# A Behaviour Based Hierarchical Fuzzy Control Architecture For Agricultural Real Time Autonomous Mobile Robots

**Hani Hagras* , Victor Callaghan* , Martin Colley*, Malcom Carr-West**.**

***The Computer Science Department**
**Essex University, Wivenhoe Park**
**Colchester CO43SQ, England (U.K).**

****The Agricultural Engineering Department**
**University College Writtle, England (U.K).**

**Abstract.** This paper describes the preparatory design and construction work for a of real-time autonomous mobile robot aimed at navigating in real farms with no operator intervention. The agricultural environment being targeted consists of an irregular terrain supporting crops or sparsely populated with objects. This results in complex problems of identification, monitoring and control.

In this paper we introduce a fuzzy hierarchical controller for autonomous mobile robot for use in such agricultural environments. This controller utilises co-operating behaviours (obstacle avoidance, edge following, goal seeking) to navigate in tight spaces and navigate towards its target (bales of hay, boxes of fruit/vegetables). The proposed system simplifies fuzzy controller design and provides a control architecture with a fast response, robust performance, and the ability to deal with dynamic environments. This paper describes the preliminary development work in which the proposed controller architecture was tested on an in-door mobile robot. The next phase will concern the transfer of this work to our outdoor electrical and diesel robots .

## 1. Introduction

### 1.1 Background

The need for increased farm productivity set against a declining workforce is driving work on developing fully automated farm machinery, including the development of unmanned agricultural vehicles [Callaghan 97]. In a factory such automation is relatively simple but in an agricultural setting the inconsistency of the terrain, the irregularity of the product and the open nature of the working environment result in complex problems of identification and sensing and control. Problems can range from the effects of varying weather conditions on vehicle sensors and traction performance, through to the need to deal with the presence of unauthorised people and animals. All these problems provide good opportunities for fuzzy systems as they excel in dealing with imprecise and varying conditions which characterises such situations. Producing fully autonomous farm vehicles is a difficult objective although there have been notable steps in this direction, such as the Aurora greenhouse robot [Mandow96], but the application and environment variation in the greenhouse is restricted with respect to the outdoor situations.

Since Lotfi A.Zadeh introduced the subject of fuzzy sets in 1965[Zadeh65], Fuzzy logic has become an increasing popular approach to reactive control. The success of fuzzy control is owed in large part to the technology's ability to convert qualitative linguistic descriptions into non-linear mathematical functions. Fuzzy rules provide an attractive means for mapping sensor data to appropriate control actions. The agricultural robot must react to dynamic events in unstructured agricultural environments, using multiple sensors such as vision, sonar, mechanical wands, GPS sensors, infrared sensors. Reactive sensors involve mapping sensor inputs to control signals quickly, usually involving little or no intermediate representation. For mobile robots, the size of the input space requires a complicated control function. This mapping can be made more manageable by breaking down the space for analysis by multiple agents, each of which responds to specific types of situations, and then integrating the recommendations of these agents. Agents also called behaviours, pioneered by Brooks, and composed of many simple co-operating units, have produced very promising results when applied to the control of mobile robots [Brooks 1986]. The behaviours can be designed independently to exhibit behaviours such as goal-seeking, obstacle avoidance, and wall following.

The problem of how to co-ordinate the simultaneous activity of several independent behaviour-producing units to obtain an overall coherent behaviour have been discussed by many authors. One of the proposals in the literature is to use an on-off switching schema [Brooks86]. In this for each situation, one behaviour set is selected and is given complete control of the effectors. This scheme has produced very encouraging results in indoor mobile environments but reportedly performs less well where several more diverse criteria need to be taken into account. Other proposals relied on dynamic arbitration policies, where the decision of which behaviour to activate depends on both the current sub-goal given by the planner and the environmental situation [Payton1986] , these proposals do not allow for the concurrent execution of behaviours.

Both fixed and dynamic arbitration policies can be implemented using the mechanisms of fuzzy logic, as it gives the ability to express partial and concurrent activation of behaviours and it allows the smooth transition between behaviours. This fuzzy arbitration have been applied by ourselves [Voudoris95] and by many researchers such as [Saffiotti97] but in these implementations, either the fuzzy arbitration was between one or two behaviours typically demonstrated navigating maze-like environments [Voudoris95] or executing some sort of pre-planned activity which in our case is not practical as we navigate in a dynamic and changing environment.

In this paper we present a *Fuzzy Hierarchical System.* The example we shall present utilises four basic fuzzy behaviours which are left/right edge following, goal seeking, obstacle avoidance. We also use the concept of fuzzy context rules developed by [Saffiotti97] in order to have smooth transition between the behaviours in a reactive way such that we do not need any plan. The proposed system provides a *simpler design* procedure than other fuzzy approaches (i.e. less rules) and achieves *smoother behaviour* transitions than other behaviour based architectures. The system was tested on an indoor mobile robot navigating an environment consisting of irregular geometrical objects and layout. The robot used low cost and imprecise sonar and infrared sensors. We found the robot successfully escaped from mazes, avoided obstacles, and reached its goals. The aim of the project is that the same real time architecture is be copied to our outdoor robots (a large diesel powered agricultural vehicle) with newly developed sensors to deal with agricultural domain. In the following sections we explain the system and the robots in more detail.

## 2. The Hierarchical Fuzzy Logic Controller (HFLC) Design

### 2.1 The Fuzzy Logic Controller (FLC) Fundamentals.

Lotfi A.Zadeh introduced the subject of fuzzy sets in 1965[Zadeh65]. In that work Zadeh proposed that one of the reasons humans are better at control than conventional controllers is that they are able to make effective decisions on the basis of imprecise linguistic information, Hence it should be possible to improve the performance of electromechanical controllers by modelling the way in which humans reason with this type of information.

