Whole Iguanas: Autonomy in Interaction

Richard T. Vaughan
Robotics Research Labs.
University of Southern California

David McFarland
Department of Zoology
University of Oxford

May 3, 2000

Abstract

Rover is a mobile robot that gathers a flock of ducks and manoeuvres them safely to a specified goal position. As such it is a valuable example of a robot that performs compleex real-world task by interacting with other agents. This paper examines the implications and requirements of making the robot sheepdog a 'Whole Iguana'; an autonomous artificial creature as desribed by Dennet [7].

1 Introduction

This paper examines the requirements for a complete creature; a 'whole iguana' whether animal or robot, when engaged in significant interaction with other agents. The outcome of any behaviour performed by our interacting creature depends on the actions of others as well as itself.

McFarland has asserted that animals behave as if they are maximizing cost functions, that is they act to optimize the values of a set of internal variables [**?**]. A motivation is identified with such a variable or collection of variables. For example an animal must maintain its body temperature within certain bounds; in hot sun it is motivated to find shade. The same behaviour may satisfy more than one motivation, or may be contrary to some while satisfying others.

This paper describes a real robot system that strongly interacts with animals. We show how a simple model of relevant animal motivation was used as a tool to help design a closely related robot controller. After presenting the system's design and behaviour in some detail, we discuss how the robot controller could be extended to manage its energy budget while still performing useful interaction.

Useful robots manipulate objects. Biological objects have driven robotics research as their mechanical properties are often more challenging and more variable than artifacts. Specialized robots have been developed for mushroom picking [**?**], tomato packing [**?**], and peach sorting [**?**].

A robot that manipulates behaving objects such as animals or other robots has particular constraints, such as real time operation and safety. But the main problem and opportunity of manipulating other creatures is that they have their own motivations that may confound or confirm your own. An animal may not remain where you left it; on the other hand it may move itself where you wanted it. This is the essential chracteristic of interaction: the actions of others have an effect on you and vice versa. The behaviour of other agents is a part of the ecological niche of the interactor. An interactive whole iguana must be designed to accomodate and exploit the behaviour of others. This can be done by incorporating the motivations of external agents into the iguana's motivational system, whether explictly or otherwise.

2 Robots and animals

Few extant robot systems are concerned at all with (live) animals. Those that do are almost all concerned with avoiding them, eg. *Herbert* the can-collecting robot which avoids bumping into human office workers [5]. Robot-like machines designed to have a direct effect on humans include medical tools (largely tele-operational, eg. [1]) and aids for the disabled (eg. [25]).

Those concerned with non-human animals include a robot sheep shearer [19], and a dairy cow milking robot [8]. The sheep shearing system incorporates a special restrainer to minimize the sheep's movement while the robot shears the fleece. The milking robot is part of a larger system which exploits cow behaviour. The cows come into the milking parlour when they choose, and are milked and fed without human intervention. This gives advantages in welfare and (potentially) labour efficiency [14]. Once in the parlour though, the cow is constrained in a stall while the robot attaches the milking equipment. Neither of these systems are interactive in any strong sense. Both rely on keeping the animal as still and non-reactive as possible while the robot does its job.

The next section describes in detail Vaughan's work on a robot that strongly interacts with real ducks.

3 Rover the Robot Sheepdog

The Robot Sheepdog Project (RSP) demonstrated a robot system that gathers a flock of ducks in a circular arena and manoeuvres them safely to a pre-determined goal position, then holds them there indefinitely. It was the first robot to meaningfully control the behaviour of a non-human animal.

The RSP was a collaborative, multidisciplinary project that reported gresults in machine vision [16, 17, 18], behavioural modeling and ethology [10], as well as the central robotics experiments [20, 21, 22, 23]. The central research and methodological goal was to develop a machine that could usefully interact with an animal *without using the animal directly in the development process.*

The sheepdog's gather-and-fetch task was chosen because of its familiarity and the strong interaction between the dog, shepherd and flock animals. Using ducks instead of sheep allows us to experiment on a conveniently small scale, in a controlled indoor environment. For similar reasons ducks are often used to train sheepdogs, as seen in Figure 1.

The robot task we examined was deliberately restricted to allow us the maximum chance of success in the three-year life of the project, while still offering an interesting demonstration of robot/animal interaction. This task presents a major difference from conventional robot applications in that the objects cannot be manipulated directly, but must be influenced to move *themselves* to the goal. We sought to identify those aspects of duck behaviour which make them controllable, and to design a herding strategy to exploit those features and effectively control the flock.

In order to identify the appropriate robot-animal interactions we built a minimal generalised model of the underlying flock behaviour. We hypothesised that if the model accurately captures the basis of the behaviour, then a system which controls the model should control the real-world behaviour.

