A Fuzzy-Genetic Based Embedded-Agent Approach to Learning & Control in Agricultural Autonomous Vehicles

Hani Hagras, Victor Callaghan, Martin Colley
The Computer Science Department,
Essex University, Wivenhoe Park, Colchester,
England (U.K).
email: robots@essex.ac.uk

Malcolm Carr-West
The Agricultural Engineering Department
University College Writtle,
England (U.K).

Abstract

This paper describes the design of a fuzzy controlled autonomous robot, incorporating Genetic Algorithms (GA) based rule learning, for use in an outdoor agricultural environment for path and edge following processes which involves spraying insecticide, distributing fertilisers, ploughing, harvesting, etc. The robot has to navigate under different ground and weather conditions. This results in complex problems of identification, monitoring and control. This paper addresses the development of an online self-learning system based on modified version of the Fuzzy Classifier system (FCS). The proposed technique has resulted in rapid convergence suitable for learning individual behaviours online without the need for simulation. The controller was tested on both an in-door and out-door mobile robot operating with different types of sensors (including a novel wands), propulsion and steering. Experiments include operating the vehicle following irregular crop edges (full of gaps) under different weather and ground conditions within a tolerance of roughly 2 inches.

1. Introduction

A casual glance around our world reveals how dependent we are on vehicles and their drivers. As a society, much of our resources are associated with driving vehicles. A long cherished dream has been driver-less cars, in which we are transported to our destination by an unseen “electronic chauffeur” whilst we indulge in more productive activities. The aircraft and boat industry already routinely use auto-pilots as a means of automatic guidance One of the most difficult technical challenges vehicle guidance is presented by the agricultural industry due to the inconsistency of the terrain, the irregularity of the product and the open nature of the working environment. These situations result in complex problems of identification, dealing with sensing errors and control. Problems include dealing with the consequences of the robotic tractor being deeply embedded into a dynamic and partly non-deterministic physical world (e.g. wheel-slip, imprecise sensing and other effects of varying weather and ground conditions on sensors and actuators). One of the most important tasks in a field are those based on crops planted in rows or other geometric patterns that involve making a vehicle drive in straight lines, turn at row ends and activate machinery at the start and finish of each run. Examples of this are in spraying, ploughing and harvesting. Our work addresses this challenge. We utilise a much-developed form of fuzzy logic augmented by GA learning that excels in dealing with such imprecise sensors and varying conditions, which characterises these applications.

2. Background

AI techniques including expert systems and machine vision have been successfully applied in agriculture. Recently, artificial neural network and fuzzy theory have been utilised for intelligent automation of farm machinery and facilities along with improvement of various sensors. Ziteraya and Yamahaso [11] showed the pattern recognition of farm products by linguistic description with fuzzy theory was possible. Zhang et al [12] developed a fuzzy control system that could control corn drying. Ollis [7] used machine vision to follow and cut an edge of a hay crop but however he did not address the problem of turning around corners and detection of the end of a crop row. Cho [1] used a simulation of a fuzzy unmanned combine harvester operation but adopted only on-off touch sensors for his fuzzy systems. Thus, he lost the advantage of fuzzy systems in dealing with continuos data which had led him not obtaining a smooth response and presenting problems when turning around corners. Also all of his work was simulated which is different from the real world farm environment. Yamasita[9] tested the practical use of an unmanned vehicle for green house with fuzzy control. Mandow[5] had developed the greenhouse robot Aurora, but the application and environment variation in the greenhouse is restricted with respect to the outdoor situations.

Little work has been done in implementing a real robot vehicle using fuzzy logic that can operate in open out door agricultural situations. Broadly speaking, our work situates itself in the recent line of research that concentrates on the realisation of artificial agents strongly coupled with the physical world [2]. A first fundamental requirement is that agents must be grounded in that they must be able to carry on their activity in the real world in real time. Another important point is that adaptive behaviour cannot be considered as a product of an agent considered in isolation from the world, but can only emerge from strong coupling of the agent and its environment [2]. Despite most robotics regularly use simulations to test their models, the validity of computer simulations to build autonomous robots is criticised and is subject to much debate [6].