Figure(1) shows the basic configuration of an FLC , which consists of four principal components :

1. The fuzzification interface which:
   a) Measures the values of the input variables.
   b) Performs a scale mapping that transfers the range of values of input variables into corresponding universes of discourse.
   c) Perform the function of fuzzification that converts input data into suitable linguistic values, which may be viewed as labels of fuzzy sets.

2. The knowledge base comprises knowledge of the application domain and the attendant control goals. It consists of :
   a) The database which provides the necessary definitions which are used to define linguistic control rules and fuzzy data manipulation. on FLC.
   b) The rule base characterises the control goals and control policy of the domain experts by means of a set of linguistic control rules.

3. The decision making logic which is the kernel of an FLC , it has the capability of simulating human decision-making based on fuzzy concepts and of inferring fuzzy control actions employing fuzzy implication and the rules of inference in fuzzy logic.

4. The defuzzification interface which:
   a) Makes a scale mapping which converts the range of values of output variables into corresponding universes of discourse.
   b) Defuzzification , which yields a nonfuzzy control actions from an inferred fuzzy control action.

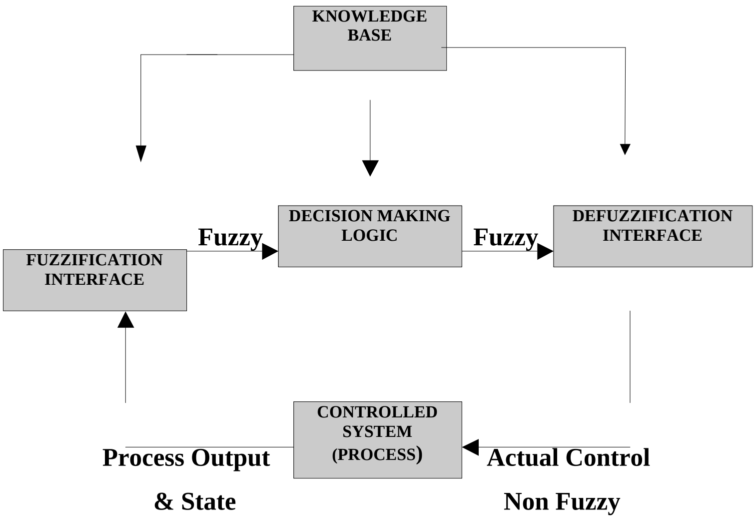



Figure (1): The basic configuration of an FLC.

In the following design of each single behaviour we will use singleton fuzzifier, triangular membership functions, product inference, max-product composition, height defuzzification. The selected techniques are chosen due to their computational simplicity.

The equation that maps the system input to output is given by

$$\frac{\sum_{p=1}^{M} y_p \prod_{i=1}^{G} \alpha_{Aip}}{\sum_{p=1}^{M} \prod_{i=1}^{G} \alpha_{Aip}} \qquad (1)$$

Where M is the total number of rules , y is the crisp output for each rule, $\alpha$Ai is the product of the membership functions of each rule inputs. More information about fuzzy logic can be found in [Lee90].

### 2.2 Design Complexity

The benchmark tasks targeted by this work are *collecting field based objects* (e.g. hay bales, fruit boxes) and *cutting crops*. In a previous paper we outlined the operation of a single level fuzzy controller that used a mechanical wand (a new sensor developed at Essex) to follow crop edges [Hani98]. In this paper we extend this work to a hierarchical fuzzy controller (consisting of numerous co-operating fuzzy processes) which aims at enabling farm vehicles to safely navigate to target objects such as hay bales. The initial design and experimentation has taken place within our laboratory but the controller has been designed in such a way as to allow the work to be transferred to an outdoors environment at a later stage.

Most commercial fuzzy control implementations feature a single layer of inferencing between two or three inputs and one or two outputs. For autonomous robots, however the numbers of inputs and outputs are usually large and the desired control behaviours are much more complex. For example in our case we have seven sonar inputs and an infrared bearing reading (i.e. eight inputs) and we have two outputs which are the left and right wheel speed. In this case using single layer of inferencing will lead to three fuzzy sets representing each input variable, and five fuzzy sets representing each output variable. In this case we have to define (3*7+5*2)=31 membership functions and $3^7$= 2178 rules which is difficult to determine if not impossible.

Whilst behaviour based architectures have produced better results than their traditional AI counterparts there is a problem of how to optimise behaviour co-ordination. Many proposals in the literature use on-off switching schema [Brooks 86] which typically operate by allowing one behaviour set to be selected and given complete control of the effectors for each situation. The adaptability of such architectures depends on both having a sufficiently rich set of behaviours present and the coarseness of the switching or co-ordination scheme. An alternative which seeks to improve upon the above switching scheme has been advocated by several researchers and is based on a co-ordination schemes that allow parallel execution of different behaviours [Ruspini91]. Our proposed architecture adopts this concurrent execution approach.

### 2.3 Proposed Hierarchical Fuzzy Logic Control (HFLC) Solution

The work described in this paper suggests a solution based on using fuzzy logic to both implement individual behaviour elements and necessary arbitration (allowing both fixed and dynamic arbitration policies to be implemented). We achieve this by implementing each behaviour as a fuzzy process and then using other fuzzy processess to coordinate them. The resultant architecture takes a hierarchical tree structure form and is shown in Figure 2.

