Implementing Behaviour Based Control in an Autonomous Mine Detecting Robot (AMDR)

Ampikathasan Aravinthan and Thrishantha Nanayakkara
Department of Electrical Engineering, University of Moratuwa, Sri Lanka.

ABSTRACT

The goal of our research is to develop a low cost mine detecting autonomous mobile platform to search for landmines in the minefields. The robot is developed to suit the local conditions, small in size and has only required sensors to achieve the goal assigned. In this paper, a novel Behavior Based Controller is presented for a nonholonomic mobile robot like AMDR used for landmine detection in mine fields. The controller of the robot provides basic facilities to the mine detector, which is rigidly fixed on the robot, to ensure complete coverage of mine field while avoiding obstacles and try to minimise the time spent in searching landmines wherever possible. The controller has Move, keep In Line, Avoid Obstacles as low level behaviors. Also the controller has safety behaviors embedded while some top level behaviors facilitate the coordination among the other robots in the mine field and a remote operator outside mine field. The most of the behaviors are encoded using Fuzzy Sets and the coordination of the behaviors is done through Subsumption Architecture. The SimRobot, a Matlab based mobile robot simulator was used to simulate the robot's behaviors individually in real world situations. The effectiveness of the proposed controller is demonstrated by simulation studies in all the foreseeable real world situations.

1. INTRODUCTION

Humanitarian Demining is a very urgent and necessary activity to be carried out in land mine affected countries. Most of the landmine affected countries are developing countries in Asian and Africa continents. These countries are spending a lot of money to other developed countries which has the technology to detect and remove landmines in removing landmines and for rehabilitation works to people who were affected by landmine explosion. The cost of a landmine varies about $ 3 to $4, but to remove a landmine it costs about $1000.

Demining is very risky operation when human involves directly. Also it is very slow process as there is no suitable mine detector available and due to human involvement. All of the above process can be replaced with a fully autonomous mine detecting robot. But there is no low cost mine detecting mobile robot available today to be used in landmine affected developing countries. Also the environmental conditions (very hot, very dusty and less humid) in most of the landmine affected countries prevent the robots made in completely different environment conditions to be used in humanitarian Demining.

To over come all these problems there must be a locally developed technology in each landmine affected countries which could ease the landmine clearing process. The cost of developing that technology must be as low as possible and also human involvement should be less. An attempt was made in Industrial Automation Research Centre at University of Moratuwa to build an Autonomous Mine Detecting Mobile Robot which will solve landmine problem in Sri Lanka.

An autonomous robot is a physical system that has its own resources to operate independently in a dynamically changing world. Many researchers have been focused on robot navigation in achieving some goal assigned while avoiding obstacles and other critical situations, within a structured or man made environment[2]. Very few have done research on robot navigation on field robots which faces lot of different kind of obstacles and changes in the environment.

Some examples are “Shakey”-constructed in the late 1960s at Stanford Research Institute, “CMU Rover”- constructed to test stereo vision as a means for navigation in 1980[2]. Very few robots are available for real outdoor works. Experimental results in above researches show that the performance of the behavior based control architecture is very much superior to conventional hierarchical controllers.

All of the researches have had considered all kind of obstacles the robot faces as one and most of the time try to avoid it with some clearance[1]. But this is not allowed for the robots searching for landmines. The robot should cover all the areas in the mine field while avoiding obstacles. The complete coverage cannot be achieved if the robot does not have the basic features of complete coverage i.e. let’s take obstacle avoiding feature as an example of a mobile robot if it uses sonar sensor for detecting obstacles it should avoid the obstacle by having some clearance between obstacle and the robot, same time if we use bumper sensors for obstacle detection we can avoid it with less loss in coverage.

In this paper a novel Fuzzy Logic Based Behavior Controller proposed for real time navigation of AMDR in mine field. The algorithm guarantees maximum coverage of mine field and adjusts velocities according to different terrain conditions in order not to exceed speed limit set by mine detector. This achieved by taking advantage of bumper sensors fixed inside the buffer of the AMDR, sonar sensors and shaft encoders.

