

Robot Flocking: Sensors and Control

Lenka Pitonakova
contact@lenkaspace.net
University of Sussex, January 2010

Abstract

This paper discusses various kinds of robot sensory input, approaches to motor control and ways they could be used for flocking. Focus is put on vision and Gibsonian optic flow that could be utilised by robots with advanced behaviour. Also, infrared sensors and their advantages and disadvantages are discussed. Finally, the paper gives an overview of currently available robots which demonstrate collective behaviour and speculates how and whether adding 2D-image-based vision could give an advantage in terms of higher-level behaviour.

Keywords

Flocking, optic flow, infrared vision and communication, collective robot behaviour

1. Introduction

Although building an autonomous intelligent robot able to survive and make decisions on its own would certainly be a great milestone in science, its engineering would probably be very difficult and expensive. Fortunately, the nature proves that intelligence and high-level behaviour can also be achieved by cooperation of much simpler agents. So-called ‘swarm intelligence’ can be mostly observed in insect colonies. A colony is able to survive for much longer than an individual insect and is able to perform tasks not observed in individuals including choosing the best of available food sources (Seeley T.D. et. al., 1991), or attacking a larger predator. Levy S. (1992) mentions NASA’s space program which proposed using a colony of intelligent robots which could self-replicate and gather resources. The robots themselves could be small and cheap, yet the colony would have much better chance of surviving and providing results in comparison with a single heavy robot (Brooks and Flynn, 1989), mostly because success of a mission would not depend on one individual (Seeley T.D. et. al., 1991).

The topic of this paper is robot flocking. The author believes that flocking and swarm movement are the first step in building robots that can cooperate autonomously. Firstly, movement is a basic characteristic of most of the living organisms. Secondly, robots which would move in a flock would have to be aware of each other and act in cooperation with others, thus provide a basic feature of a swarm.

2. Flocking Inside of a Computer

When Craig Reynolds (1987) wrote his paper about flocks he argued that traditional animation techniques (like pre-defining a path for each member of a group) would be very difficult to script, maintain and edit. Moreover, there were not many applications that could generate animations automatically. Reynolds suggested simulating behaviour of only one member, duplicating this member and let it interact with other identical units. More importantly, he believed that to get a realistic behaviour one had to simulate a bird’s perception as well physics of the environment.

Reynolds’ ‘boids’ were able to fly in a simulated 3D space using three simple behavioural rules. Although Reynolds himself did not use the exact terminology, further work by Bourg and Seeman (2004: 53) or Moeslinger et al. (2009) uses them.

- **Cohesion rule** - each unit tries to stay inside of a group by steering towards average location of neighbours it can see
- **Alignment rule** - a unit adjusts its heading and speed (commonly referred to as velocity) according to its neighbours. This results in a common direction of a flock.
- **Separation rule** - units tend to turn away if they are too close to a neighbour

According to Reynolds' model (1987) each of the behaviours forms and 'acceleration request' and different requests are added together to determine a resultant course. A problem could arise when the requests cancel each other or produce a move towards an obstacle. Reynolds suggests 'prioritized acceleration allocation' where each request has a predefined priority. A 'navigation module' uses the priorities and a maximum acceleration value to control the motion by selecting which behaviour needs to be satisfied first (e.g. obstacle avoidance versus aligning with neighbours) and whether a low-priority behaviour should be suppressed in a potentially dangerous situation.

As a result of the extensive work by Reynolds and others, flocking seems to be a solved problem when it comes to the algorithm itself. It is used today's in games like Flock! (Gamershell: Flock) or Weapon of Choice (YouTube: Weapon Choice) and animations for movies like Lion King or Batman (MgTaylor).

To extend the algorithm further, Bourg and Seeman (2004:55) suggest that each unit should have a specified field of view (a constant angle shared by all units) that sets how much around it can see. They showed that a wide field of view produces widely spread flocks, while a very narrow field of view results in a moving queue of units.