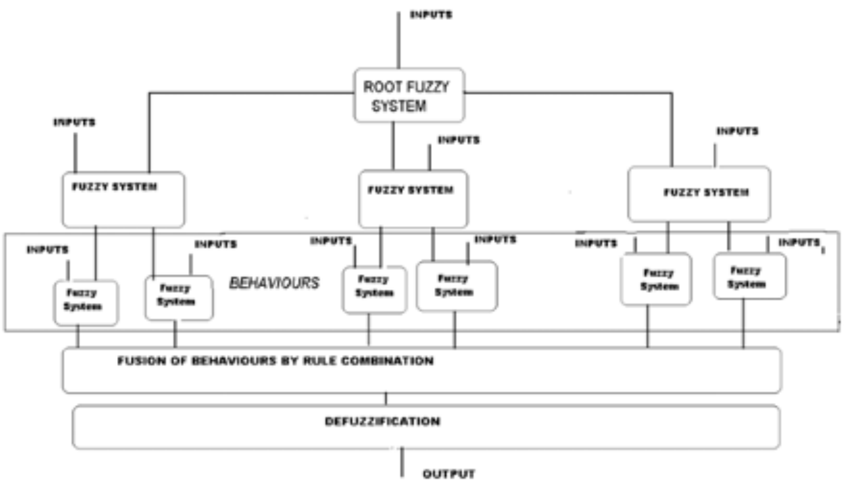



Figure (2): The fuzzy decision tree architecture.

### 2.4 Overview of the HFLC Architecture

[Ruspini91] defines fuzzy command fusion as interpretation of each behaviour producing unit as an agent expressing preferences as to which command to apply. Degrees of preferences are represented by a possibility distribution (or fuzzy as in our case) over the command space. In our HFLC architecture we use a fuzzy operator to combine the preferences of different behaviour into a collective preference. According to this view, command fusion is decomposed into two steps: preference combination and decision. In case of using fuzzy numbers for preferences, product-sum combination and height defuzzifcation. The final output equation is:

$$C= \frac{\sum_i (BW_i * C_i)}{\sum_i BW_i} \qquad (2)$$

Where i = right behaviour, left behaviour, obstacle avoidance,navigation. Ci is the behaviour command output (left and right velocity in our case). These vectors have to be fused in order to produce a single vector C to be applied to the mobile robot. Bwi is the behaviour weight. The behaviour weights are calculated dynamically taking into account the situation of the mobile robot. By doing this we do not need any pre-plan as the system plans for its self depending on the current situation of the environment. In figure(2) each behaviour is treated as an independent fuzzy controller and then using fuzzy behaviour combination we obtain a collective fuzzy output which is then deffuzzified to obtain a final crisp output. By implementing this as hierarchical set of fuzzy engines the following advantages are gained:

a) The ability to express partial and concurrent activation's of behaviours.
b) Smooth transitions between behaviours.
c) Much reduced number of rules (i.e. a much simplified design problem).

Such an approach results in a *large decrease in the number of rules* (reduced to 52 for the previous example) and enables *more flexible arbitration* policies to be achieved by using fuzzy meta-rules or context rules. These have the form IF context THEN behaviour [Saffiotti97] which means that a behaviour should be activated with a strength determined by the context (i.e. a formula in fuzzy logic). When more then one behaviour is activated , their outputs will have to be fused and each behaviour output will be scaled by the strength of its context. Fuzzy context rules have been initially applied by [Surgeo93] to switch between modes in a fuzzy-controlled unmanned helicopter and by Saffiotti [Saffiotti 97] in the mobile robot Flakey and by our previous work [Voudoris95].

In this work we aim to implement the system with no plan of the field so that the system is fully reactive to the environmental changes and able to achieve its goal as fast as possible while avoiding and obstacles or navigating pathways on its journey. The behaviours implemented in this system are the minimum set of behaviours we need to demonstrate this architecture working in our target “farm object fetch” application. This are described in greater detail later in this paper.

## 3. The Experimental Set up

Before we describe the experimental results in detail, it will be first necessary for us to set out the problem we are setting for our robots. In addition to this we will provide a brief description of the mobile robots themselves.

### 3.1 The Essex Robots

At Essex we have over 20 mobile robots ranging in type from micro-robots that move around on desk-tops to large out-door diesel robots that are used in our farm automation research. The simplified diagram of the computer system of our outdoor robots (a large diesel powered vehicle and a smaller electric robot) is shown in figure (3) and their photos are shown in figure (4),figure (5).

We use a distributed control system which provides many advantages. For example distribution enables the processing to be sized cost-effectively across a range of agricultural vehicle applications (e.g. processing power is added in proportion to sensor quantity and type). Properly structured distribution can also simplify and reduce the amount of interconnection that is required between sensors, actuators and controllers and thus reduce the wiring complexity. As the control system is now part of the device it is in theory possible to reduce the connection to a single ‘command’ connection that interconnects all devices. This is the concept behind the CANbus (Controller Area Network) developed for automotive industry and utilised in our robots. Also distribution makes it possible to build ‘damage containment’ into the distributed system that prevents that spread of damage from one part of the system to another. The current design is influenced largely by the requirements for both parallel and distributed processing in a real-time environment. The two main components are :

a) The VME Bus system

We have several Motorola MVME167 boards each with an MC68040 processor and up to 8MB of RAM for the use on the experimental vehicle. The processor boards communicate with each other and with the CAN bus via an HM CAN01 board and the VME bus. Each processor runs VxWorks as the real-time kernel running the target application code. Radio modems form a link via serial ports to a remote PC for teleoperation.

