Multiple Mobile Robots Controlled by Artificial Neural Networks

R C Seals
University of Greenwich
Dept of Engineering Science
Chatham Maritime
01634 883953
R.Seals@gre.ac.uk

T A Seals
Formerly of the University of Reading
Dept of Cybernetics
Whiteknights
Reading

ABSTRACT
Multiple small mobile robots have been created that were controlled by individual artificial neural networks. Each mobile robot was self-contained and capable of independent actions, as determined by the on-board artificial neural network. Information about the environment was collected from sensors mounted on each individual mobile robot chassis. Different sensors were available that were capable of providing information about different aspects of the environment. Currently there were sensors for detecting and following a black line as well as short range distance sensors for detecting and interacting with objects and other mobile robots. The artificial neural networks on the individual mobile robots were all provided with the same training data and a standard back-propagation training algorithm was used. However the randomised component of training the artificial neural networks did mean that there could have been subtle differences in the responses of individual mobile robots to the same sensor data. This effect was eliminated when needed by using an off-line training process and programming all the mobile robots with the same trained ANN. The small group of mobile robots was used to investigate two simple aspects of swarm behaviour; that of flocking and also of follow-my-leader, which are examples where the swarm appeared to operate with more intelligence than the individual members.

Keywords
Mobile robots, Artificial Neural Networks.

1. INTRODUCTION
Swarm robotics [1] uses a group of similar mobile robots to investigate the behaviour of several uncoordinated individual mobile robots when operating within the same environment. This study investigated a small group of similar mobile robots using a variety of simple environment sensors, controlled by a small artificial neural network.

1.1 Sensors Used
All the sensors used some variation on light. The simplest sensor was an optical reflectance module [2] that detected differences in surface colour, and were used to follow a black line on a white or light coloured background. The distance sensor [3] used tri-angulation to provide a short range distance measurement and these were used to detect the presence of other objects within the environment as needed for the follow-my-leader and flocking activities. These sensors were used individually or as arrays. When used as an array these experiments then three equidistantly spaced sensors were aligned in a straight line.

1.2 Artificial Neural Network
It has been reasonably easy to control a small mobile robot equipped with the simple sensors outlined when using a sequential C programme written to respond to the environment conditions, in order to achieve the mobile robot response desired. However to be effective this approach needed to consider all the possible scenarios the mobile robot woul encounter and then provide a specific response for each one. An alternative approach was to use biologically originated control methods such as an artificial neural network (ANN) [4]. The ANN was trained using representative examples of the situations that could be encountered. The benefit was that the ANN was able to fill in the ‘gaps’ between the training scenarios used and those encountered in real-life that may be considerably different. One of the drawbacks of ANNs was also that same characteristic of filling in the gaps as it was not possible to predict withmuch degree of certainty how the ANN would react in all situations. However, for the purposes of a small swarm of mobile robots operating within a closed and protected environment, this was not a significant problem.

2. THE MOBILE ROBOTS
The mobile robots used for this activity were designed to be low cost and simple in order to facilitate the manufacturer of a group or swarm of them. This resulted in a small chassis with corresponding small motor drive and sensor systems. The smaller size and relatively lower power requirements enabled primary battery cells to be used. This removed the need to incorporate battery charging technology or use more expensive secondary rechargeable battery cells. This did limit the operating life of the swarm but useful experiments were still performed within the timeframe achievable with the batteries.

2.1 DRIVE MECHANISM
In line with the requirement to keep the cost for individual mobile robots low, a small dual DC motor unit was used to provide the motive force for each mobile robot. The gear ratio of the gearbox was fixed at assembly and all the units were assembled with the same ratio. A H-bridge driver [5] was used to control the forward and reverse motion of the DC motors. This provided tank-style direction control. Although not implemented, it would be possible to control the speed of the DC motors as well using pulse width modulation with this H bridge driver.

2.2 SENSORS
The initial operation required for the mobile robot was to be able to detect and follow a black line and this was achieved using LED transmitter/receiver modules designed to detect objects at close distances, as illustrated in figure 1.