There are of course many other parameters a programmer could play with, including how close the units are allowed to be to each other, how rapidly they turn, what their speed is, etc. Bourg and Seeman (2004:73-79) give an algorithm for obstacle avoidance and the follow-the-leader-rule. The latter was also mentioned by Reynolds and it could be an especially useful feature for games, for example a number of enemy aircrafts following their commander. However, as Moeslinger et al. (2009) point out, animal flocks in nature do not seem to have a leader. Thus a simulation of a real flock would have to be designed without such a rule.

It seems intuitive (if not necessary) to use the embodied animats approach in simulations. Not only it is easier to apply physics to the units and make the behaviour more realistic (Reynolds, 1987; Franceschini at. al., 1992; Bourg and Seeman, 2004), but also such simulations can provide a starting point for building real robots. The simulation of Moeslinger et al. (2009) was specifically designed to accommodate restrictions of real robots like simple infrared vision.
It is however difficult to simulate what will happen in real situations because of many parameters of real environments and unexpected situations that can happen. It can be rather tricky to find the right level of simplification. Even Moeslinger et al. (2009) mention that their simulation is simplified and does not take into account image recognition, low processing power or a very restricted memory of today's small robots. Although their algorithm works inside of a computer, there are potentially many problems with it if it is to be applied to real robots. Other issues apart from vision should probably be considered as well, for example irregularities of a terrain, moving obstacles, etc.

While simulations give programmers insight into how flocking works in theory, robotic engineering encounters problems like vision and the environment around. Although it seems like we understand flocking as a procedure, there are very few successful attempts to apply our knowledge to real-life robots. The following sections will discuss different types of sensors and how useful they could be for flocking before looking at the currently available robots.

3. Robot Vision and Control

Classical AI believed in the perception-reasoning-action concept where each of the three was a separated module (Mackworth, 1992). Data flew from one end to another, input produced output through algorithmic evaluation. As Brooks (1991) and later Mackworth (1992) pointed out, this approach carried many difficulties. Robots had to have reliable perception mechanisms and needed long time to process inputs and compute actions. To perceive, they had to have a camera and build a complete 3D model of their environment before they were able to interact with the world. Because robot vision never got too far, they were only able to deal with simplified 'block worlds'. If we started building such robots, for example even only to give us a tour through a zoo, engineers would have to concentrate on complicated vision, even though all robot would have to do would be to navigate through a known environment, stop at some places and play a record about a particular animal. This would be an expensive design in terms of material and energy (Pfeifer R., 1996).

Gibson (1998) showed that much less information than a complete 3D model is necessary not only to navigate but also to interact with objects in the real world. He introduced the idea of an optic flow and how comparison of 2D projections of the environment on retina can be used to compute one's own movement relative to the environment. Furthermore, one is able to extract data such as objects outlines from the 2D image and decide whether to interact with an object or not, even whilst standing at the same place. Gibson also proposed how animals and robots could distinguish between solid and liquid surfaces and the air based on different characteristics of objects' boundaries and the way they reflect light.

There seem to be the following four rules applied to visually-guided modern robots (the 'new' AI robots) throughout multiple papers:

1. **Simple vision** - usually an ordinary camera with a standard resolution is necessary (Horswill and Brooks, 1988). 3D world is transformed into 2D images.

2. **There is an image processing mechanism**, which is able to use the extracted 2D images to compute optic flow of the robot (Franceschini et. al., 1992) and to distinguish between objects (Horswill and Brooks, 1988)

3. **Parallel computation and simple reasoning**- many engineers tend to use Brooks' subsumption architecture or some hybrids of it. The systems usually have modules with their own inputs and outputs that work together to produce a resultant behaviour. Horswill and Brooks (1988) proved that modules are able to correct each other's errors and therefore do not have to produce perfect outputs. This is a definitive advantage over the

classical AI where a sophisticated 3D model of the world and reasoning about it were required.

The input-output communication can sometimes be based purely on ‘reflexes’ and use a very simple neural network. However, this is usually the case only for very simple robots like Webb’s Cricket Robot (1996). Many authors argue that purely reflex-based behaviour is not enough to perform sophisticated actions like classifying objects (Horswill and Brooks, 1988; Arbib and Liaw, 1995) or playing football (Mackworth, 1992). In order to demonstrate flocking, there certainly has to be a mechanism that could classify objects and distinguish between other robots, other objects, walls and the ground.