Models of flocking behaviour exist in the literature and are generally derived from Hamilton's observation that flocking may be produced by the mass action of individual animals, each seeking the proximity of its nearest neighbours [9]. It was later suggested that this behaviour can be well model-led by an attractive 'force' acting between the animals, with the magnitude of the attraction varying with the inverse square of the animals' mutual distance [13] [24]. It is argued that this relationship represents a linear response to sensory

Figure 1: A young sheepdog in training with a group of ducks.

information which itself varies with the inverse square of distance. Similar models have produced realistic computer animations of bird flocks [15]. Flocks of mobile robots have also been demonstrated [12].

These ideas are familiar in robotics, where such *potential field* techniques are used for path planning [11, 6]. This class of algorithm uses the analogy of forces acting on particles, such that the robot will move as if it were a particle attracted or repelled from features in its environment. A robot is typically attracted to a goal position and repelled from obstacles.

The commonality of these animal and robot behaviour models forms the basis of an effective flock-gathering strategy, described below.

3.1 System Overview

The experimental system comprises a robot vehicle, a workstation and a video camera (Figure 2, left). The vehicle was designed to work in a duck's environment: outdoors, on short grass, and in real time. Thus our robot has an acceleration $\approx 1\text{ms}^{-2}$ and a top speed $\approx 4\text{ms}^{-1}$, which is about twice as fast as the ducks. It is covered in a soft plastic bumper mounted on rubber springs, ensuring duck safety. In the tradition of mobile robotics, we call it 'Rover' (Figure 2, right).

The vehicle and ducks are free to move in a visually uniform arena of 7m diameter, in view of the overhead camera. The arena is shown in Figure 3 (left). The positions of the robot and flock are determined by processing the video image stream. The robot's position and orientation are found by matching a template of its black and white cover to a region of the image. Tracking the ducks was a more unusual vision task and although an ideal system would track the positions of individual ducks, it was concluded early on that there were no reliable, fast methods available to achieve this. However, it seemed likely that we could track the whole flock as one object, with some measurement of its size and shape. Such 'blob detectors' are common in machine vision, and are implemented via standard techniques such as background subtraction and thresholding (see, for example, [3]). Flock

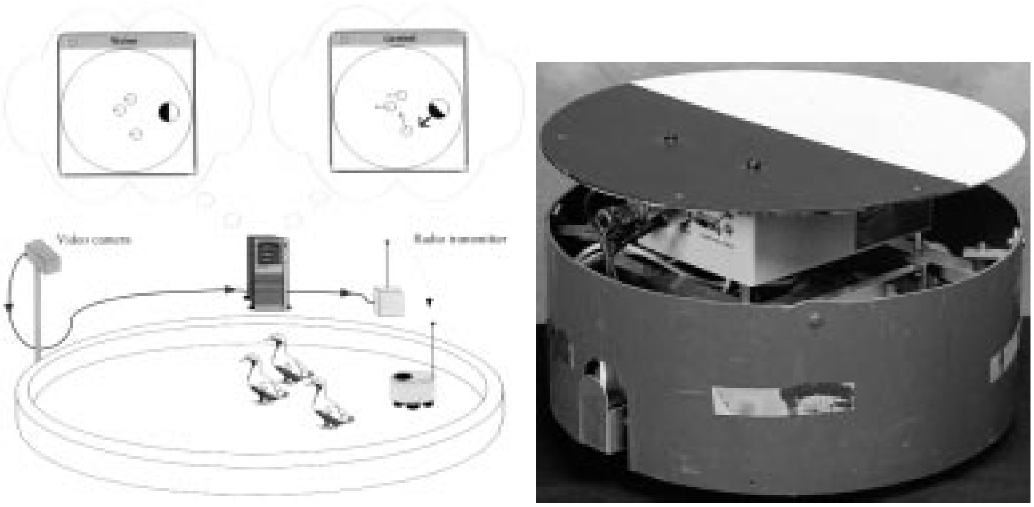


Figure 2: Robot Sheepdog system overview (left) and vehicle (right)