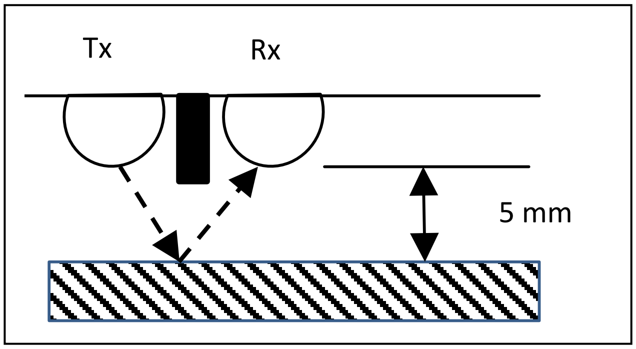

Figure 1 : Sensor Height arrangement

The black line to be detected and followed consisted of 18 mm width insulation tape as it could be shaped to a wide variety of corners, straight lines and other curves. Three of the LED Tx/Rx modules illustrated in figure 1 were then designed onto a PCB spaced 22mm apart, as illustrated in figure 2.

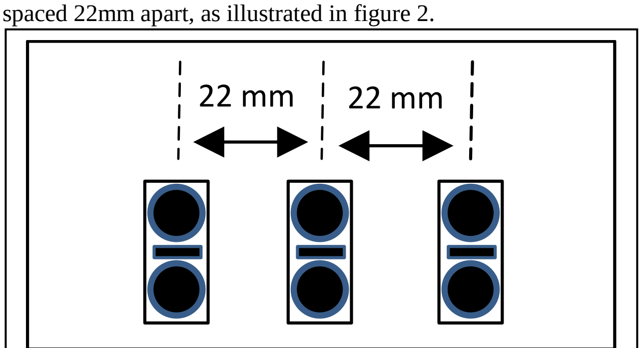

Figure 2 : Black Line Sensor Arrangement

This provided a 2 mm gap either side of the black tape which was generally sufficient as it was enough for the middle sensor to detect the black line and the two outer sensors to detect the background enabling the mobile robot to move forward in a straight line when the black line was straight. However, it was a sufficiently small gap that as soon as the black line started to move away from being directly forward, that the sensors detected the deviation quickly allowing the mobile robot to quickly track the change in direction and adjust the motor speeds accordingly.

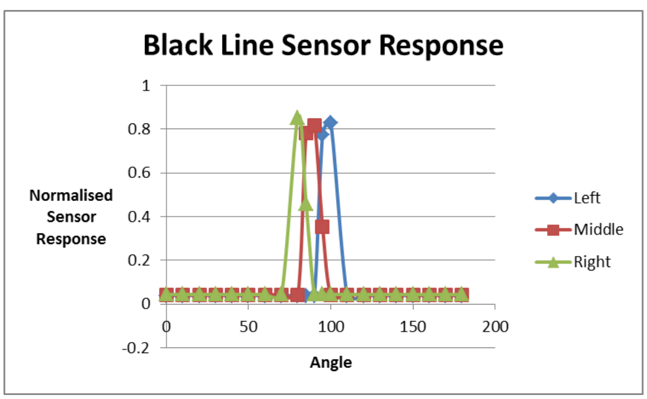

Figure 3 : Black Line Sensor System Response

Figure 3 illustrates the sensitivity of the black line sensor obtained by maintaining the mobile robot stationary over the black line and then rotating it 180 degrees, with 0 degrees being orthogonal to the black line. Rotation then took place clockwise with 90 degrees being perfectly aligned with the black line. The sensors provided a good discrimination and were able to detect mis-alignments of only 5 degrees. The drawback of this sensor was that if a mis-alignment of greater than 20 degrees occurred, for example at a sharp corner, then the sensors was not always able to detect the black line and did not perform the correct tracking motion. Figure 4 illustrates the difference in the black line sensor responses if the height of the sensors was raised from 5mm to 12 mm.