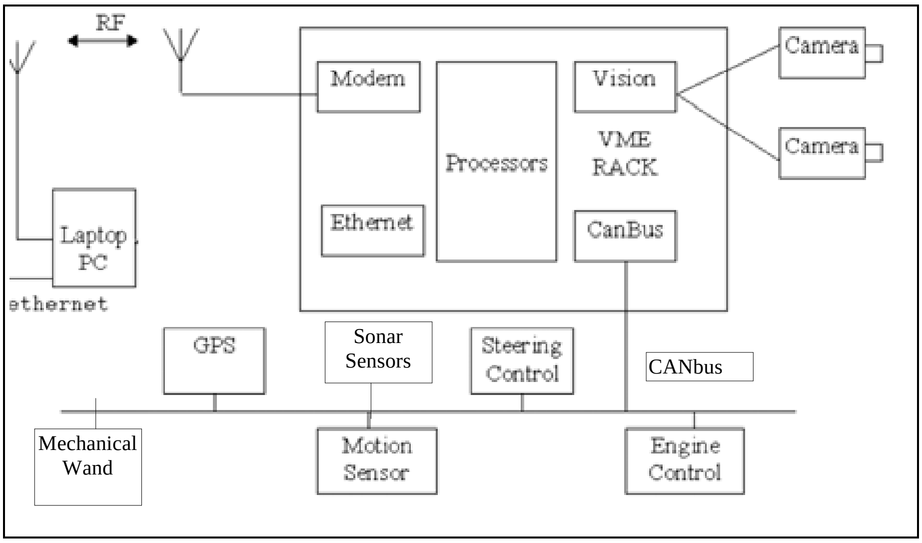



Figure (3) : Simplified Diagram of the computer system of the electrical and diesel vehicles.

b) CAN network

CAN (Controller Area Network) is a serial field bus protocol originating from the automotive industry. It was designed for use in a noisy environment and has numerous error-detection and correction facilities which has already made it popular in some farm vehicles. Being bus-based, nodes are simply connected to the bus, obviating the need for a traditional wiring harness. A bit-wise arbitration scheme ensures bus access priority for the highest priority message, which combined with an upper time bound on message latency, provides for a potential real-time response at any node. Each node in our system has local intelligence, and consists of sensors and/or actuators connected to an I/O port. The system can support multiple CAN buses.

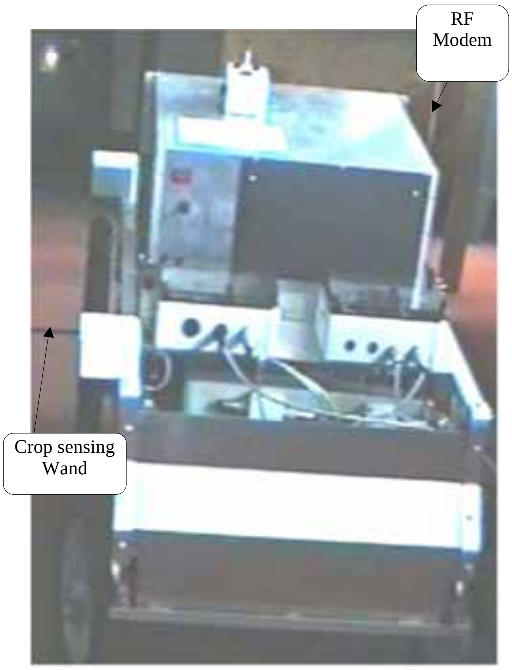



Figure (4): The outdoor electrical robot.


Figure (5): The outdoor diesel robot.

To devop programs to run on the VxWorks kernel we use Tornado which is a PC based cross-development system which we deployed across both PC hosts and Motorola targets. Programs developed on PC can be compiled and transparently downloaded, run and debugged on any target (i.e. robot) connected to it via a network, bus or other port (i.e. targets may be remotely located). We have found this particularly useful for field work where we use Tornado on a portable PC to debug and develop the vehicle control software. Code changes can be made by download from a laptop to the system using a wireless modem link.

The main outdoor robot we use is powered by a diesel engine. In addition to providing traction, the engine generates also electrical power which can be used to charge the batteries powering the computers. Both robots have mechanical wands (potentiometer arms to sense the edge of a crop), ultra-sound sensor, GPS, and a camera. The camera forms part of a system developed by our group [Schallter96] to locate hay bales. The first version of the fuzzy hierarchical controller described herein was implemented on a medium sized indoor robot (about 1ft cubed).

The indoor robot has a ring of 7 ultrasonic proximity detectors, an 8-axis vectored bump switch and an IR beacon sensor to aid navigation. We try to give all our robots a similar architecture (to simplify development work) so its hardware is also based on embedded Motorola processors (68030) running VxWorks RTOS. In the initial experiments infrared beacons were used to simulate the bales of hay in the field. The physical manifestation of the robot is shown in figure (6). In the outdoor robot the same controller will be used but the infra-red beacons will be replaced by the vision based bale detection developed by [Schallter97] shown in figure (7). The mechanical wand system will be used in conjunction with ultrasound sensors developed in Essex which have been designed to have a high level of noise immunity.

### 3.2 The Challenge

An aerial view of a field would show that it has many similarities to the floor of a laboratory. For instance field boundaries, entrances and objects (e.g. hay bales, fruit boxes, trees, crops, pylons, other vehicles….etc.) have similar forms to many laboratory objects (or at least can be made to have). Sometimes the field objects may be aligned to produce corridor or maze like geometry (e.g. rows of plants or trees). Of course there are also important differences such as the variation of ground, weather and lighting conditions that make the real-world a particularly different challenge for a robot. However, whilst it is self evident that ultimate test of a farm robot is on a real outdoors farm (and our outdoor electric and diesel vehicles are built for this purpose), we have concluded it is possible to undertake some useful and meaningful work within a laboratory by carefully selecting the tasks. For our initial work on the underlying principles we have used the following test set up, that we believe sufficiently mirrors the targeted tasks and thus provides a meaningful benchmark.

#### 3.2.2 The Hay-Bale Benchmark test

Figure (15),(16),(17) show the benchmark test we are using. In these tests we emulate:

- farm vehicles by indoor mobile robots
- farm objects by pillars
- boundaries of the field by artificial walls
- objects to be collected by IR beacons
- moving objects (eg people & animals) by human beings moving around the robot.