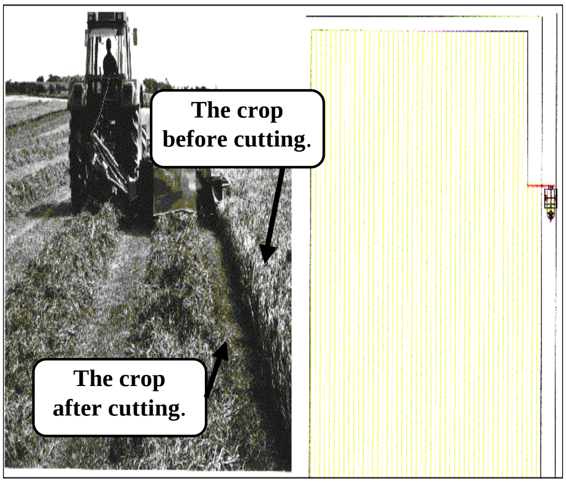
3. Overview of Paper

The aim of the work described in this paper is to develop a fuzzy vehicle controller for real farm crop harvesting. In earlier work [3] we developed a hierarchical fuzzy logic controller which had many advantages including reducing the number of rules needed and facilitating better behaviour arbitration. In this paper we describe how we have added Gas to provide rule learning where reinforcement can be given as actions are performed. A modified version of the Fuzzy Classifier system (FCS) is used in this algorithm. The FCS is equipped with a rule-cache making it possible for learnt expertise to be applied to future situations and to allow GA learning to start the search from the best point found. The system uses sensory information in-order to narrow the search space for the GA. This process can be viewed as a hierarchy. The proposed techniques have resulted in rapid convergence suitable for learning individual behaviours online without need for simulation. The focus of this paper is on the GA learning aspects of the controller.

4. The Target Environment

The robot is designed to harvest a crop by following its edge while maintaining a safe distance, in this case 45 cm from the vehicle, while at the same time allowing the cutter, which is fixed to the side of the vehicle, to cut the crop. Figure (1a) shows a hay harvester with the associated cutting technique being depicted in Figure (1b). The robot can also follow the crop edge or lines for other purposes like spraying insecticide, distributing fertilisers, ploughing, harvesting, etc.

Initially we have tested our design with an indoor mobile robot, introducing to it all the hard conditions that it might encounter in a real field. 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 such as using noisy and imprecise sensors, irregular geometrical shapes and fences constructed from hay (in baled form). However, it is self evident that ultimate test of a farm robot is on a real outdoors farm and we thus included as a subsequent stage an assessment stage based on the use of our outdoor electric and diesel vehicles. We feel that this approach is better than a computer simulation which suffers from well known modelling difficulties (especially when trying to model the physical environment comprising varying ground and weather conditions and objects such as trees and hay).



(a) (b)

Figure 1: a) A real world manned harvester to cut hay, b) The harvesting technique.

5. The Fuzzy Logic Controller

Zadeh [10] suggested 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. In the following analysis we will use a singleton fuzzifier, triangular membership functions, product inference, max-product composition, height defuzzification. The selected techniques are selected 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, α_{Aip} is the product of the membership functions of each rule inputs, G is the total number of inputs. The input Membership Functions (MF) shown in Figure (2) are the front and back side distance sensors (sensed by sonar or wands). The output MF are the wheel speeds (in case of the indoor robots)) and the robot speed and steering angle (in case of the outdoor robot). More details about the MF and the fuzzy controller can be found in [3]. The MF were designed according to the human experience. The rule bases were learnt using the proposed online algorithm described below.