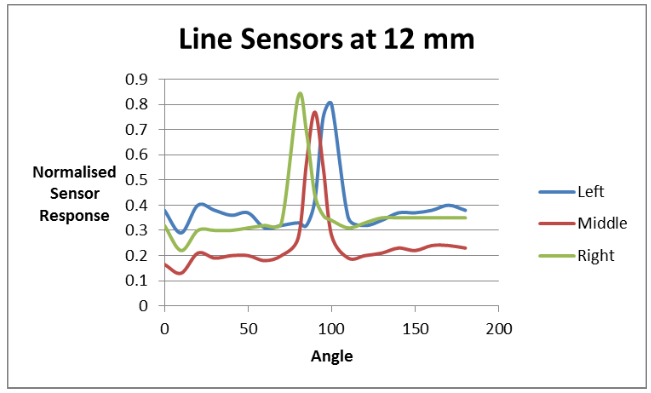

Figure 4 : Line Sensor Response at 12 mm Height

The white level threshold increased from 0.02 to 0.3 due to ambient light being able to be detected by the receiver LED. A background light source was located at 0 degrees leading to a white level of 0.4 which showed a slight reduction to 0.3 as the mobile robot was rotated from 10 degrees to 80 degrees. This was because the ambient light was increasing obscured by the mobile robot body. The overall result was a reduction in the discrimination capability between the background and the black line and confirmed the need to maintain the black line sensor at the optimum height of 5 mm.

As well as the black line sensors an array of three distances sensors were also mounted on the front of the mobile robot at a height of 3.5 mm as illustrated in figure 5.

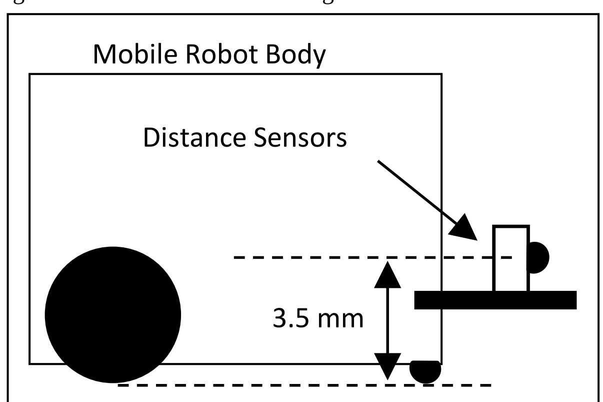

Figure 5 : Distance Sensors Fixed to Mobile Robot

The chassis of the mobile robot had a width of 50 mm and the three sensors mounted at the front spanned a width of 100mm in order to scan an area wider than the mobile robot.

3. CONTROL

The control microprocessor used was a customised Arduino UNO [6] compatible containing a dual H bridge driver, 6 digital/analogue inputs/outputs, a USB interface and a Bluetooth wireless communication link. The USB link was used to connect to the Arduino IDE for programming and the Bluetooth communication link for data gathering while the mobile robots were operating.

3.1 The Artificial Neural Network

The control of the mobile robot was implemented by using a simple feedforward fully connected artificial neural network as illustrated in figure 6. Training was performed using back-propagation.

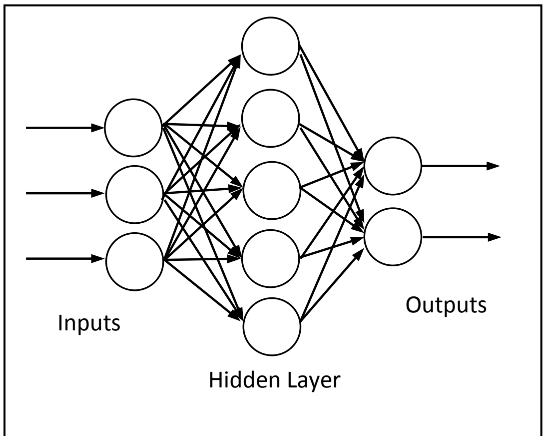

Figure 6 : Fully Connected Feedforward ANN

The design of the ANN was simple due to the limited processing capability of the microprocessor used. The number of input neurons was determined by the number of inputs. There was always a single hidden layer with 9 neurons and two output neurons, one for the left motor and one for the right motor.