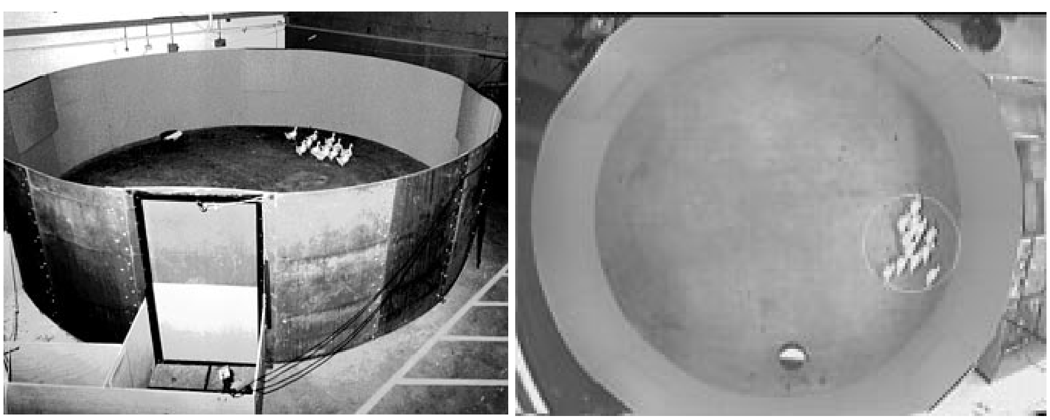
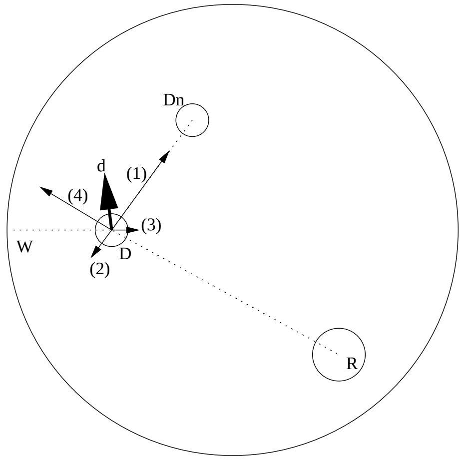


Figure 3: The experimental arena (left) and view from overhead camera (right). Note the positions of the robot and flock overlaid by the tracker



$$\vec{d} = \sum_{n=1}^{N}\left(\left(\frac{K_1}{(|\vec{DD}_n| + L)^2}\right)\widehat{DD_n} - \left(\frac{K_2}{|\vec{DD}_n|^2}\right)\widehat{DD_n}\right) - \left(\frac{K_3}{|\vec{DW}|^2}\right)\widehat{DW} - \left(\frac{K_4}{|\vec{DR}|^2}\right)\widehat{DR}$$

(1) (2) (3) (4)

Figure 4: Flock model (schematic not drawn to scale). Key: gain parameters $K_{1\to4}$; repulsion bias parameter L (ensures repulsion > attraction at small distances, preventing collisions); ducklet position D, other ducklet D_n; Robot position R; Nearest point on wall W; algorithm terms ($1 \to 4$) and resultant velocity $\vec{d}$ (where $\widehat{a}$ is the unit vector of $\vec{a}$).

position was defined as the as the centre of area of the detected flock 'blob'.

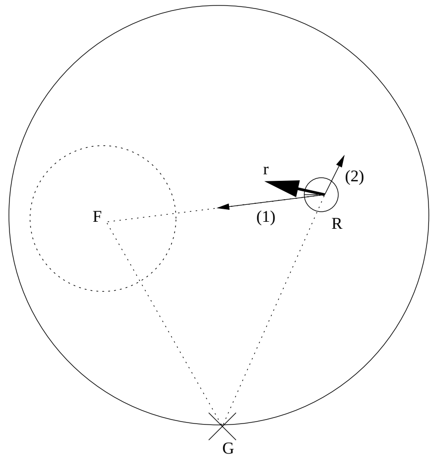
This gave an interesting constraint to the rest of the system; it would have to work without knowing the positions of individual birds, but only with a centre position, size and shape. In fact, the flock control algorithms that were devised do not require the shape information, so it was possible to abandon the shape-finding and produce a very fast tracker that finds just the centre and radius of the flock. The final vision system runs very quickly (update frequency > 25Hz), and has proved adequate for these experiments. Figure 3 (right) shows a example image with the robot and flock correctly identified.

The robot's movement is guided by a flock-control algorithm running on the workstation. This algorithm takes the vision data (positions of the robot R, flock F and goal G) as input and returns a desired vehicle trajectory $(R, F, G) \to \vec{r}$. This is passed to the robot by radio modem, and a conventional high-frequency proportional controller governs the robot's wheel speeds to closely approximate this vector.

3.2 A model flock

A minimal simulation model of the duck-herding scenario was created, in which a flock of model ducks (ducklets) moves in a circular arena containing a model robot.

A potential field algorithm is used to generate movement for each ducklet. Given a ducklet's position D, the positions of the N other ducklets $D_{1\to N}$, the robot's position R and the nearest point on the wall W, the ducklet's movement vector $\vec{d}$ is determined by the function shown in Figure 4. The ducklets are (1) attracted to each other, aggregating the flock; (2) repelled from each other, preventing collisions and maintaining inter-ducklet spacing; (3) repelled from the arena wall, preventing collisions. A further term (4) which produces repulsion from the robot is proposed to model the aversive response of the duck-



$$\vec{r} = (K_{r1}|\vec{GF}|)\widehat{\vec{RF}} - K_{r2}\widehat{\vec{RG}}$$
$$(1) \qquad\qquad (2)$$

Figure 5: Robot controller(schematic not drawn to scale). Key: gain parameters $K_{1,2}$; flock centre F; Robot position R; Flock goal position G; algorithm terms ($1 \rightarrow 3$) and resultant $\vec{r}$ (where $\widehat{a}$ is the unit vector of $\vec{a}$)

lets to the robot. Note that all these forces are scaled according to the inverse square of distance. Each ducklet moves according to the resultant of the forces acting upon it, subject to a simulated inertia that smoothes acceleration, and limited by a top speed chosen to approximately match that of the real ducks. The simulation produces a realistic-looking flock which can be manipulated by steering the model robot.