4. Visual feedback - robots are able to evaluate their actions through visual feedback. The environment and the position of a robot are changed through robot’s actions. This change is a part of the input in the next time step (Horswill and Brooks, 1988), thus the fitness of the actions can be evaluated. This is what Gibson (1998) referred to as ‘visual kinesthesis’. His most interesting remarks showed how one could find out whether one was to collide with an object, what is an object’s movement direction relative to one’s position or how one could aim towards an objects, all by comparing the object’s flow through the visual field. Utilising vision and the environment this way saves computation time in comparison with classical AI models and provides reactive real-time behaviour.

Some of the first people who showed the advantages of parallel computation, embodiment and 2D vision were Horswill and Brooks (1988). They built a robot using the subsumption architecture that was able to chase objects in real time. The robot used optic flow of an object to decide how to steer and when to accelerate based on rules similar to Gibson's (1998). It had a number of modules that processed 2D images from a camera and fed the processed data into further modules in the chain. The robot first chose an ‘interesting’ moving blob of pixels by comparing current and past camera input. The blob was then segmented from the rest of the image and the movement controlled so that the object stayed in the centre of the view field. There was a memory module which helped focus on the object if it disappeared for a short period of time (e.g. by imprecise steering).
The robot was surprisingly good in distinguishing between objects and floor or walls by evaluating movement of pixels on its ‘retina’. It was even able to detect objects of a similar colour than the floor. The real-time behaviour was a great step forward, although as Horswill and Brooks pointed out, the robot sometimes interpreted snapshots from the camera incorrectly and its movement was misled.

In 1992 Franceschini and his colleagues built a robot with compound-eye vision based on principles of fly navigation. The movement control was analogue and used EMD’s (Elementary Motion Detectors) around its body. The detectors used analogue signals to control motors and a neural network between them worked as a nervous-system-like obstacle avoidance algorithm. However, the robot could only see during movement and it was constrained to move on a flat floor and among obstacles that were in high contrast with the environment.

Nevertheless, the nature-inspired real-time control system was an important step in AI and robotics.

It is apparent that the main problem with visually controlled motion is extracting relevant detail of information from the camera input. However, if data is understood correctly, vision provides a powerful feedback to robot's action as it directly reflects the environment. Snapshots from a camera contain all existing objects around a robot and can provide information about terrain, character of obstacles or direction of movement.

4. Other Sensor Types

There are other sensors apart from cameras available for today's robots. Webb (1996) built her 'Cricket robot' to show how a robot could be guided by a sound. She wanted to simulate the way real crickets approach mates by listening to their sound and steering towards them. The robot had sound sensory neurons connected in a neural network that controlled the motors. The sensors fired and provided data to the output (control) neurons when they were excited enough. The robot was able to react to a specific sound and even get to it around obstacles. Some behavioural characteristics of a real cricket emerged, including a zigzag path towards a goal or the ability to choose a correct one from two sounds. Webb reminded us that the nature does not necessarily work with vision only and that control can be achieved by other means as long as a robot 'understands' other signals.

Bats are another example from the nature where vision is not the only mechanism for navigation. Because they usually hunt at night, they use ultra sound that bounces off obstacles in order to find free space. Such navigation could be an alternative to Gibson's distinguishing between substances based on light reflection. A similar approach can be taken in robotics by using infra red light. However, a lot of testing and calibration is usually required before sensible values can be obtained (Society of Robots). As Moeslinger and his colleagues (1992) proposed, infra red (IR) sensors could be used for flocking as well.

There are two main components necessary for IR object detection: emitter and detector (Seattle Robotics). Two emitters, one on each side, allow a robot to see objects on the right, left and front. In the obstacle checking algorithm one of the emitters is turned on and response from the detector is checked. Then the other emitter is turned on and response is checked again. The Seattle Robotics web site (Seattle Robotics) provides an algorithm that returns a certain value for each relative position of an obstacle.
It is also necessary to be able to filter detector's input in order to get rid of unwanted noise. For example the sunlight can provide misleading data. Therefore, the detector needs to be shielded (Society of Robots). Another form of filtering is light modulation, i.e. letting the emitters give off light at a certain frequency recognized by the detector (Seattle Robotics).