3.2 Training

When training an ANN using back-propagation then a sequence of training vectors were required that specified the input values and the corresponding output values required.

```
const float Input[PatternCount][InputNodes] =
{                       // left  middle  right
  { 0.1, 0.1, 0.1 },  // white white white
  { 0.1, 0.1, 0.9 },  // white white black
  { 0.1, 0.9, 0.1 },  // white black white
  { 0.1, 0.9, 0.9 },  // white black black
  { 0.9, 0.1, 0.1 },  // black white white
  { 0.9, 0.9, 0.1 },  // black black white
  { 0.9, 0.9, 0.9 },  // black black black
};
```

Figure 7 : Input Training Values

```
const float Target[PatternCount][OutputNode]
= {             // Left  Right
  { 0.0, 0.0 },  //  off   off
  { 1.0, 0.0 },  //  on    off
  { 1.0, 1.0 },  //  on    on
  { 1.0, 0.0 },  //  on    off
  { 0.0, 1.0 },  //  off   on
  { 0.0, 1.0 },  //  off   on
  { 1.0, 1.0 }   //  on    on
};
```

Figure 8 : Output Training Values

The inputs and outputs could have any numerical values but with back-propagation training it was quicker if the values were all normalised to be between 0.0 and 1.0. Figure 7 illustrates the ideal input values from the three black line sensors used for the black line following control experiment and figure 8 illustrates the corresponding output values used to drive the motors.

3.3 Off-Line Learning

Two approaches were taken with training the artificial neural networks, on-line and off-line training. With on-line training the back-propagation training algorithm was implemented on each of the mobile robots each time the power was applied or the microprocessor was reset. During the development process this was not too much of a problem with the black line following function training taking approximately 1 minute and the obstacle avoidance function taking over three minutes.

However, in a longer term this delay lengthened the experimentation process as it introduced delays at the beginning of each test cycle as the individual mobile robots were stationary and unresponsive during training. A further issue with the online approach was that the trained ANNs could vary due to the use of randomness in the training process so that individual mobile robots could respond in subtly different ways to the same data. Additionally the same mobile robot with the same training data could result in different responses which were a feature of using ANNs.

By using an off-line training process the delay when restarting all the mobile robots at the beginning of each experiment can be avoided. Additionally, it would enable all the mobile robots being used to be programmed with exactly the same trained ANN rather than potentially different ones. This eliminated another variable in the experimental process. Off-line training was achieved by transferring the training part of the programme to a PC which when executed produced the numerical values needed for the trained artificial neural network. The numerical values were output in the form of a header file to be included into the rest of the programme used to implement the ANN. An example of the off-line header file produced is illustrated in figure 9.

```
#ifndef WEIGHTS_H
#define WEIGHTS_H
float OutputWeights[9][2] = {{-0.40751,-
0.48041},{-1.1036,-0.308797},{-0.341976,-
0.665995},{-0.53793,-0.273634},{-1.0652,-
1.08821},{-0.976137,-1.0877},{-0.256931,-
0.756558},{-0.200505,-1.0429},{-0.939134,-
1.2244}};
float HiddenWeights[3][9] = {{0.29,0.41,0.35,-
0.02,-0.39,0.19,0.06,-0.36,0.46},{-
0.4,0.34,0.31,-0.06,-0.05,-0.1,-
0.49,0.38,0.23},{-0.23,0.28,0.3,-0.5,-
0.29,0.13,-0.2,-0.28,0.42}};
#endif //WEIGHTS_H
```

Figure 9 : Example Header File

4. RESULTS

Three different scenarios were investigated to determine the capability of the artificial neural network being used to control the mobile robots. The first was a simple black line follower, secondly a black line follower with collision avoidance and thirdly a follow-me function in a less constrained environment (one not using the black line follower).