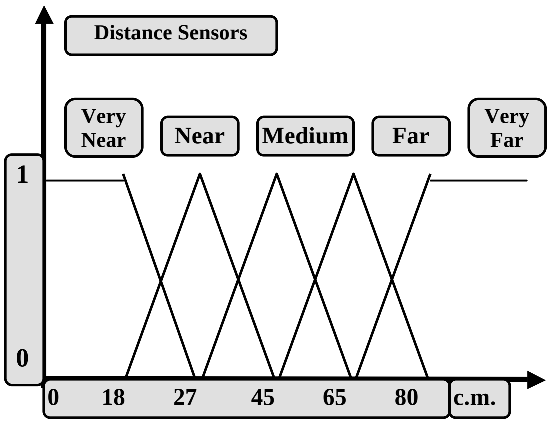


Figure 2: The MF of the input sensors.

6. The Online Algorithm description

In a real-time GA, it is desirable to achieve a high level of online performance while, at the same time being capable of reacting rapidly to process changes requiring new actions. Hence it is not necessary to achieve a total convergence of the population to a single string, but rather to maintain a limited amount of exploration and diversity in the population. Incidentally, it can be observed that near-convergence can be achieved in terms of fitness, with diverse structures [4]. These requirements mean that the population size should be kept sufficiently small, so that progression towards near-convergence can be achieved within a relatively short time. Similarly the genetic operators should be used in a way that achieves high-fitness individuals in the population rapidly [4]. Figure (3) introduces a block diagram of the operation of the proposed on-line algorithm. The rule base of the behaviour to be learnt is initialised randomly. In the following sections we will introduce the various steps of the algorithm.

6.1 Identifying Poor Rules

After the rule base initialisation, the robot starts moving. If it contains poor rules then it will begin deviating from its objective (e.g. not maintaining a constant distance from an edge). In this case an on-line algorithm is fired to generate new set of rules to correct this deviation. The GA population consists of all the rules contributing in an action (which is usually a small number as the rules base for each behaviour consists only of 25 rules). As in classifier systems, in order to preserve the system performance the GA is allowed to replace a subset of the classifiers (the rules in our case). The worst m classifiers are replaced by m new classifiers created by the application of the GA on the population [3]. The new rules are tested by the combined action of the performance and apportionment of credit algorithms. In our case, only two rules actions will be replaced (those already identifies with being predominantly responsible for the error).

6.2 Fitness Determination and Credit Assignment

The system fitness is evaluated by how much it reduces the absolute deviation (d) from the nominal value, which is given by:

$$d = \frac{|nominal.value - deviated.value|}{\max.deviation} \qquad (2)$$

Where the nominal value will correspond to the value that gives maximum normal membership function (45 c.m in case of wall following and zero degrees in case of goal seeking). The deviated value is any value deviating from the nominal value. The maximum deviation correspond to the maximum deviation that can occur (which is equal to 80-45= 35 c.m). So the fitness of the solution is given by d1-d2 where d2 is the absolute deviation before introducing a new solution and d1 is the absolute deviation following the new solution. The deviation is measured using the robot's physical sensors (the sonar in case of the wall following), which gives the robot the ability to adapt to the imprecision and the noise found in the real sensors rather than relying on estimates from previous simulations.

The fitness of each rule at a given situation is calculated as follows. As we have two output variables (left and the right wheel speeds or steering and speed), then we have Y_{t1} and Y_{t2}. Then the contribution of each rule p output (Y_{p1}, Y_{p2}) to the total output Y_{t1} and Y_{t2} is denoted by S_{r1}, S_{r2} where S_{r1} and S_{r2} is given by:

$$\mathbf{S_{r1}} = \frac{Y_{t1} - \dfrac{Y_{p1}\prod_{i=1}^{G}\alpha_{Aip}}{\prod_{i=1}^{G}\alpha_{Aip}}}{Y_{t1}},$$

$$\mathbf{S_{r2}} = \frac{Y_{t2} - \dfrac{Y_{p2}\prod_{i=1}^{G}\alpha_{Aip}}{\prod_{i=1}^{G}\alpha_{Aip}}}{Y_{t2}} \qquad (3)$$

We then calculate each rule's contribution to the final action S_c by $S_c = \frac{S_{r1} + S_{r2}}{2}$. Then the most two effective rules are those that have the two greatest value of S_c, we use mutation only to generate new solutions because of the small population formed by the fired rules.