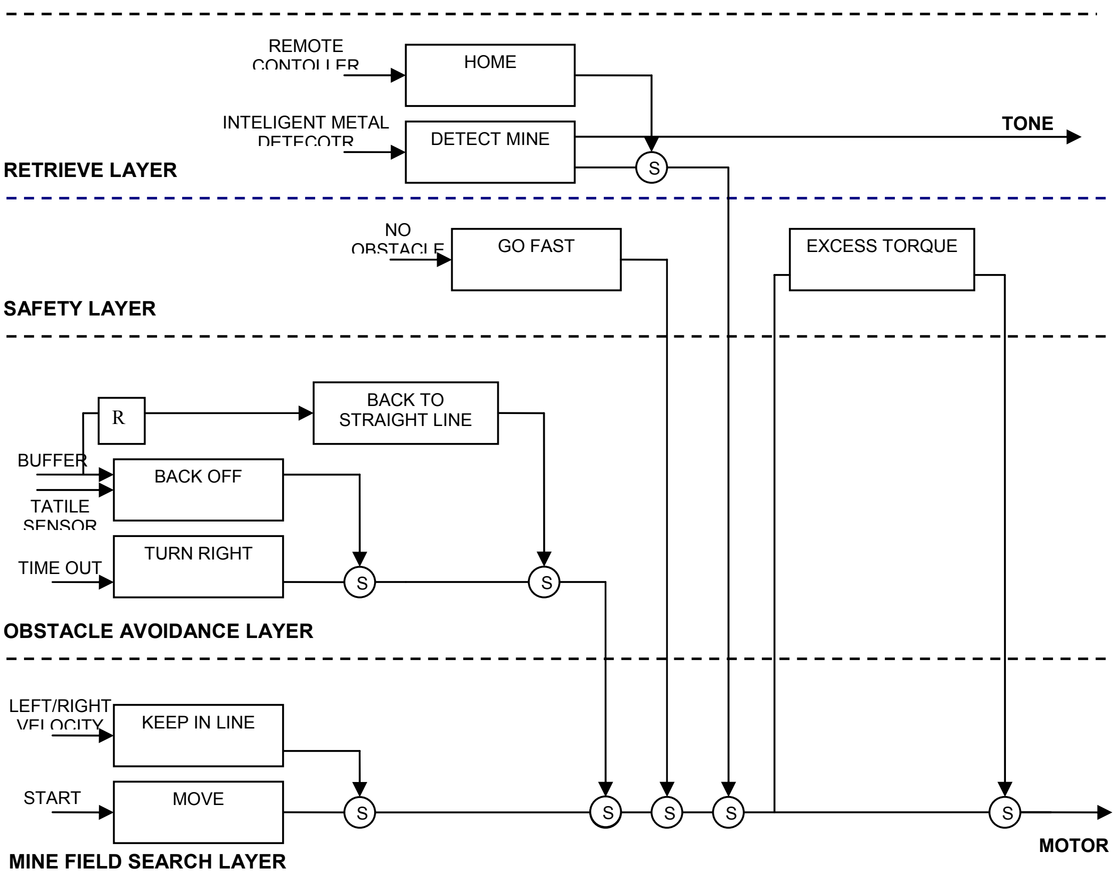
In the following sections we will describe about the work we have done at IARC with AMDR. Section 2 describes about the proposed controller of the AMDR robot which makes it to perform an effective landmine search. In section 3 the simulation setup and results are presented. Finally we present some conclusions based on the simulations that we have performed.

2. PROPOSED BEHAVIOR BASED CONTOLLER

The Autonomous Mine Detecting Robot was designed and built at IARC with locally available products and technology. The robot is almost in round shape with 15 cm radius. The robot has two sensed actuated wheels driven by DC motors and one non actuated non sensed caster wheel. In front the robot has a buffer mounted using springs and has three bumper switches. Also AMDR has two sonar sensors and a CCD camera. The robot controller is designed using two PIC18F452 micro controllers networked with all the sensors and actuators are connected to different ports of the microcontroller. The power to the controller is supplied with 9V dry cell batteries and motor gets power from a liquid cell of 12V. The whole weight of the robot is 3.850Kg. This weight is far bellow the minimum force required to activate an anti-personal landmine. Also the robot is made small such that it can move easily through thick vegetation which is often the case in mine fields.