Ground and weather conditions were ignored in these control experiments.

## 4. The Proposed Architecture

### 4.1Berhaviour-based Decomposition

As explained earlier, the underlying principle is that the robot controller will be built from a set of fuzzy processes each providing some basic machine behaviour. Thus the first task is to define a set of basic behaviours that will allow the robot to complete the challenge. From other work [Brooks 86] we know that four behaviours are sufficient, namely goal seeking, obstacle avoidance, right edge-following and left edge-following.

#### 4.1.1 Obstacle avoidance:

In this behaviour , the three front sensors of the robot are used , which are the left front sensors (LF), the medium front sensors (MF), the right front sensors (RF), the configuration is shown in figure (6). We use here only three input membership function for each input as shown in figure (8), the rule base for this behaviour is shown in table(1). The output membership function is shown in figure(9) which is the same for all behaviours.

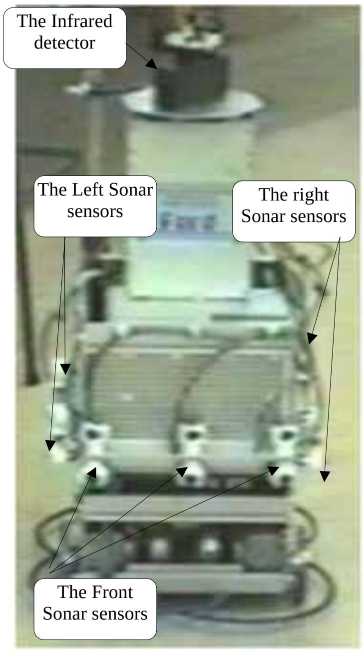
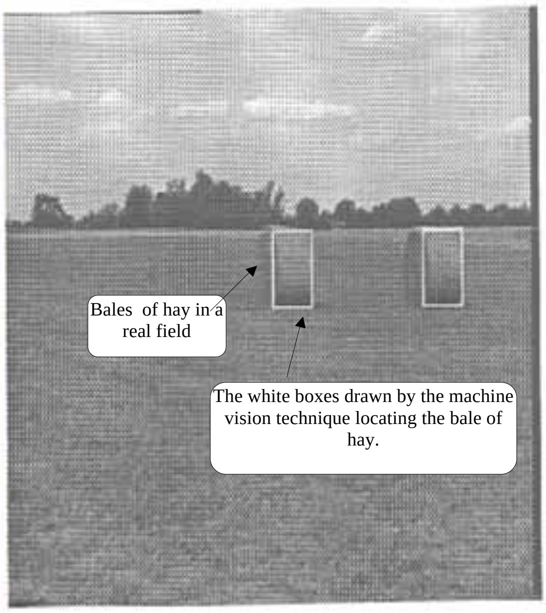



Figure (6): The indoor robot and its sensor configuration.




Figure(7) : The bales of hay detection using Machine vision technique.

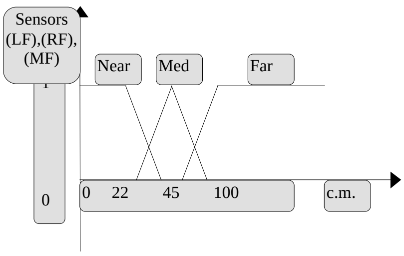



Figure (8): The membership function of the front sensors.

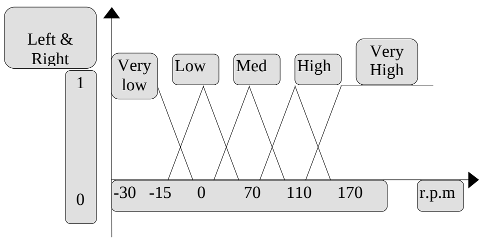



Figure (9): The output membership functions velocities of the robot.

### 4.1.2 Left and Right edge Following

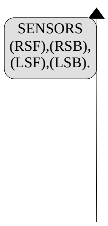
These two behaviours are used to follow a wall on the right or left side of the robot. The right edge following behaviour uses two right side sensors RSF (right side front) and RSB (right side back) . The left edge following behaviour uses two left side sensors LSF (left side front) and LSB (left side back).

The sensor configuration is shown in figure (5). We use only three membership functions for each input as shown in figure (10). The output membership function is shown in figure (11) and is the same for all the behaviours. The fuzzy rule base of the right edge following is shown in table (2) and its complement is the rule base for the left edge following.

| LF | MF | RF | LEFT VELOCITY | RIGHT VELOCITY |
|---|---|---|---|---|
| NEAR | NEAR | NEAR | LOW | LOW |
| NEAR | NEAR | MEDIUM | HIGH | VERY LOW |
| NEAR | NEAR | FAR | HIGH | VERY LOW |
| NEAR | MEDIUM | NEAR | MEDIUM | MEDIUM |
| NEAR | MEDIUM | MEDIUM | HIGH | VERY LOW |
| NEAR | MEDIUM | FAR | HIGH | VERY LOW |
| NEAR | FAR | NEAR | MEDIUM | MEDIUM |
| NEAR | FAR | MEDIUM | MEDIUM | LOW |
| NEAR | FAR | FAR | HIGH | LOW |
| MEDIUM | NEAR | NEAR | VERY LOW | HIGH |
| MEDIUM | NEAR | MEDIUM | VERY LOW | HIGH |
| MEDIUM | NEAR | FAR | HIGH | VERY LOW |
| MEDIUM | MEDIUM | NEAR | VERY LOW | HIGH |
| MEDIUM | MEDIUM | MEDIUM | MEDIUM | MEDIUM |
| MEDIUM | MEDIUM | FAR | HIGH | LOW |
| MEDIUM | FAR | NEAR | LOW | MEDIUM |
| MEDIUM | FAR | MEDIUM | MEDIUM | MEDIUM |
| MEDIUM | FAR | FAR | HIGH | MEDIUM |
| FAR | NEAR | NEAR | VERY LOW | HIGH |
| FAR | NEAR | MEDIUM | VERY LOW | HIGH |
| FAR | NEAR | FAR | VERY LOW | HIGH |
| FAR | MEDIUM | NEAR | VERY LOW | HIGH |
| FAR | MEDIUM | MEDIUM | LOW | HIGH |
| FAR | MEDIUM | FAR | NORM | HIGH |
| FAR | FAR | NEAR | LOW | HIGH |
| FAR | FAR | MEDIUM | MEDIUM | HIGH |
| FAR | FAR | FAR | VERY HIGH | VERY HIGH |