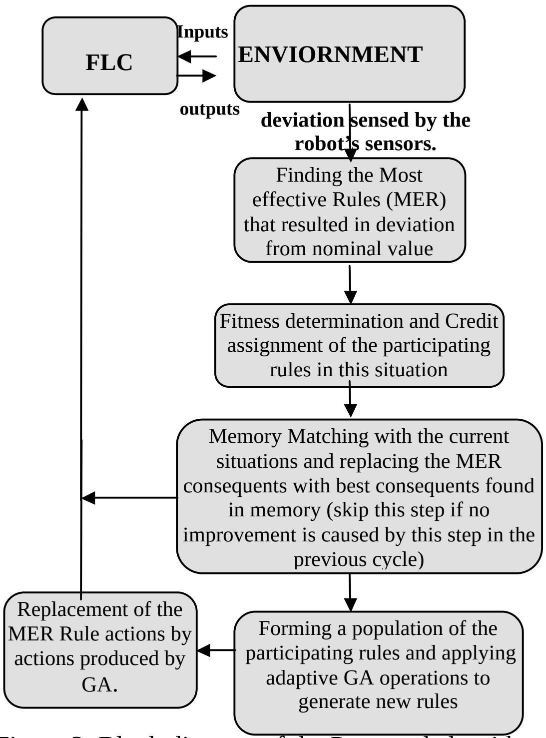


Figure 3: Block diagram of the Proposed algorithm.

6.3 Memory Application

After determining the rules actions to be replaced, the robot then matches the current rules to sets of rules stored in a memory containing each rule and its best fitness value up to date. The fitness of the rule in a given solution is given by**:**

$$\mathbf{S_{rt}} = \text{Constant} + (d_1 - d_2)\, S_c \qquad (4)$$

d_1-d_2 is the absolute deviation improvement or degradation caused by the adjusted rule base produced by the algorithm. If there is improvement in the deviation, then the rules that have contributed most will be given more fitness to boost their actions. If there is degradation then the rules that contributed more must be punished by reducing their fitness w.r.t to other rules, repeating the process for the next most responsible rule. For every rule action to be replaced the best fitness rule will replace the current action in the behaviour rule base. If the deviation decreases, then the robots will keep the best rules in the behaviour rule base. If the deviation still the same or it increases the robot fires the GA to produce new solutions by mutating these best rules until the deviation begins decreasing or the rule is proved ineffective when the robot is moving thus indicating another rule might be more effective. This action is supposed to speed up the GA search as it starts the GA from the best found point in the solution space instead of starting from a random point. This is then considered a solution for the current situation and the rule fitness is calculated and is compared with the maximum fitness rule. If its fitness is greater than the best kept one then it replaces the best one, otherwise the best one still is kept in the memory.

6.4 Using GA to Produce New Solutions

The GA begins its search for new rule actions to replace those identified with poor performance. Mutating the two most effective rules generates new solutions. A mutation rate of 0.5 was chosen after experimenting of different mutation rates from 0 to 1.0 and monitoring the time the robot needs to achieve its purpose (e.g. reaching its goal or following a wall). It was noticed that at mutation values less than 0.3 there is nearly no convergence as the population size and the chromosome size is small, and the low mutation rates does not introduce a lot of new genetic materials to introduce new solutions. The same occurs for high mutation rates (higher than 0.7) as the mutation rate reaches 1.0 the genetic materials available are the primary chromosomes (e.g. 0101) and its inversion (1010) which is not enough for introducing new solutions. So 0.5 gave the optimum value of finding a solution after, on average, 96 seconds. The robot also uses its sensory information to narrow up the search space of the GA and thus reducing the learning time. For example if the robot is implementing left wall following and it is moving toward the wall, then any action that suggests going to the left will be a bad action, thus if we use the front left side sensor and it senses that we are going toward the wall, then the GA solutions will have a constraint not to go left.