Conventional hierarchical control system was first replaced by layered control system based on Subsumption Architecture (A branch Behavior Based Control System) when Brooks presented his first paper in 1986[3]. From 1986 onwards the Behavior Based Control (BBC) system is being attracted by several researchers in AI field. As a result several areas of AI have been absorbed in and became a very successful area in Automation and Control especially for robots[4]. Behaviors are the fundamental building blocks of BBC and the basis for a behavior arises from different view points for example it may a copy of an animal’s behavior or it may a behavior designed experimentally. Behaviors are simply taking sensory inputs from sensors and producing some outputs to the actuators. The mapping of input to output is done in two ways called discrete encoding and continuous encoding. Discrete encoding may be a rule base based on Fuzzy Logic or it may be some finite set of (situation, response) pair. Fuzzy Logic based encoding provides robust behavior while facilitating online adaptation.

The assembling of behaviors is necessary when we have more than one behaviors producing output to the same actuator at the same time. Assembling is done in two ways in behavior based control system, one is called competitive where the behaviors are assembled in priority order and when the behaviors with high priority is active it suppress or inhibits the output of the low priority behavior output. The other one is cooperative where the outputs or active behaviors are considered in some way (may be just an addition) to produce an out put to the actuators.



Figure(1) Proposed Controller base on Subsumption Architecture

The proposed controller consists of competing behaviors which generates torques for motors. As figure(1) shows the controller for the AMDR robot for landmine search consist of seven behaviors. These are grouped into four tasks specific layers. The boxes denote behaviors or behavior states which get inputs from sensors or from other behaviors and generate output to the actuators or to other behaviors. All behaviors interact through suppressor nodes, marked with (S) which effectively suppress the signal through it and replaces it with the signal coming from the top. The robot can perform useful actions with low level implemented but when behaviors in higher level added the performance improves.

The search mine field layer in which the robot search for mines while moving in a straight line. This layer has two behaviors called MOVE and KEEP IN LINE. The MOVE behavior just starts the robot form rest and makes it to move the robot in set velocity. The KEEP IN LINE behavior monitors robot to move in a straight line by getting input from shaft encoders and other behaviors and sets the left and right velocity accordingly.

The behavior uses Fuzzy Logic based controller to determine how the robot's left and right wheel velocities to be adjusted to keep the robot in straight line and to set the robot to the correct speed. The two inputs to this behaviors are the "set velocity" and "shaft encoder counts". The speed is adjusted alternatively to the left wheel and then to the right wheel. The left wheel is adjusted for set speed always while the right wheel is adjusted to match with the speed of the left wheel. By doing like this we can keep the robot to move in straight line as well as to move at set velocity. The "set velocity" is represented by three fuzzy sets {Bellow_Set_Vel, Set_Vel, Above_Set_Vel}. The "shaft encoder counts" is represented by five fuzzy sets {High_Pos_Error, Low_Pos_Error, Zero_Error, Low_Neg_Error, High_Neg_Error}. And the "velocity correction" contains five fuzzy sets {High_Neg_Adj, Low_Neg_Adj, Zero_Adj, Low_Pos_Adj, High_Pos_Adj}.

The left wheel speed rules are as follows:

If V_f is Bellow_Set_Vel **Then** *Vset* is Low_Pos_Adj

If V_f is Set_Vel **Then** *Vset* is Zero

If V_f is Above_Set_Vel **Then** *Vset* is Low_Neg_Adj

The right wheel speed synchronization rules are as follows:

If E_c is High_Pos_Error **Then** *Vset* is High_Neg_Adj

If E_c is Low_Pos_Error **Then** *Vset* is Low_Pos_Adj

If E_c is Zero_Error **Then** *Vset* is Zero_Adj

If E_c is Low_Neg_Error **Then** *Vset* is Low_Neg_Adj

If E_c is High_Neg_Error **Then** *Vset* is High_Neg_Adj

The torque is calculated by the fuzzy controller is directly written to the left and right motors. The Fuzzy controller is capable to learn in real time which will improve the performance over the time.