4.1 Black Line Follower

The training values in figures 7 and 8 were used to implement the ANN version of the black line follower. A simple oval shape was used for this purpose, see figure 10 as the objective was to see how good the ANN was at following a relatively straight black line. By using an oval then only clockwise orientated turns (if travelling clockwise) or anti-clockwise turns (if travelling anti-clockwise) would be needed, rather than a combination of the two. It is intended to test the ANN with alternating clockwise and anti-clockwise turns at a later time. The Bluetooth communication link was used to capture the data. The data from the three black sensors and the motor driver outputs was plotted as graphs. Figure 11 illustrates the data obtained from the middle black line sensor. The mobile robot was swerving left and right of the black line as the motor control implemented by the ANN was a simple ON/OFF as shown by the periodic nature of the signal. The two portions where there does not seem to be much oscillation corresponded to the straighter sections on the longer sides of the oval.

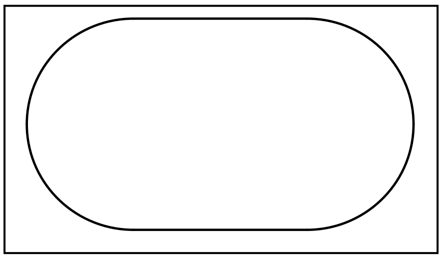

Figure 10 : Oval Shape Used for Testing

Figure 12 illustrates the right hand motor drive values produced by the ANN. Although the ANN produced a variable drive value between 0 and 255 the PCB available was not able to produce a variable voltage output to control the speed of the DC motor. Instead a threshold value of 196 was selected and for any value below this threshold the motor was turned off and for values above this threshold the motor was turn full on. The two straight sections of the oval can be clearly seen where the motor was continuously on.

4.2 Collision Avoidance

A distance sensor was added to the front of the mobile robot in order to be able to implement a collision avoidance function for multiple mobile robots placed onto the same black line. This required a fourth neuron to be added to the ANN input layer and new training data values. The updated training data set is illustrated in figure 13. The first half of the training data set was effectively the same as in figure 7 but with the fourth input listed and shown as 0.2 to represent no obstacle detected. The second half of the training data set had the distance sensor set at a normalised threshold value of 0.4 which represented a distance of approximately 15 cm. In the second half of the training data set the outputs to the motors were always set to 0.0 (motors off) no matter what the input sensor values are. With this additional input neuron and updated training data set the training time increased from approximately 1 minute to over three minutes. This was not a significant problem as the training only took place once when the mobile robot was turned on. Additionally, by using off-line training as describe in section 3.3 this long delay on start-up could be eliminated.

```
{ 0.1, 0.1, 0.1, 0.2 },  // No obstacle
{ 0.1, 0.1, 0.9, 0.2 },  //
{ 0.1, 0.9, 0.1, 0.2 },  //
{ 0.1, 0.9, 0.9, 0.2 },  //
{ 0.9, 0.1, 0.1, 0.2 },  //
{ 0.9, 0.9, 0.1, 0.2 },  //
{ 0.9, 0.9, 0.9, 0.2 },  //
{ 0.1, 0.1, 0.9, 0.4 },  // Obstacle detected
{ 0.1, 0.9, 0.1, 0.4 },  //
{ 0.1, 0.9, 0.9, 0.4 },  //
{ 0.9, 0.1, 0.1, 0.4 },  //
{ 0.9, 0.9, 0.1, 0.4 },  //
{ 0.9, 0.9, 0.9, 0.4 }  //
```

Figure 13: Training Data with Obstacle detection

A more significant problem was that the increase in the number of floating point operations needed resulted in a longer sampling time and this led to a more unstable black line following outcome as the slightly longer time that the motors were on for resulted in the mobile robot losing the black line altogether on a more frequent basis. With some careful adjusting of other parameters it was possible to reduce the sampling time again so that the system operated effectively but this did indicate the processing limitations of the microprocessor used. With this training data the mobile robots were able to follow the black line and at the same time maintained a minimum distance between adjacent mobile robots.