6.5 The Learning Length Criteria

The robot assumes it had learnt the required behaviour if it succeeds in maintaining the nominal value for the behaviour for a distance enough to proof that the learnt rule base is sufficient. The optimal learning distance has been related to units of length of the robot, so that the algorithm can be applied in an invariant manner to different size robots. In order to determine the optimal learning distance we have conducted numerous experiments evaluating performance relative to the robot's length (e.g. 1x robot's length, 2x robot's length, etc.). We then followed the same track that was used during learning to determine the absolute deviation at each control cycle from the optimum value (which would be maintaining a constant distance from a wall in case of edge following). Then we calculated the average and standard deviation of this error and compared different sizes for the learning length criteria (i.e. as short as possible whilst producing a stable rule base). It was found that the average and standard error for the wall following stabilises at three times the robot's length at average value of 2 c.m and standard deviation of 1 c.m.. Thus we use three times the robot the length as our learning length criteria.

7. Experimental Results

The performance of the architecture has been assessed in two main ways. Firstly, we physically emulated (rather than simulating) the crop following process. In this emulation we have conducted practical experiments with the indoor robots to track the robots paths and reactions to the irregular geometrical shapes forming fences (which fake the crop edge) including real bales of hay (forming a fence). These offer a real challenge to the robot because of their irregularity and low sensitivity of sonar sensors toward them. In the next phase we have tested the same architecture in outdoor environments tracking fences and crop edges in real farms. Each experiment was repeated 5 times recording the path to test the system repeatability and stability for different weather and ground conditions (eg rain, wind , holes in the ground, going up and down hill etc).

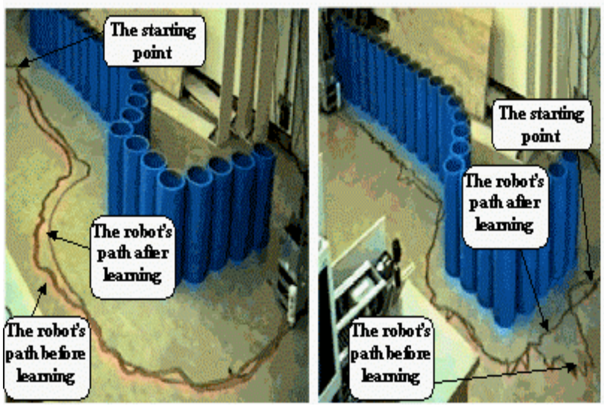
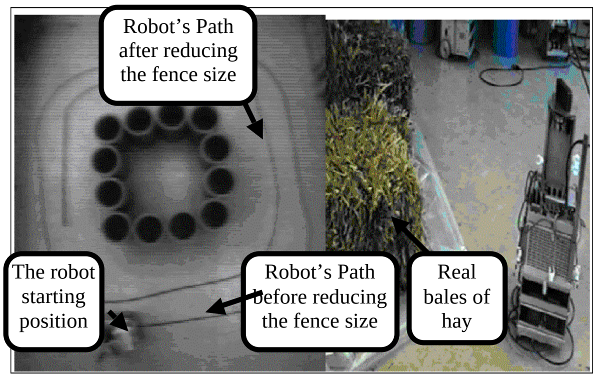


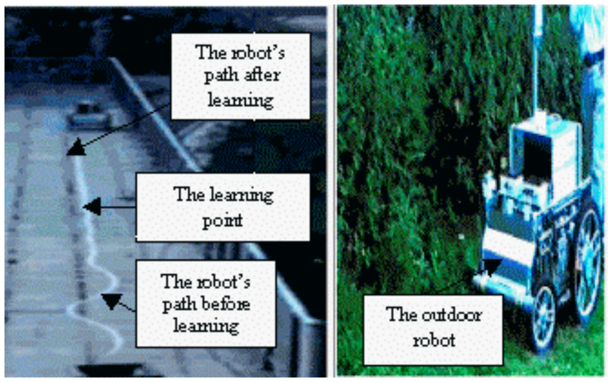
Figure 4: a) Learning left wall following. b) Learning right wall following.