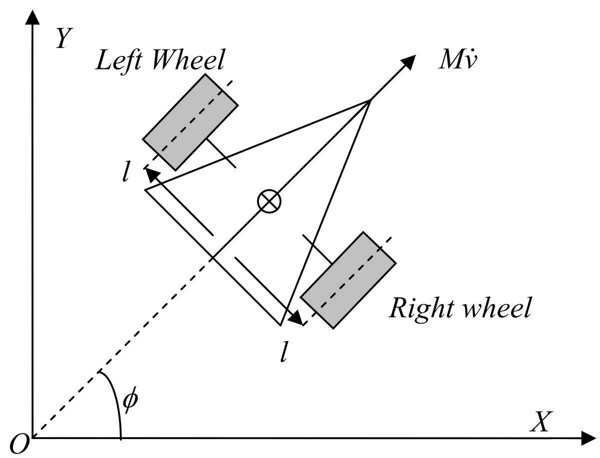
The next level is called Avoid Obstacle Layer in which the robot takes certain action when it hits with hard obstacles. The robot sees real world obstacles in two different categories called "soft obstacle" and "hard obstacle". Soft obstacles are the ones which the AMDR robot can go over. Hard obstacles are the ones the robot has to take actions to avoid while minimising uncovered area. This behavior is encoded as a finite set of (situation response) pairs.

When the robot gets the signals from bumper sensors the Avoid obstacle behavior activates with BACK OFF state and proposes torque to the robot to stay for some time and move back for specified time and then stop. Once the timeout for this behavior has finished the behavior state changes to TURN RIGHT state and produces torques to turn the robot by 45 degrees and move forward for specified time, turn by 90 degrees in opposite direction and stop. Soon after this state's time elapsed the control jumps to BACK TO STRAIGHT LINE state which makes the robot to come back to its previous line of travel. When ever the robot encounters a hard obstacle this sequence of behaviors are activated even it is within this state transition process.

Safety layer has two behaviors which mainly concerns mainly about the robot's safety. The GOFAST behavior get signal from sonar sensors. When sonar detects the obstacles position above 1.5m then the GOFAST behavior directs the robot to move in "High Speed" mode which effectively speedup the process of searching mines. EXCESS TORQUE behavior controls torque to the motor not to exceed the limit.

Retrieve Layer has DETECT MINE and HOME behavior. HOME behavior provides room for other robots to communicate with each other and to execute commands from a remote operator. DETECT MINE gets input from mine detector and gives alarms and waits to next action from the remote operator.

3. SIMULATION SETUP



Figure(2) The schematic diagram of the Mobile robot

In this section the proposed controller is applied to the AMDR robot model for simulation purpose. The simulation is done to one of the important behavior which can be simulated in SimRobot simulator. As shown in the figure(2) the state space model of a nonholonomic mobile robot can be written as[5]

$$\dot{x}(t) = Ax(t) + Bu(t)$$

Where,

$$\dot{x}(t) = \left[v(t)\phi(t)\dot{\phi}(t)\right]^T$$

be the state variable vector and

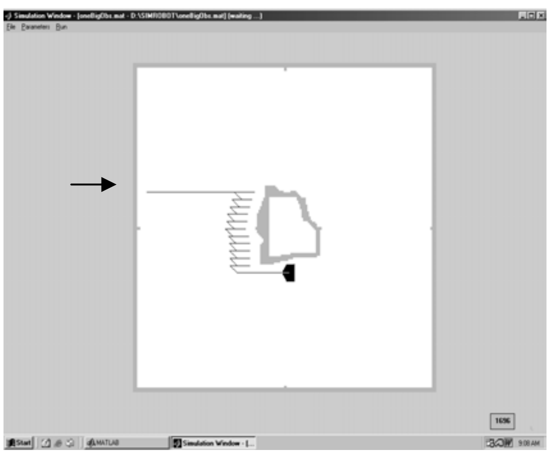
$$u(t) = [u_r u_l]$$

with
$$A = \begin{bmatrix} \frac{-2c}{Mr^2 + 2I_w} & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & \frac{-2cl^2}{I_v r^2 + 2I_w l^2} \end{bmatrix},$$