Table 1: The fuzzy rule base of the obstacle avoidance.

| RSF | RSB | LEFT VELOCITY | RIGHT VELOCITY |
|---|---|---|---|



| | | | |
|---|---|---|---|
| NEAR | NEAR | MEDIUM | MEDIUM |
| NEAR | MEDIUM | HIGH | VERY LOW |
| NEAR | FAR | HIGH | VERY LOW |
| MEDIUM | NEAR | VERY LOW | HIGH |
| MEDIUM | MEDIUM | MEDIUM | MEDIUM |
| MEDIUM | FAR | HIGH | VERY LOW |
| FAR | NEAR | VERY LOW | VERY HIGH |
| FAR | MEDIUM | VERY LOW | VERY HIGH |
| FAR | FAR | MEDIUM | MEDIUM |

Table 2: The Fuzzy rule base of the left edge following.

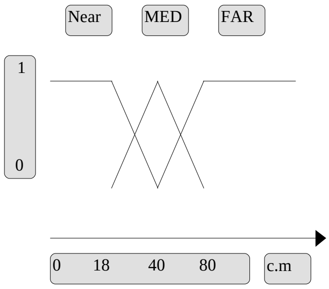



Figure (10):The membership functions of the side sensor.

### 4.1.3 The Goal Seeking

In this behaviour, we have one input from the infrared detector and the goals are in the form of infrared emitting beacons, the input is in the form of bearing of the robot from its goal, its membership function is denoted by goal and is shown in figure (11), the output membership function is the same as in figure (9), the fuzzy rule base is shown in table (3). Note that the imprecision of the infrared detector is high which helps simulate the problems in the open environment and give our controller a real challenge. In the outdoor robot, we will determine the bearing from our goal by using our machine vision technique for locating bales [Schallter97].

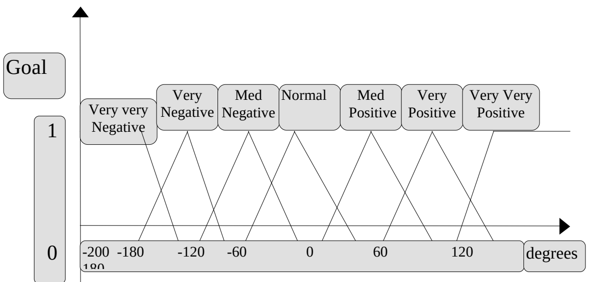



Figure (11): The membership function of the goal seeking.

| GOAL | LEFT VELOCITY | RIGHT VELOCITY |
|---|---|---|
| VERY VERY NEGATIVE | VERY LOW | VERY HIGH |
| VERY NEGATIVE | VERY LOW | VERY HIGH |
| MEDIUM NEGATIVE | LOW | MEDIUM |
| NORMAL | VERY HIGH | VERY HIGH |
| MEDIUM POSITIVE | MEDIUM | LOW |
| VERY POSITIVE | HIGH | VERY LOW |
| VERY VERY POSITIVE | VERY HIGH | VERY LOW |

Table 3: The fuzzy rule base of the goal seeking behaviour.

### 4.2 Behaviour Co-ordination.

In behaviour co-ordination there are some few parameters that must be calculated in the root fuzzy system. These parameters are the minimum distance of the front sensors which is represented by d1, the minimum distance of the left side sensors which is represented by d2 , the minimum distance of the right side sensors is represented by d3.

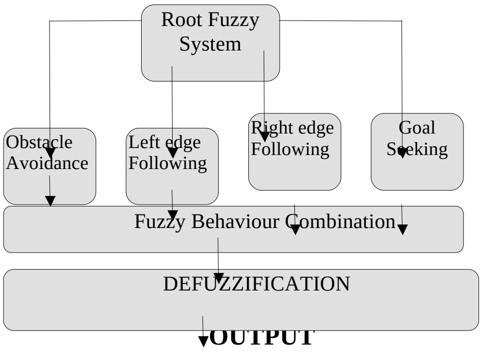
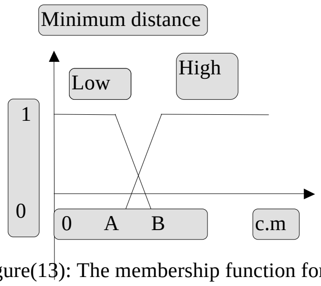



Figure (12): The behaviour co-ordinated system.




Figure(13): The membership function for minimum front sensors in case of d1 A=40 c.m , B=100 c.m. In case of d2,d3 A=18 c.m , B=36 c.m.

After calculating these values, each of them is matched to its membership function which are shown in figure(13) and these fuzzy values are used as inputs to the context rules which are :

*IF d1 IS LOW THEN OBSTACLE AVOIDANCE.*

*IF d2 IS LOW THEN LEFT WALL FOLLOWING*

*IF d3 IS LOW THEN RIGHT WALL FOLLOWING*

*IF d1 IS HIGH AND d2 IS HIGH AND d3 IS HIGH THEN GOAL SEEKING.*

These context rules determines which behaviour is fired and to what degree, then the final robot output is calculated using equation 2. The hierarchical architecture used in our system is shown in figure(12).