Figure (4-a) shows the robot learning left wall following whilst dealing with an irregular edge and imprecise ultra sound sensing. The robot succeeded in learning the desired behaviour in an average of 96 seconds. In Figure (4-b) the robot learnt right wall following in 96 seconds. Note the robot learnt to follow the wall in approximately a straight line with minimum deviation. All of the learnt behaviours were tested in different (and difficult) terrain from those in which they were originally trained. In these tests the robots produced an average deviation of 2 cm and standard deviation of 1cm. This is very encouraging given the irregularity of the terrain, and the imprecision of the sensors. Figure (5-a) shows the robot emulating the crop cutting operation. Here it continues going inwards to complete the harvesting operations. The cutting action was simulated by reducing the size of the fence. Note that the response is smooth especially when the robot turns. This is due to the smooth transition between rules and the smooth interpolation between different actions that are characteristics of fuzzy logic. The same experiment was repeated but with real bales of hay and gave a very smooth and a repeatable response as in Figure (5-b).



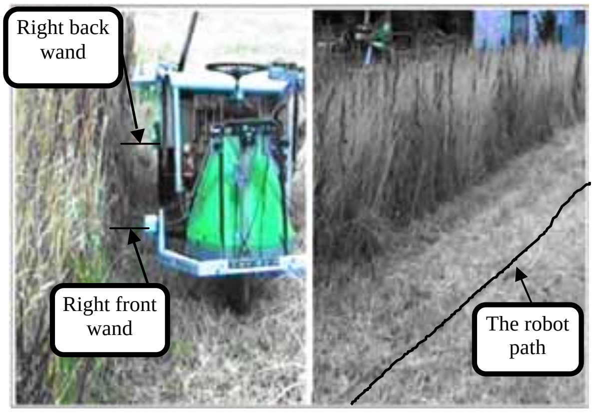
(a) (b)

Figure 5: a) The robot emulating the harvesting operation. b) The robot following fences of bales of hay.



(a) (b)

Figure 6: a) The outdoor robot learning to follow an irregular fence in outdoor environment. b) The electrical robot following out door irregular tree hedges.



(a) (b)

Figure 7: a) Diesel robot following irregular hay crop edge using mechanical wands. b) The robot path.

To show that the proposed system can deal with open outdoor environment we used another different outdoor robot (with 68040 microprocessor and 0.5 m/s maximum speed) to learn the right edge following behaviour of an irregular metallic fence. Figure (6-a) shows the robots learning this wall behaviour. The figure shows the robot reducing its deviation rapidly until it succeeds in maintaining the robot with almost zero deviation after a distance equal to three times the robot length. The robot's path after 79 seconds learning is very smooth (it uses faster processor), with only a very small deviation despite the irregularity of the fence and the highly imprecise sensors. Figure (6-b) shows the electric robot in a real farm following a plant edge characterised by high irregularity (eg gaps in edge, plants falling from the edge) and varying ground conditions (eg slopes and holes). It had used two ultra sound sensors to sense the crop edge. Once more the robot performed well following the crop at a safe distance from the edge. Although we currently have no quantitative means for evaluating the precision of the crop following, we estimate that the crop edge was tracked successfully within a tolerance of 2 inches.

In Figure (7-a) we tried the diesel robot in a hay field using the mechanical wand sensors following an irregular crop edge. The robot gave stable, repeatable and robust response as shown in Figure (7-b), and tracked the edge of the crop successfully within a tolerance of 2 inches. The robot also turned smoothly around the poorly defined hay crop corners.