Note that the model describes a small subset of the ducks' behaviour. Of course, many other mechanisms generate the behaviour of real ducks, but our hypothesis is that this model captures enough of the real animals' behaviour to be a useful design tool. The model is a *generalised* description of flocking behaviour and as such could be applied to any flocking animal in two or three dimensions.

3.3 Experiments

Experiments with the simulator guided the development of two novel flock control algorithms which are closely related to the flock model described above. Only the most recent and successful of these is described here. The original method is described along with our first results in [22].

The distance $|GF|$ in Figure 5 is the system variable we are trying to control, ie. reduce to zero. In a classical proportional controller a control output would be applied to correct this variable, with a magnitude proportional to the size of the error. If we include this term in the flock controller, we can design an analogous system whereby the repelling stimulus experienced by the ducks is proportional to their distance from the goal.

The robot's movement vector $\vec{r}$ is given by the function shown in Figure 5. The robot is (1) attracted to the flock with magnitude proportional to the distance from the flock to the goal; (2) repelled from the goal with constant magnitude. Note the simplicity of the algorithm and that it is expressed in similar terms to the flock model.

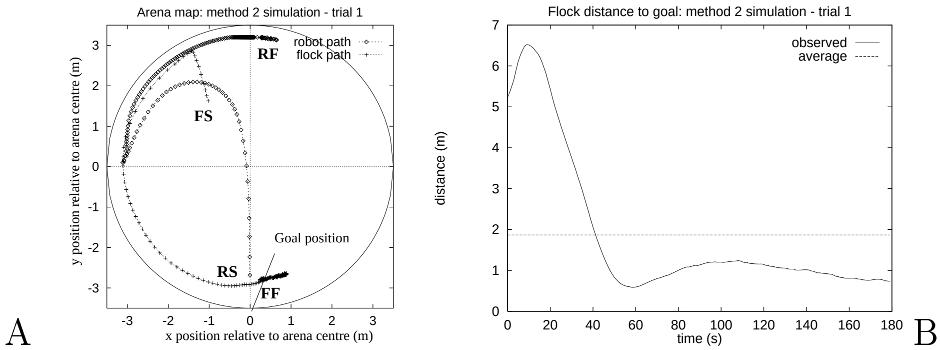
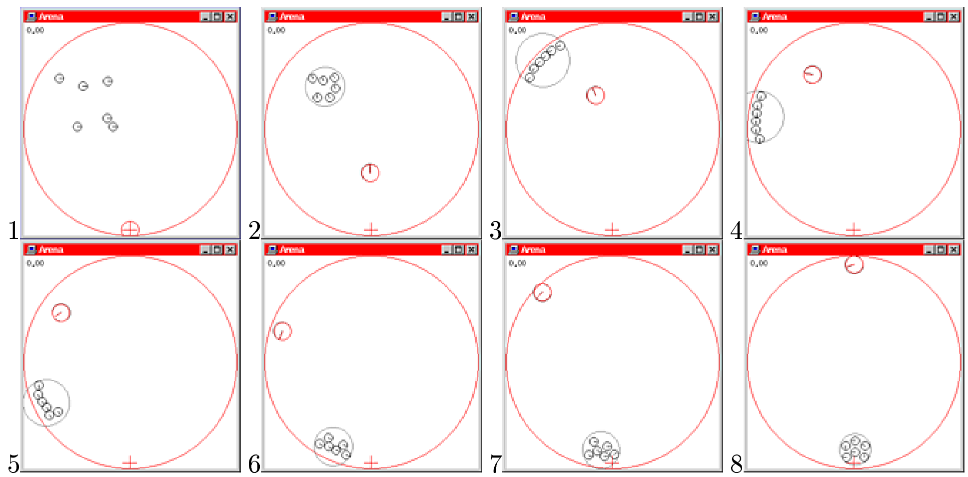


Figure 6: Simulation results: (A) paths in arena [Key: RS = robot start, RF = robot finish, FS = flock start, FF = flock finish] and (B) distance to goal over time.



Key: ◯ = robot, ○ = ducklets, + = flock goal

Figure 7: Sequence of images from the simulator during a trial.