## 5 Performance Evaluation

The performance of the architecture has been assessed in two main ways. Firstly we computed the control surfaces for the behaviours that provide proof of stability, secondly we conducted practical experiments with the robots to track the robots paths and reactions to the benchmark tasks, as well as trying the robot in tight areas and mazes with irregular geometrical shapes and moving objects and trying it as well with real bales of hay which is a real challenge to the robot because of its irregularity and low sensitivity of sonar sensors toward them. The results of these tests are now presented.

### 5.1 Control Surfaces

The control surfaces visually present the unknown function articulated by the rules [Kosko92]. The linguistic rules have transformed to mapping from inputs to outputs which satisfy the stability requirement since small changes in the inputs correspond to small output changes (i.e. the surface gradient is not steep). Figure (14) shows the control surface of the right wall behaviour, the other individual behaviours have also given smooth and continuos response, and as the final output is a weighted sum of these continuos behaviours, then the system stability and smooth response is guaranteed because of the fuzzy transition between behaviours which a smooth transition.

### 5.2 Practical Experiments

The system was applied to various situations such as navigating in tight corridors, getting out of mazes, navigating towards certain goals (bales of hay or boxes of fruit in our case), and avoiding obstacles. In these tasks it showed a highly sensitive response, reacting to people and moving objects quickly, Although there are clearly big differences between the indoor environment and that of a farm we have done what we could to make the experiments more realistic For example we used noisy and imprecise ultra sound and infra red sensors together with irregular geometrical shapes. It was also corridors constructed from hay (in baled form).

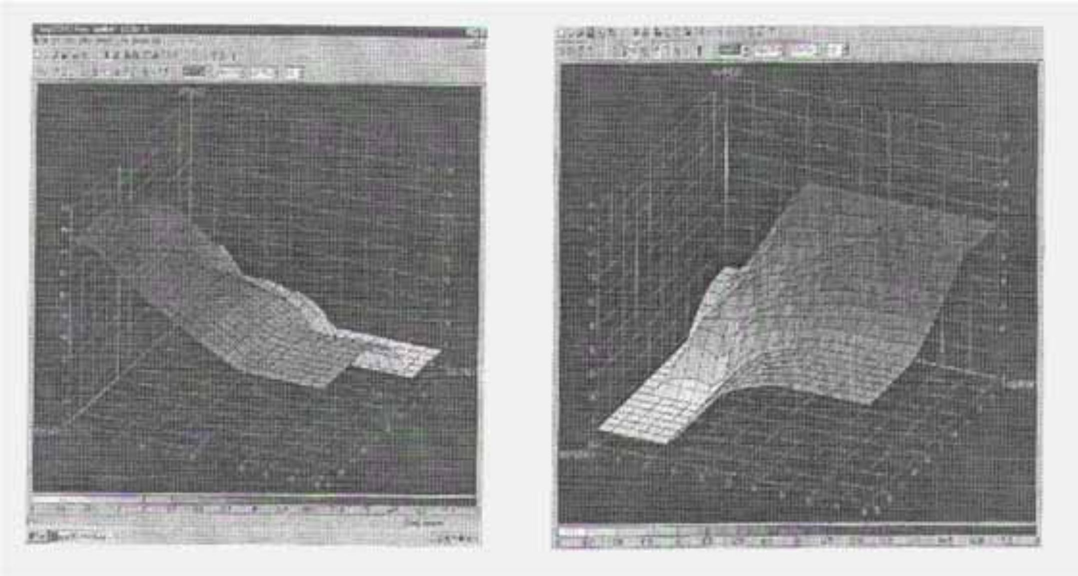


Figure (14) The control surfaces of the RSF and RSB against right and left velocities for the right behaviour.

We have tried numerous experiments but we will present here only a few illustrative cases. The robot path was drawn using a pen fixed in front of the robot to record the actual paths.

In figure (15) the robot starts in a tight corridor. The balanced behaviours of the right & left edge following plus the obstacle avoidance enable it to navigate successfully. It then enters a wider corridor, again navigating successfully and maintaining itself to the centre line. Note that it can deal with the different shaped and irregular areas which represent a real challenge to the ultrasound sensors (especially with the high noise and imprecise sensors we use). The experiments illustrate that our HFL controller deals with all these situations with a smooth and fast response. In the last stage of the journey the robot overcome another difficult challenge before successfully exiting the enclosure (note the smooth path).

In figure (16) the robot having left the enclosure, it steers towards goals(1), then it backs off simulating that it had collected the bale of hay then it begins its trip towards goal (2) then (3). It succeeds again with a very smooth control path shown in figure(16). As previously explained a similar procedure will be adopted for the outdoor robots but using our vision system rather than IR beacon.

Figure (17) represents a slightly more difficult situation in which the corridor is pinched making the corridor less regular. Again the balanced combination of behaviours enables it to navigate smoothly. Note its action on encountering an obstacle. It chooses to follow the larger path, because the obstacle avoidance behaviour will have the greater effect, after this the left edge following behaviour will dominate due to it encountering a wall on its left. The combined action allows it to get out from such maze situations.

