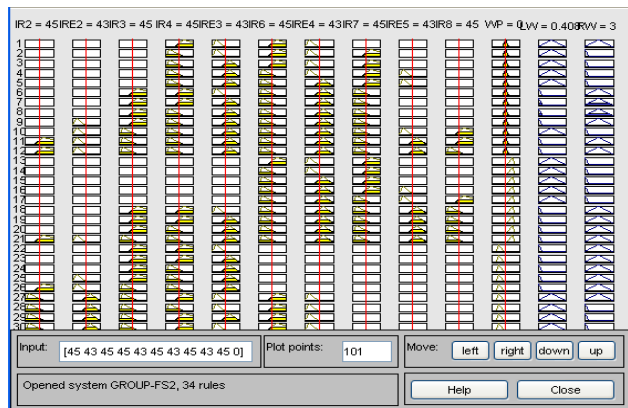


Fig. 7 Rule viewer graph for FIS-B group

Fig. 8 represents the graph of X vs Y in cluttered environment, Path traced in real time experiment was plotted in a graph. The cluttered environment consists of pits, obstacles, walls, slopes and target. The description of position and specification of the objects are listed in Table 8.

*Description: Complex Environment*

Table 8.7: Complex environment description

Sl. No.	Objects	Position	Specification
1	Starting Point	(1,1)	-
2	Obstacles 1-7	3.5x1.5, 13.5x4.5, 6.5x8.5, 10.5x9.5, 11.5x14.5, 12.5x12.5, 16.5x10.5	1 x 1
3	Walls 1-3	4.5x11, 6x6, 13.5x8.5	1.5x3, 6x1.5, 3.5x1
4	Slopes 1-2	3x4.5, 15.5x15	4 x 1, 4 x 2
5	Pits 1-6	2.5x11.5, 5.5x2.5, 7.5x13.5, 8.5x1.5 10.5x5.5, 15.5x8.5	1 x 1
6	Goal Point	(17.5,16.5)	-

**Hardware Simulator**

Free state based navigation has high rate of successful navigation. It is very difficult to validate the algorithm using unreliable sensor based autonomous robot. In order to validate Free State based navigation algorithm the simulation model is introduced here. The simulation model consists of three main components 1) Robotic Manipulator/Arm 2) Micro controller contained circuit board 3) Input board (which contains two rows of buttons).

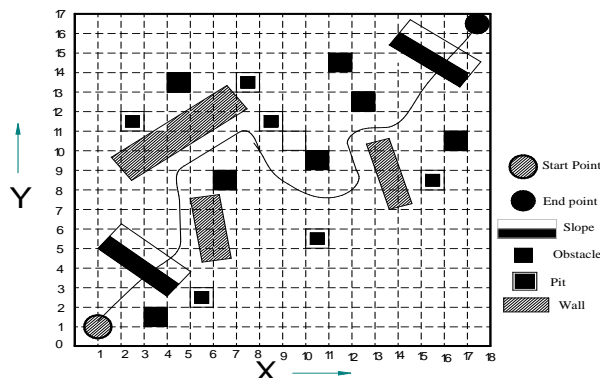


Fig. 8: Graphical representation of FSBCA based navigation

Micro controller is programmed by using Free State based logic and input to this micro controller is done through manual buttons. By selecting different goal direction and available Free states, the robotic arm selects and validates desired direction.

Many types of software simulators are available for Autonomous Guided Vehicle navigation but it is not possible to validate the generated algorithm for all multi featured behavior. Fig.9 shows the block diagram of hardware simulator and Fig.10 shows the working model of hardware simulator which replaces the sensors input by manual input buttons. This can be used to validate any algorithm generated for Autonomous Guided Vehicle navigation.

Through our model we have introduced hardware simulator to test the complex robotic systems. Here the program validation of Autonomous Guided Vehicle is done by using the hardware simulator. The toggle switches in our model have acted as a replacement for sensors as there is goal defining and freeway defining switches. The selected freeway switches are the only directions where the robot can move and the switches which have not been selected acts as obstacles in the real world functioning of an autonomous robot. The program here decides whether the way towards the goal is free otherwise the free way which is close to the goal will be selected. For the future implementation the same concept can be extended for very close angles (less than 45 deg). This will be useful for real time application.

Hardware simulator designed for validating the algorithm contains

- Input buttons (two rows): One set of buttons for selecting freeways and other set for selecting goal or way point.
- Micro controller with PLC board: Atmel microcontroller is used along with the microcontroller board; micro controller board contains LED's for indicating pulse, H-bridge to obtain the required control of motors, USB burner for flushing program.
- Two servo motor with tachometer: As the AGV movement is differential drive based. In the hardware simulator we used two servomotors with tachometer to indicate the speed of the wheels. The wheels are

not having contact with the surface, so error due to friction can be avoided here.



Fig. 9 Block diagram of hardware simulator



Fig. 10 Hardware Simulator

The validation of Free State based navigation algorithm is done using hardware model. Free State based logical program is flushed into the micro controller to check the validation. To run this simulator model, first select or press the goal button and then select the freeway buttons (while selecting freeway buttons option is either to select single button or up to four buttons leaving at least one freeway path) after selecting input buttons micro controller takes decision and it make the Right and left wheel which are mounted without surface contact, their speed is measured by using tachometer. This simulation results holds good for navigating autonomous robot.