3.3.1 Performance in simulation

The algorithm is first tested in simulation. A point on the arena boundary is chosen as the flock goal, twelve ducklets are placed randomly in the arena, and the robot positioned near the goal. The simulation starts and the positions of the robot and flock centre are recorded for the next 3 minutes, as the robot attempts to manoeuvre the flock to the goal. This experiment was repeated nine times with the ducklets at different random start positions, and the robot at a slightly different position near the flock goal in each trial.

The results show that this controller successfully performs the required task. Figure 6 (A) shows a representative plot of the simulated robot and flock paths around the arena, while Figure 7 shows a series of screenshots from a similar trial. It can be seen that the flock is brought near the goal. The success plot Figure 6 (B) shows the distance of the flock to the goal over the length of the trial, plus the average distance over the entire trial. This is used as a measure of the the trial's success for comparison with other experiments. It can be seen that the flock-to-goal distance decreases rapidly then stabilises as the ducks settle near the goal. This trial scores an average flock-to-goal distance of 1.9m. The average score

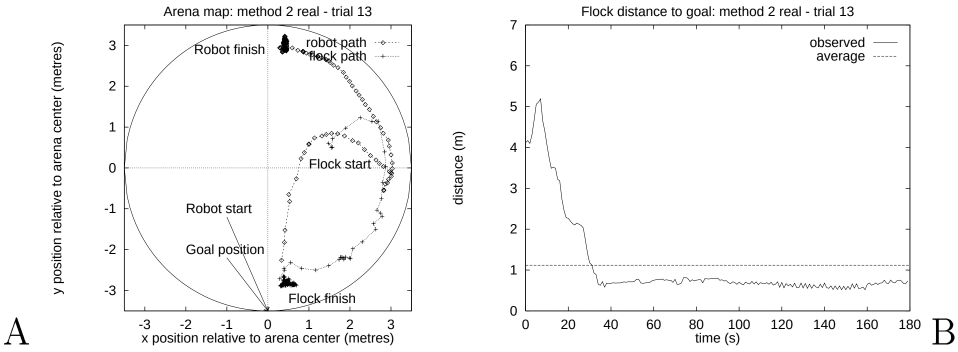


Figure 8: Real-world results: (A) paths in arena and (B) distance to goal over time.

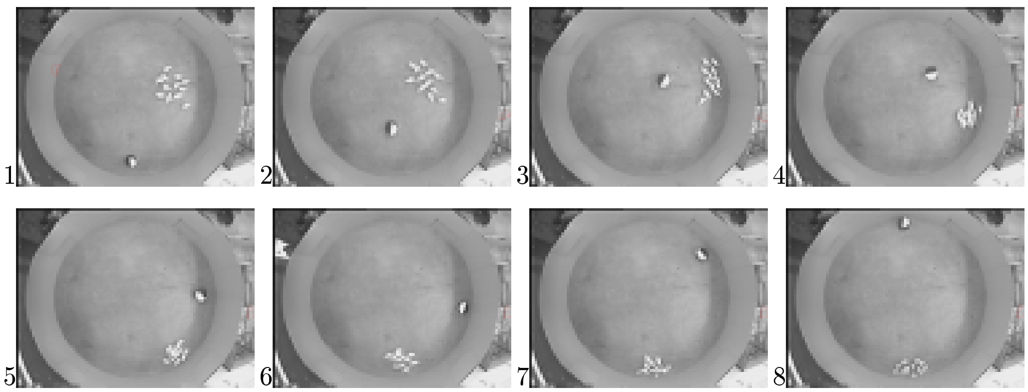


Figure 9: Sequence of images from the overhead camera during an experiment. The goal position is at the bottom of the picture.

over all 9 trials was 1.8m, with a standard deviation of 0.16m.

3.3.2 Performance in real world

A similar experiment was then performed in the real world. A random point along the arena boundary is chosen as the flock goal. With the robot inactive and positioned near the goal, a flock of twelve ducks is introduced into the arena. For three minutes the ducks were left to become accommodated to the arena. Their positions were not recorded during this time, but they were typically observed to settle into a stationary, loosely-aggregated group with no common orientation. The settled position of the flock varied apparently at random between trials. After three minutes the robot is activated. The positions of the robot and flock centre are recorded for the next 3 minutes, as the robot attempts to manoeuvre the flock to the goal. At the end of the trial, the robot is deactivated and the ducks move freely again for 2 minutes before being allowed out of the arena. This experiment was repeated seven times with the same flock, with the robot at a slightly different position near the flock goal in each trial. All the ducks were the same age and had been raised under similar conditions (described completely in [10]).