Computer Science
**Comment:**

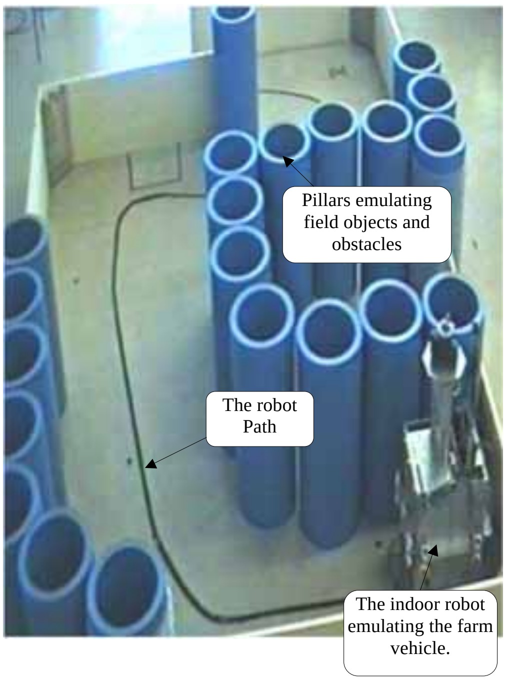



Figure (15): Experiment number 1.


Figure (16): Experiment number 2.

Note that if the left side was blocked and the exit was the other side, the robot will still follow the left wall until it finds its way out.

In figure(18) we repeated experiments 1,2 but with placing real bales of hay to investigate the sensitivity of the ultrasound sensors to them. The robot proved an excellent response to hay, navigating in tight corridors of bales of hay and avoiding them and reaching its goal safely.

In our experiments we have tried obstructing the path by getting people and moving objects to move in front of the robot to test its reactivity in dealing with the moving objects that the robot will encounter in the real field like animals, people, tractors ..etc. The system is very reactive and has shown it responds quickly to any moving object avoiding people which makes it suitable for static and dynamic environments.


Figure (17) : Experiment number 3.


Figure (18): Experiment number 4.

## 6. Conclusion

In this paper we have introduced a new fuzzy hierarchical controller for mobile robots for application in agriculture. Its main advantages are that it provides a *simpler design* procedure than other fuzzy approaches (i.e. less rules) and achieves *smoother behaviour transitions* than other behaviour based architectures. By way of an example of the former, the system was implemented using *only 52 rules* in contrast in the single stage fuzzy system which needed *2178 rules* to be of the same form as our 52 rule HFLC. In our experiments it navigated successfully in tight corridors, avoided obstacles and escaped from mazes, dealt with all the irregular shapes that were presented to it, even real bales of hay which are characterised by a highly irregular surface. The robot demonstrated a robust and fast performance being able to deal well with dynamic environments and moving objects. Our current work includes moving the architecture reported here to our outdoor robots and investigating the integration of on-line learning using genetic algorithms into the HFLC in order to make the system adaptive to the environment.

## Acknowledgements

We are pleased to acknowledge the support from Wind River Systems Inc. and HM Computing Ltd. We are especially grateful to other members of our group namely Adrian Simmomds, Libor Spacek, Malcom Lear, John Standeven, Paul Chernett, Tony Lawson, Robin Dowling. Thanks also to our partners at Agricultural University College Writtle namely Malcom Ragget and Ying Huodong for their advice and cooperation.

## References

[Brooks 86] Brooks,R.A. " A robust layered control system for a mobile robot " , IEEE Journal of Robotics and automation, RA-2(1):14-23,1986.

[Callaghan 97] Callaghan V, Chernett P, Colley M, Lawson A, Standeven J, "*Automating Agricultural Vehicles*", Industrial Robot Journal, Aug 1997

[Hani 98] Hani Hagras, Victor Callaghan, Martin Colley, Malcom Carr-West. "Developing a fuzzy logic Controlled agricultural vehicle", to appear in the Third International Conference on the application of Fuzzy Logic, Germany, 1998.

[Kosko 92] Kosko,B. " Fuzzy systems as universal approximators" ,Proceedings of the IEEE International conference on fuzzy systems , SanDiego , California, pp1153-1162,March 1992.

[Lee 90] Lee,C.C , " Fuzzy logic in control systems : Fuzzy logic controller -Part I, Part II", IEEE Trans , syst , Man , Cybern , Vol 20. No2, pp404-434, March 1990.

[Mandow96] Madow A et-al, "The Autonomous Mobile Robot AURORA for Greenhouse Operation", IEEE Robotics and Automation Magazine, Vol 3 No 4,1996 pp 18-28.

[Payton 86] Payton D.W. , "An architecture for reflexive autonomous vehicle control ", In Procs of the IEEE Int. Conf. On Robotics and Automation, pp 1838-1845, San Francisco, CA, 1986.

[Ruspini, 91] Ruspini,E.H., "Truth as utility: a conceptual synthesis. In procs of the 7th Conf. On uncertainity in Artificial Intelligence, Los Angeles, CA, 1991.

[Saffiotti 97] Saffiotti,A., "The uses of fuzzy logic for autonomous robot navigation: a catalogue raisomme", Technical report , IRIDIA ,Universite Libre de Bruxelles, Brussels, Beligum, 1997.

[Schallter 96] Schallter,S. ,"Object recognition in agricultural enviornment", M.Sc thesis, Essex University 1996.

[Sugeno 93] Sugeno, M. , "A fuzzy hierarchical control of unmanned helicopter. In Procs of Int. Fuzzy system Association Conference (IFSA), pp 179-182, seoul Korea, 1993.

[Surmann 95] Surmann, H. , Huser,J. , Peters, L., " A fuzzy system for Indoor Mobile Robot navigation", Fuzz IEEE , pp83-88, 1995.

[Voudoris 95] Voudoris.C. , Chernett,P., Wang, C.J.Wang, Callaghan,V. "Hierarchical behavioural control for autonomous vehicles", In Procs . of the 2nd IFAC Int. Conf. On Intelligent Autonomous Vehicles IAV-95,
pp 267-272, Helsinki University of Technology, Espoo, Finland, Pergamon Press, June 1995.

[Zadeh 65] Zadeh ,L.A., "Fuzzy sets ", Informat. Confr , vol 8 .pp338-353, 1965.