4.3 Clustering

A second aspect of multiple mobile robots that was implemented was clustering [7] where a number of mobile robots are placed into an environment and their objective was to gather together. The control process needed to achieve this was simple with the mobile robots moving about either randomly or in some systematic search pattern mode until another mobile robot was detected. At that point they switched to a clustering mode whereby they tracked the detected mobile robot. The tracking mode was implemented using the same ANN training data as for the black line following mode but with the three black line sensor inputs replaced with three distance sensor inputs. No other changes were made and although the data obtained from the distance sensors was not exactly the same as that received form the black line sensors it was sufficiently similar that the ANN trained with the black line following data was also able to cluster the mobile robots together.

5. DISCUSSION

One of the purposes of this activity was to work with multiple mobile robots that were not exactly the same to determine how effective the artificial neural network would be at accommodating differences in implementation. By allowing a greater variation in performance then the mobile robots can be made more cost effectively and for swarm robotics where larger numbers of individual mobile robots are required, this was an important factor. The ANN used has been shown to be good at coping with differences between individual mobile robots and achieving the control outcomes required. It has also been demonstrated that the same training data can be used for different control functions by using different sensor inputs and this illustrated the flexibility of this biologically influenced control mechanism.

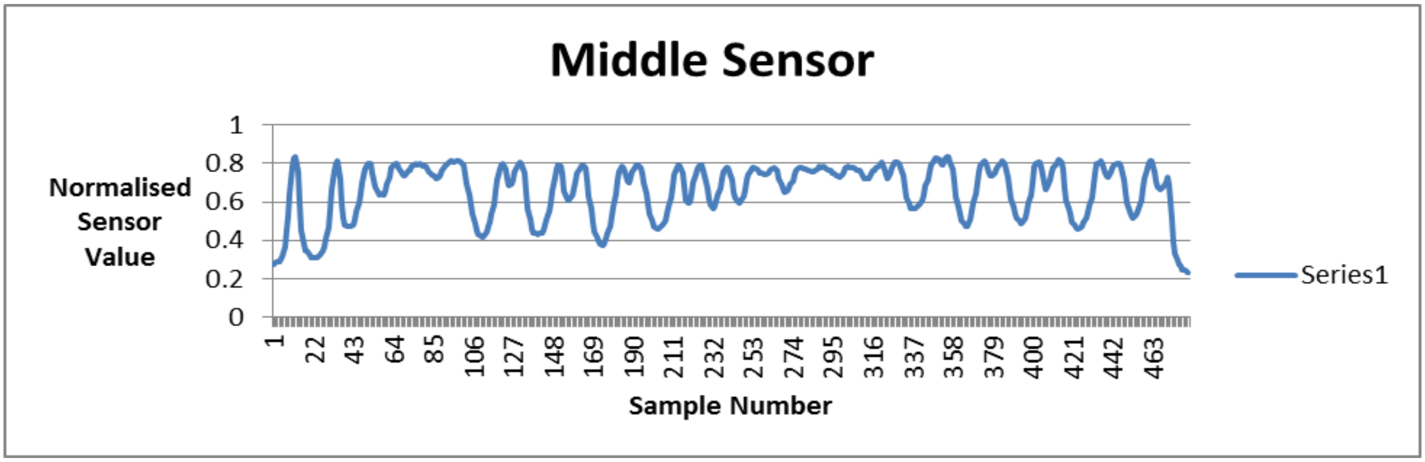

Figure 11 Middle Sensor Values obtained for a Complete Oval

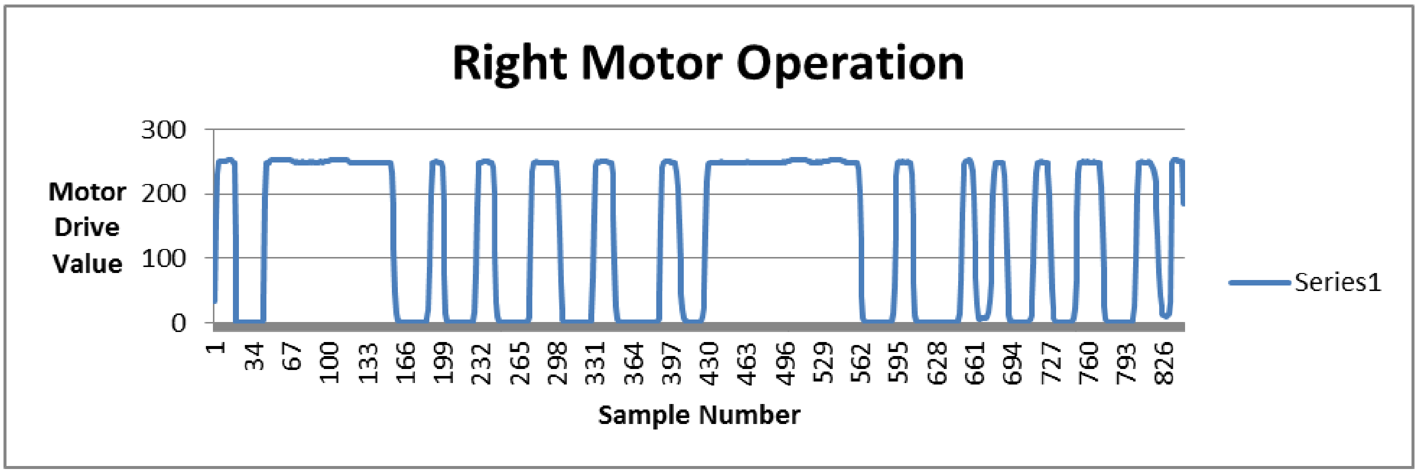

Figure 12 : Right Motor Drive Values

The artificial neural networks were easy and simple to train and required only example sensor input patterns and associated output actuator drive values so that complex control theory was not needed.