Figure 7 (A) shows a representative plot of the real robot and flock paths around the arena, while Figure 9 shoes a series of overhead camera images from the same trial. It can be seen that the robot approaches the flock, moving round behind them with respect to the goal. The flock moves away from the robot and towards the goal. As the flock approaches the goal, the robot is less attracted to the flock and the goal repulsion becomes dominant. The robot retreats to the far side of the arena, applying minimum stimulus to the ducks. The ducks settle near the goal position. The success plot (Figure 7 (B)) clearly shows the initial fetching phase, followed by the stable, settled behaviour. This trial scores an average flock-to-goal distance of 1.12m. The average score over all seven trials was 1.67m with a standard deviation of 0.28m.

The larger average score in the real world compared to simulation was largely due to an overshoot effect, whereby the flock approached the goal but went past it and had to be fetched back by the robot. This effect is visible in the simulated trial in Figure 6 (the second, wider peak in plot B). The overshoot is caused by moving the ducks too quickly to the goal and not backing away quickly enough. Subsequent trials (simulated and real) have shown that the overshoot can be eliminated by tuning the gain parameter K_1 which controls the amount of attraction to the flock. The optimum setting of this parameter varies from flock to flock, and from day to day. As the success of this method varies (slightly) with this setting, there is scope to devise further algorithms that may be more robust with respect to the inevitable variation between flocks.

3.4 Rover as Behavioural Robotics

Let us characterize the RSP system in terms of Brooks' criteria for Behavioural Robotics:

- Rover is *situated*: its actions are determined by sensing the actions of independent external agents, whose behaviour is not explicitly modeled in the controller.
- Rover is *embodied*: it has a physical vehicle which is used to affect changes in the world in human (and animal) time scales. Its behaviour has an immediate and direct effect on the world.
- Rover is *intelligent*: it does a job which is considered to require intelligence in humans and animals [2]. Its efficacy depends on its correct ongoing interaction with the complex dynamic world.
- Rover's effect on the world is *emergent*: the goal of its actions is predetermined, but its trajectories are not prescribed entirely by its controller. Rather they emerge from interaction with unpredictable external agents. Rover's behaviour is *meaningless* except with respect to the behaviour of the flock.

3.5 Summary

This section presented a robot system that achieves a sheepdog-like task, gathering and fetching live animals to a pre-defined goal position. This is the first automatic system to exploit an animal's behaviour to achieve a useful task. A flock control method was designed and tested using a minimal simulation model of the ducks' flocking behaviour, and successfully transferred directly to the real world.

The robot sheepdog's controller does not require an explicit model of the behaviour of the animals with which it interacts. In this example task, reaction to sensory data is sufficient for a robot to control a system as complex as a flock of ducks. Their threat-avoidance behaviour is reliable enough to be built implicitly into the robot controller; it is as much a part of the robot's environment as the walls and floor.

3.6 Intentional description of Rover's controller

[this section is in note form] - the variables in the sheepdog controller are readily described in intentional language: the robot wants the ducks to be at the goal; the robot wants to move towards the ducks; the robot wants to move away form the goal. these desires are combined in a simple but precise way to produce the desired system behaviour. gain parameters can be adjusted to optimize the outcome in a particular situation.

- central to the function of the controller, though not represented within it, is the ducks' intention: the ducks want to be far away from threatening objects.

- the flock model is also readily expressible in intentional language: the ducks want to be close together; they want to avoid collisions with each other and with obstacles; and (explicitly this time) they want to be far from threatening objects.

- at first sight it seems curious that all our important system variables can be directly expressed as intentions. But this is not surprising if we consider that our intentional language is designed to deal with exactly these interaction scenarios. Intentional language describes the actions of agents as they respond to external stimul and regulate internal state.

- In interactive scenarios the goals of the various agents may be mutually compatible - the ducks want to be close together - or incompatible - the fox wants to eat a duck, the duck wants to survive. They may also be partly compatible, i.e. satisfied in certain conditions - the robot wants the ducks at the goal and wants to be far from the ducks, the ducks want to be far from the robot. The engineering problem in a system like Rover is to arrange things so that the most acceptable state for robot and ducks is the goal state of the designer.

4 The cost of interaction

Chasing the ducks provides a system benefit in terms of progress towards the goal state. However, there are three major costs to chasing ducks:

1. welfare cost - the ducks are stressed by contact with the robot (though we have shown that the ducks exhibit a smaller stress response to the robot that to a human, dog or fox [10]). The operator of the robot desires unstressed ducks.

2. energy cost - the robot expends energy continuously until the ducks are at the goal position. This energy must be recovered somehow.

3. time cost - time spent chasing ducks is lost to other tasks, such as gathering energy.

In any real system there are also subsidiary costs in performing any action in terms of increasing risk of damage and component failure.

None of these costs are explicitly represented in the existing controller. Indeed they can not be directly represented because all are time dependent phenomena and as it stands the controller is purely reactive; it outputs a desired action in response to an instantaneous snapshot of the world.