While simple IR components can be built from other electronic devices (Seattle Robotics) at home, there are also ready-made components available to buy. People at ROBOTmaker offer

IRCFs (Infrared Control Freak) sensory module (ROBOTmaker 1) that is able to detect objects and their distances using a 180-degree IR proximity sensor and light sensors. It can also convert IR signals to serial data that can be used for communication with other robots and human control. (ROBOTmaker 2) claims that IRCF can be used for robots that follow or hide from light, implement flocking or simulate insect behaviour.
Other web sites including (Active Robots), (Robot Shop), (Hobby Engineering) and others also provide parts for IR robotics.

IR sensors seem to be fairly easy and cheap to implement and more popular nowadays than camera vision, especially for 'garage engineering'. Even though IR can be used for obstacle avoidance and path following, the issues like distinguishing between fellow robots and other objects and coping with a complicated terrain remain. The next section will look at some of the techniques used in robots that demonstrate collective behaviour.

5. Robots with Collective Behaviour

Mackworth (1992) proposed Constrained Nets as architecture for robots that would be aware of themselves and others. CNs use real-time symbolic reasoning able to cope with a specific task. Mackworth's project DYNAMO is interested in building robots with cooperative behaviour, specifically playing of football (Dynamo Project). However, CNs require internal model of a robot and its environment and most of the reasoning is pre-programmed, i.e. done by designer not by the robots. Also, robots of the Dynamo project ('Dynamites') are radio-controlled and use an off-line control system. This makes them uninteresting for flocking, since flocking robots would have to be autonomous and able to demonstrate their behaviour anywhere. It would probably be interesting though to bring the symbolic reasoning into the robots themselves as it seems like a potentially good idea for tackling of complex tasks.

A fairly advanced robot available to buy is the e-puck robot. It has sound sensors, IR proximity sensors, a colour camera and an accelerometer able to sense the robot's own movement (E-puck).
The results of cooperation and communication of e-puck robots seem promising. A video on (YouTube: e-puck 1) shows how a number of robots can synchronise to face the same direction. The robots send IR signals about their orientation to each other. Each robot adjusts its own orientation slightly according to its neighbours until all units are aligned. Different formations of robots lead to a successful result over different times. For example a circle or a square formation requires more time than a simple row formation. Number of neighbours seems to be the determining factor, although the time in which the result is achieved is always acceptable. As it was mentioned in section 2, this behaviour is necessary for flocking. (YouTube: e-puck 2) shows a similar behaviour, with four robots following each other. Although e-puck robots are rather impressive, they are clearly only able to move on a plain surface with no obstacles. The measurement of angle correction and accelerometer work well, but 3D environment (water, air or even an uneven surface) would require more complex computation and could bring yet unnoticed problems.

Similar robots called S-bots have been developed for the Swarm bots project (Swarm bots) that aims to build 'self-organising and self-assembling artefacts'. The robots are given a task and cooperate to fulfil it. They have more sensors than the e-puck robots, which help them deal with a complicated terrain (Swarm bots 2). They communicate via IR, coloured light and sound signals (Swarm bots 3). The interesting thing about the S-bots is that they can form physical connections to establish a larger aggregate. Their representation of other robots is dynamic and can be switched from single to aggregate in real-time. A group of S-bots can deal with obstacles and holes bigger than a single unit and a 'swarm' can divide work and complete multiple tasks at the same time. (Swarm bots 3)
The Swarmanoid project (Swarmanoid) uses extended S-bots able to deal with 3D environment and object manipulation. Individual units are able to form a robust creature which could be used for underwater or space exploration and work. Unfortunately the project is still experimental and no real-life results have been achieved yet.

6. What Could Be Done

S-bots seem to be the most capable of collective behaviour including flocking. They use a circular array of IR sensors around their body to see (Swarm bots 2). However, as the web site mentions, this has been proven sufficient only for simulations. Also, as (Society of Robots) and (Seattle Robotics) argue, IR sensors can be misled by other light sources which makes them rather difficult to calibrate in natural environments.
On the other hand, camera vision can provide a more reliable source of input data. However, there are some problems related to camera vision including:

- **Visual field** – a robot needs to be aware of objects around it when it moves. While sensors around the whole body can be used for IR objects detection, implementing a 360-degrees camera could be expensive.