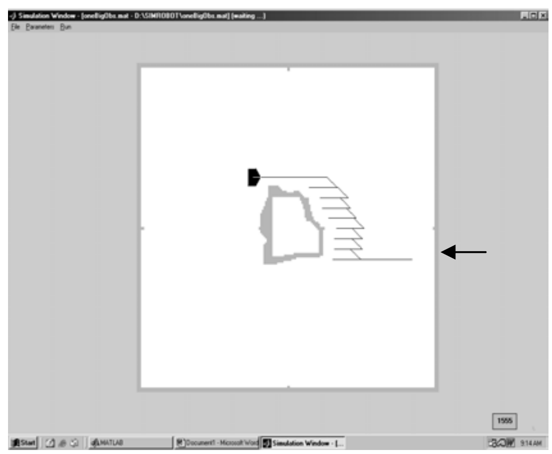
$$B = \begin{bmatrix} \frac{-kr}{Mr^2 + 2I_w} & \frac{-kr}{Mr^2 + 2I_w} \\ 0 & 0 \\ \frac{-krl}{I_v r^2 + 2I_w l^2} & \frac{-krl}{I_v r^2 + 2I_w l^2} \end{bmatrix}.$$

Physical parameters of AMDR are given by:
Moment of inertia around the cg of robot $I_v = 0.1682 Kgm^2$
Mass of the robot $M = 3.850 Kg$
Distance between the cg and the left/right wheel $l = 0.120m$
Radius of the wheel $r = 0.07m$
Moment of Inertial of wheel $I_w = 0.0034 Kgm^2$
Driving gain factor $K = 90$
Viscous friction factor of wheel $c = 0.0497 Kgm^2 / s$

The simulation is done for Avoid Obstacle behavior form the proposed controller. The initial positions of the robot were changed and appropriateness of the behavior is accessed.



Figure(3a) The robot's path when the robot is in the left side of the obstacle



Figure(3b) The robot's path when the robot is in the right side of the obstacle

The figure(3a) and figure(3b) shows path of the robot when the it is on either sides of the obstacle. The robot tries to avoid the obstacle by closely following the obstacle's shape. The arrow indicates the starting position of the robot. White space is robot's environment and the obstacle is in the middle. The obstacle is 5 times larger than the robot and the dimensions of the obstacle are 1.5 m in length and 1.0 m in width.

4. CONCLUSION

The proposed behavior based navigation strategy using fuzzy logic rules and other simple rules has major advantages over the conventional hierarchical algorithms. First if we use fuzzy logic to encode behaviors it provides us with a robust behavior and learning capability to the behavior. For the simulated behavior which is the crucial one the controller is well behaving for different types of obstacles as shown in the figures.

The simulator does not have enough features to simulate all the proposed behaviors. This restricted us to do the simulation to only to the important behavior. The simulation for the rest of the behaviors and the implementation of this controller is kept for future work.

5. REFERENCE

1. Homayoun Seraji, "Behavior-Based Robot Navigation on Challenging Terrain: A Fuzzy Logic Approach" IEEE Transactions on Robotics and Automation, Vol. 18, No 3. June 2002.

2. Rodanld C. Arkin (1991) Behavior-Based Robots, The MIT Press, Cambridge, Massachusetts.

3. Brooks, R. "A Robust Layered Control System for a Mobile Robot"*IEEE Journal of Robotics and Automation*" Vol.RA-2, NO.1 1, March 1986.

4. Perez, M.C. "A Proposal of a Behavior Based Control Architecture with Reinforcement Learning for an Autonomous Underwater Robot" *PhD Thesis*, University of Girona, May 2003.

5. D.P.Thrishantha Nanayakkara, Keigo Watnabe and Kazuo Kiguchi "Evolutionary Learning of a Fuzzy Behavior Base Controller for a Nonholonomic Mobile Robot in a class of Dynamic Environments" Journal of Intelligent and Robotic Systems 32: 255-277, 2001.