Inevitably, energy expenditure means that time-dependent costs are a vital part of the motivational make up of a complete creature. However, let us be clear that the creature does not necessarily need to represent time explicitly and have a 'sense of time'. The time-related state variables discussed can be maintained by integration, or by monitoring a physical state that varies with time such as the metabolism of sugar. This is analagous to a a battery charge indicator that codes in some sense for both energy and time expenditure: it is instantaneous value that corresponds to a temporal process. It would be simple to incorporate such a sensor and perform actions conditional on the value of this parameter.

In the sheepdog controller, the distance from the goal state (controller variable $|G|$) could be used as an estimate of the

4.1 The risk in suspension

The sheepdog task cannot be halted and resumed at no cost. Many conventional robot tasks, such as a solitary robot stacking bricks, can be halted and resumed with little cost; the arrangement of bricks will be the same at suspension and resumption. In interactive tasks the state is likely to change once the controlling behaviour is aborted. In our example task, soon after the robot ceases to present a threat, the ducks will resume feeding, social behaviour, etc. and move around the arena. It is likely that they will be further from the goal condition when the task is resumed. The penalty to the system is that the cost of completion following suspend-and-resume is likely to be higher than completing before suspending.

Once suspended the ducks will tend to diverge more and more from the goal condition. Thus suspending the task becomes more costly the longer it is suspended. Even worse, the future cost can be effected by the choice of new activity, as the presence and action of the robot will continue to influence the ducks' behaviour even if it does not intend to. Thus to minimize future cost of completion the robot should choose alternative behaviours that disturb the state of the ducks as little as possible.

5 Motivational priorities

Internal state variables that are relevant to the viability of an animal place behavioural demands upon the individual. The same principle applies to self-sufficient robots (McFarland and Bosser, 1993). Thus the robot must continually monitor the relevant state variables, such as on-board energy availability, and be able to take appropriate action when necessary. This means that it must be possible for the ongoing behaviour to be stopped and superceded by behaviour of higher priority. In animal behaviour studies, this topic comes under the general heading of motivational priorities (McFarland, 19xx). As a robotics topic, this is an example of *multiple-objective action selection* [**?**, **?**].

Thus if the sheepdog robot were an iguana, it would be designed to fulfil tasks other than rounding up ducks, such as obtaining energy and avoiding danger. Such design would inevitably involve the problem of motivational priorities.

In its general form, the problem is that a particular behavioural system, such as that for rounding up ducks, is in competition with other behavioural systems for control of the behavioural final common path (McFarland and Sibly, 1975). However, it may not be entirely clear which variable, of a particular behavioural system, is the one that counts in the competition. As we have seen, the cost of terminating the rounding-up behaviour increases as the behaviour progresses. We would expect the competitiveness of the system to be related to this cost, but there is no variable within the system that is the obvious representative of this cost.

One solution to this problem, first formulated by x and y, is that the cost is implicit in a combination of variables that 'represent' the system in competition with others. For state variables that have a lethal boundary (e,g, hunger, thirst, on-board energy), the cost function is quadratic, and the optimal decision amongst alternatives of this type can be represented by a simple rule. In its general form this rule is of the following form:

If deficit x availability for behaviour A > that of behaviour B, then do A, else do B.

The deficit is the extent to which the state variable in question (e.g. on-board energy) is below maximum (i.e. hunger). The availability is the perceived (by the animat) indication of the resource (e.g. food) availability in the environment (measured in terms of the rate of change of deficit that doing the appropriate behaviour would engender). For reviews of this body of theory see REFERENCES. Spier and McFarland (DATES) extended the theory to include the accessibility of the resource (a limit on the rate at which the animat can exploit the resource, as a result of its own behavioural skills). They showed that this formulation was superior to others in a series of simulation tests.

The problem is that implicit rules of this type are hard to find, and require a type of reverse engineering that is difficult to perform on animals (McCleery ref). Nevertheless, the point remains, that some combination of variables representative of a particular motivational system must enter the competition with those of other systems. Moreover, motivational competition theory assumes that the competition between the representatives of behavioural alternatives takes place on a level playing field. In other words, every type of motivational system has a chance of scoring. This may seem obvious, but historically it has not always been the case. For example, it was once thought that grooming behaviour took place only between 'gaps' in behaviour, when the animal had nothing better to do.(Rowell)

Returning to the sheepdog robot HERE WE NEED TO SUGGEST ALTERNATIVE COMBINATIONS OF VARIABLES FROM YOUR SHEEPDOG CONTROL SYSTEM, THAT MIGHT BE CANDIDATES FOR COMPETITION WITH OTHER MOTIVATIONAL SYSTEMS WITHIN THE IGUANA ??