8. Conclusions

In this paper we have developed a fuzzy controller for a robot aimed at automating crop following processes which includes spraying, ploughing and harvesting. We have developed a novel sensor design (outdoor mechanical wands) to be used in real farms under different conditions. We tested the fuzzy control architecture on an in-door mobile robot with only two ultrasound sensors. It had learnt to maintain itself at a constant distance from the emulated crop, in-spite of boundary irregularities and the imprecision in the ultrasound sensors. After testing the architecture successfully indoors, the control architecture was moved to the outdoor robots and environment in which the robot displayed a smooth and fast response and was able to track various edges under different environmental and ground conditions. The outdoor robots tracked irregular crop edges successfully within a tolerance of 2 inches. To the authors' knowledge, the work described in this paper is the only system which has successfully guided a diesel tractor in outdoor environments following real crop edges (including irregular edges which include gaps) and turning around corners with a high degree of repeatability and following the crop edge with a tolerance of two inches. The system learnt online, without the need for simulation, thereby producing robust behaviours that emerged as a result of interacting with the real dynamic world. We are currently investigating the performance of other farm tasks (such as collecting bales of hay or fruit boxes etc). In these we are going to use a fuzzy hierarchical controller to combine several behaviours for safe navigation toward our goals. In this work we will integrate a vision system hay-bale detection [8] and use it with our fuzzy system for reactive navigation.

References

[1] S. Cho, H. Ki , "Unmanned Combine Operation Using Fuzzy Control – Graphic Simulation", Applied Engineering in Agricultural , pp247-251, 1996.

[2] M.Dorigo, " Genetics-based machine learning and behaviour based robotics: A new synthesis", IEEE transactions on system, man and cybernetics, pp141-154, 1993.

[3] H.Hagras, V. Callaghan, M. Colley, M. West, " A behavior based hierarchical fuzzy control architecture for agricultural autonomous robots", The Int. Conference on computational Intelligence for modeling, control and automation, 1999.

[4] G. Linkens, O. Nyongeso, "GA for fuzzy control, Online system development and application", IEE Proc control theory applications, pp177-185, 1995.

[5] M. Mandow et-al, "The Autonomous Mobile Robot Aurora for Greenhouse Operation", IEEE Robotics and Automation Magazine, Vol. 3, pp18-28, 1996.

[6] O. Miglino , H. Lund, S. Nolfi., " Evolving Mobile Robots in Simulated and Real Environments. Artificial Life" Technical Report NSAL-95007, Reparto di Sistemi Neurali e Vita Artificiale, Istituto di Psicologia, Consiglio Nazionale delle Ricerche.

[7] M. Ollis , A. Stentz , " First results in vision-based crop line tracking", Proceedings of the 1996 IEEE International Conference on Robotics and Automation, pp951-956, 1996.

[8] S. Schallter, "Object Recognition In Agricultural Environment", MSc , Essex University, 1996.

[9] S. Yamasita, "Examination Of Induction Method Of Unmanned Vehicle For Greenhouse", in Proc.49th Congress in Soc. Of Farm Machinery pp273-274, Japan, 1990.

[10] Zadeh L A, "Fuzzy Sets", Informat. Conf, Vol. 8, pp. 338-353, 1965.

[11] Z. Ziteraya, K. Yamahoso, "Pattern Recognition Of Farm Products By Linguistic Description With Fuzzy Theory", in Proc .3rd Fuzzy system Symp of IFSA Brach, pp127-132, Japan, 1987.

[12] Q. Zhang, J. Letchfield , J. Bentsman, "Fuzzy Predictive Control Systems For Corn Quality Control Drying Food Processing Automation", in proceedings of the 1990 conference, pp313-320 , ASAE Pub.2, St Joseph, Mich.: ASAE. 1990.