### Simulation Result

The validation of Free State based navigation algorithm is done using simulation model. Free State based logical program is flushed in to the micro controller to check the validation. After selecting different goals and different freeway buttons the two wheels takes the inputs through microcontroller, as the mechanism of steering is based on differential drive, the tachometer reading gives the desired output. This simulation results holds good for navigating the autonomous robot. All possible chances of goal selection and freeways are tested. The results in all the cases are same as the rules inferred in fuzzy inference system.

With this work it is proved that the validation of algorithm can be carried out using hardware simulator. Here the program validation of the robot is done by conducting trial tests. The toggle switches in our model have acted as a replacement for sensors, as there is goal defining and freeway defining switches. The selected freeway switches are the only directions where the robot can move and the switches which have not been pressed act as obstacles in the real world functioning of an autonomous robot. The control program in the microcontroller decides which one of the freeway paths is closest to the goal and then it

moves towards that direction, by selecting all possible ways the algorithm of freeway state based robot validated.

### Conclusions

Many software Simulators are available in the practice for Autonomous Guided Vehicle or robot navigation but the coding is difficult when the AGV has to face the tough environment or more number of behaviors in the environment. The developed hardware simulator is very easy to validate the freeway state based navigation algorithms. The main reasons for developing hardware simulator is to avoid other kinds of problem such as unreliability of low cost sensors and problems in the hardware parts of the robot.

### Acknowledgment

The authors would like to acknowledge the contributions made by the co-authors of this paper. We would also like to acknowledge support taken from the management of RVCE.

### References

- Anis Fatmi, Amur Al Yahmadi, Lazhar Khriji, Nouri Masmoudi (Oct 2006), A Fuzzy Logic Based Navigation of a Autonomous Robot *Proceedings of World Academy of Science, Engineering and Technology*, volume 15, issn 1307-6884
- A. Saffotti (1997), Fuzzy logic in autonomous robotics: Behavior coordination, in *Proc. 6th Int. Conf. Fuzzy Systems, Barcelona, Spain*, vol.1, pp. 573-578.
- Bin Lei, Wenfeng Li (March 2007), A Fuzzy Behaviors Fusion Algorithm for Mobile Robot Real-time Path Planning in Unknown Environment, *Proceedings of the IEEE International Conference on Integration Technology*, , Shenzhen, China, pp-173-178
- E. Gat, R. Desai, et al. (1994), Behavior Control For Exploration of Planetary Surfaces, *IEEE Transactions on Robotics and Automation*, vol. 10, no.4, pp. 490-503.
- Jafreezal Jaafar, J. Eric McKenzie (2008), A Fuzzy Action Selection Method for Virtual Agent Navigation in Unknown Virtual Environments, *Journal of Uncertain Systems* Vol.2, No.2, pp.144-154.
- J. Rosenblatt (1995), DAMN: A distributed architecture for mobile navigation, in *Proc. AAAI spring SYMP. Lessons Learns for Implemented software Architecture for Physical Agents*, Palo Alto, CA, PP 167-178.
- Kiwon Park, Nian Zhang (2007), Behavior-Based Autonomous Robot Navigation on Challenging Terrain: A Dual Fuzzy Logic Approach *Proceedings of the IEEE Symposium on Foundations of Computational Intelligence (FOCI)* , pp-239-244.
- Laura Ojeda, Johann Borenstein ( May 2002), FLEXnav: Fuzzy Logic Expert Rule-based Position Estimation for Autonomous Guided Vehicles on Rugged Terrain *Proceedings of the IEEE International Conference on Robotics and Automation*. Washington DC, USA, 10 , pp. 317-322
- Meng Wang, James N.K. LIU (Aug 2004), Autonomous Robot Navigation using Fuzzy Logic Controller, *Proceedings of the Third International Conference on Machine Learning and Cybernetics, Shanghai*, 26-29 pp-691-696.

- M. F. Selekwa and E. G. Collins (Oct 2003), A Centralized Fuzzy Behavior Control for Robot Navigation, in Proc. *IEEE International Symposium on Intelligent Control, Houston Texas*, pp. 602–607.
- M. Gerke and H. Hoyer (1992), Fuzzy Collision Avoidance for Industrial Robot, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 510-517.
- Panzarasa P., Jennings N. R., and Norman T. J. (2002), Formalising Collaborative Decision-Making and Practical Reasoning in Multi-agent Systems. *Journal of Logic and Computation* , 12(1), pp.55-117.
- S.Parasuraman, V.Ganapathy, Bijan Shirinzadeh (2005) Behavior based Robot Navigation by AI Techniques: Behavior selection and resolving *behavior conflicts using  $\alpha$  – level Fuzzy Inference System Proc. of the Intl. Conf. on Automation, Robotics and Autonomous System ARAS'05.*
- S.Parasuraman, V.Ganapathy, Bijan Shirinzadeh (June 2005)Behaviour based Autonomous Robot Navigation Technique using AI System: Experimental Investigation on Active Media Pioneer Robot *Special issue IIUM-Engineering Journal, Malaysia* ref: vol. 104.
- S.Parasuraman, V.Ganapathy, Bijan Shirinzadeh (2004), Behaviour based Autonomous Robot Navigation Technique for Real World Environments using Fuzzy Logic System *IEEE International Conference on Systems, Man and Cybernetics*, pp.3359-3364, 2004.
- Wei Li (Oct 1994), Fuzzy Logic Based Robot Navigation in Uncertain Environments by Multi sensor Integration *Proceedings of the IEEE International Conference on Multi sensor Fusion and Integration for Intelligent Systems (MF1'94) Las Vega, NV Oct. 2-5., Pp.259-265.*