[THIS PART STILL TO DO - I'M STILL NOT SURE ABOUT THE STRUCTURE AND MESSAGE OF THE PAPER - LOOKING FORWARD TO YOUR COMMENTS, EDITS AND ADDITIONS - RV]

References

[1] F. Arai, M. Tanimoto, T. Fukuda, K. Shimojima, H. Matsuura, and M. Negoro. Multimedia tele-surgery using high speed optical fiber network and its application to intervascular neurosurgery. In Caplan and Lee [4].

[2] Margaret A. Boden. *Artificial Intelligence and Natural Man.* MIT Press, Cambridge, MA., 2 edition, 1987.

[3] R. Boyle, V. Hlavac, and M. Sonka. *Image Processing, Analysis and Machine Vision.* Chapman and Hall, 1993.

[4] N. Caplan and C. S. George Lee, editors. *IEEE International Conference on Robotics and Automation*, April 22-28 1996.

[5] Jonathan H. Connell. *Minimalist Mobile Robotics.* Academic Press, Boston, 1990.

[6] Christopher I. Connelly and Roderic A. Grupen. On the applications of harmonic functions to robotics. *Journal of Robotic Systems*, 10(7):931–946, October 1993.

[7] Daniel C. Dennett. Why not the whole iguana? *Behavioral and Brian Sciences*, pages 103–104, 1978.

[8] A.R. Frost, M.J. Street, and R.C. Hall. The development of a pneumatic robot for attaching a milking machine to a cow. *Mechatronics*, 3(3):409–418, 1993.

[9] W.D. Hamilton. Geometry for the selfish herd. *Journal of Theoretical Biology*, 31:295–311, 1971.

[10] Jane V. Henderson. *Adaptive Responses of Herding Animals to a Robot Vehicle.* PhD thesis, University of Bristol, 1999.

[11] Oussama Khatib. Real-time obstacle avoidance for manipulators and mobile robots. In *Proceedings of the 1985 IEEE International Conference on Robotics and Automation*, pages 500–505. IEEE, March 1985.

[12] M.J. Mataric. Designing and understanding adpative group behavior. *Adaptive Behavior*, 4(1):51–80, 1996.

[13] B. L. Partridge. The structure and function of fish schools. *Scientific American*, pages 114–123, June 1982. June 1982 issue.

[14] Neville B. Prescott. *Dairy cow behaviour and automatic milking.* PhD thesis, University of Bristol, 1995.

[15] Craig W. Reynolds. Flocks, herds and schools: A distributed behavioural model. *Computer Graphics*, 21(4):25–34, jul 1987.

[16] N. Sumpter, R.D. Boyle, and R.D. Tillett. Modelling collective animal behaviour using extended point distribution models. In *Proceedings of the British Machine Vision Conference*, 1997.

[17] N. Sumpter and A. Bulpitt. Learning spatio-temporal patterns for predicting object behaviour. Technical report, University of Leeds, School of Computer Studies, The University of Leeds, UK, 1998.

[18] N. Sumpter, A. Bulpitt, R. Vaughan, R. Tillet, and R. Boyle. Learning models of animal behaviour for a robotic sheepdog. In *Proceedings of the IAPR Workshop on Machine Vision Applications*, 1998.

[19] James P. Trevelyan. *Robots for Shearing Sheep: Shear Magic.* Oxford University Press, 1992.

[20] R. Vaughan, N. Sumpter, A. Frost, and S. Cameron. Experiments in automatic flock control. In *Proceedings of the Sixth Symposium on Intelligent Robotic Systems (SIRS'98)*, 1998.

[21] R. Vaughan, N. Sumpter, A. Frost, and S. Cameron. Robot control of animal flocks. In *Proceedings of the Joint Conference on the Science and Technology of Intelligent Systems (ISIC/CIRA/ISAS'98)*. IEEE, 1998.

[22] R. Vaughan, N. Sumpter, A. Frost, and S. Cameron. Robot sheepdog project achieves automatic animal control. In Rolf Pfeifer, Briuce Blumberg, Jean-Arcady Meyer, and Stewart Wilson, editors, *From Animals to Animats 4: Proceedings of the fifth international conference on the simulation of adaptive behaviour*, Cambridge, Mass., September 1998. MIT Press.

[23] R.T. Vaughan, N. Sumpter, J.V. Henderson, A. Frost, and S. Cameron. Experiments in automatic flock control. *Robotics and Autonomous Systems*, to appear 2000.

[24] K. Warburton and J. Lazarus. Tendency-distance models of social cohesion in animal groups. *Journal Of Theoretical Biology*, 150:473–488, 1991.

[25] T. Yukawa, M. Khalid, M. Uchiyama, H. Inooka, T. Oshashi, G. Obinata, H. Kagaya, and Y. Shimada. Modular hybrid functional electrical stimulation system. In Caplan and Lee [4].