The use of artificial neural networks has been effective in creating a group of small mobile robots capable of following a black line, as well as obstacle avoidance and clustering. A low cost microprocessor was capable of implementing the numerous floating calculations required although this was at the limit of capacity for the number of sensors used. A larger number of sensors or more complex sensors would require a more powerful microprocessor.

6. CONCLUSION

This experimental activity has illustrated that it was possible to create biologically inspired control functions for small mobile robots that were to be used in swarm robotics research. The control functions needed were effectively implemented by using a small number of training vectors that were easy and simple to amend as needed for different scenarios. There were limitations caused by the large number of floating point calculations needed to implement the ANN but a more powerful microprocessor would easily overcome this problem. Different sensors were easy to include or remove, and this required only minimal changes to the software implementing the ANN and the corresponding training data.

7. REFERENCES

[1] J. C. Barca and Y. A. Sekercioglu, “Swarm robotics reviewed,” *Robotica* **31**(3), 345–359 (2013).

[2] TCRT5000 Reflective Optical Sensor with transistor Output, http://www.vishay.com/docs/83760/tcrt5000.pdf

[3] GP2Y0A21YK0F Distance Measuring Sensor Unit, http://www.sharp-world.com/products/device/lineup/data/pdf/datasheet/gp2y0a21yk_e.pdf

[4] Principe, Jose C., and Craig L. Fancourt. Artificial neural networks. *Handbook of Global Optimization* 2 (2013): 363-386.

[5] Gupta, Vibhor, Working and Analysis of the H-bridge motor drive circuit designed for wheeled mobile robots, 2nd International Confernce on Advanced Computer Control (ICACC), (2010), ISBN 978-1-4244-5847-9.

[6] Arduino UNO, https://www.arduino.cc/

[7] Manuele Brambilla, Eliseo Ferrante, Mauro Birattari, Marco Dorigo. Swarm Intell Swarm robotics: a review from the swarm engineering perspective. Swarm Intelligence, Springer, 2013, 7 (1), pp.1-41. .