- **Filtering of information and data representation** – camera input provides a complete snapshot of the environment around. A robot needs to distinguish relevant data and recognize what it means.

To overcome the visual field problem, one could implement a compound eye similar to the one Franceschini and his colleagues (1992) implemented. There could be a digital neural network which would process data from a number of cameras around a robot's body.

An effective approach could be to implement three cameras with N-degrees visual field. The main camera in the front would be the most expensive one and have the best resolution. Modules which would process its input would be responsible for segmentation of 2D image. The approach would be similar to Horswill's and Brooks' (1988). One of the problems with their model was that objects used to disappear from the field of view. If additional two lower-resolution (and therefore cheaper) cameras were added to each side, a robot could track an object within a much wider field.

This approach would utilise peripheral vision, i.e. extension of the visual field with lower resolution on the sides. Lower resolution would be sufficient because an object (objects) would already be segmented, i.e. the outlines and colour would already be recognized and tractable by lower resolution cameras. If a moving or interesting object would be detected, the robot's head would turn towards it so that the high resolution camera could help recognize it, similarly how animals and humans do it (Gibson J.J. 1998). A robot could continue tracking of the object afterwards, with its head turned towards another object.

In comparison with a 360 degree field of view, there would be no information from the back of a robot. This information is often unnecessary, unless a robot moves backwards. In such cases its head could be simply rotated towards its movement direction.

The IR or coloured light communication could be used to detect other robots (E-puck, Swarm bots 3). If an object gives off a certain signal, it is a robot. IR/ colour signalling or robot shape recognition could be used to indentify blobs in the visual field which represent other robots. Shape recognition would be a harder task to do if robots looked differently from different angles since there would have to be a concept of a robot in the memory (Kirsh, 1991). To some extent, a visually-controlled robot would have to recognize at least some classes of objects because of obstacle avoidance (e.g. distinguish water from solid objects). It would perhaps be worth experimenting with IR identification and dynamic representation of a 'robot' class. Robots could be trained to recognize others and form 'pattern schemata' of others (Arbib M. A. and Liaw J., 1995).

Nevertheless, detection of other robots would be necessary for applying the rules of flocking. For example to apply the separation rule, a robot would have to be able to measure distance to others. This could be calculated by the motion parallax principle (Franceschini et. al., 1992), i.e. by evaluating how fast an object of a known size moves through the visual field (Arbib M. A. and Liaw J., 1995).

Although vision could extend perception and thus capabilities of robots, experiments in realistic environments would have to show whether any of the above mentioned approaches would have an advantage against simple IR recognition. Robots would probably need more computational power and it is a question whether added possibilities and behaviours would compensate for a possible need of better processors.

7. Conclusion

This paper compared the available robot sensors including vision, infrared, colour sensors and sound and discussed how some of them could be used for flocking. Any of these sensors can be used for robot control, however IR sensors and vision seem to provide the most relevant data for flocking and are mostly used. While IR is much simpler and easier to implement, it lacks some advantages like wide visual field or object classification. Even though the latter is

possible by using 2D vision, there are no algorithms known to the author that demonstrate objects recognition at a level higher than obstacle - not at obstacle.
There are a few examples from today's robotic research which show how using more sensor types like colour detectors, IR sensors and accelerometers can work together and help achieve autonomous navigation and collective behaviour. Some robots are already able to move together and even cooperate on a given task, although most of the work is still experimental. To demonstrate flocking fully, one would have to not only implement an efficient algorithm for coordinated movement in experimental conditions but also cope with constraints of real-life environments. Cheaper and more accessible robot parts would certainly allow wider public to experiment with the existing algorithms and new implementation ideas and move the field forward.

References

Active Robots [online] http://www.active-robots.com/products/sensors/sensors-sharp.shtml [accessed 01 January 2010]

Arbib M. A. and Liaw J. (1995) ‘Sensorimotor Transformations in the Worlds of Frogs and Robots’. *Artificial Intelligence* 72:53-79

Bourg D. M and Seeman G. (2004) *AI for Game Developers*. 1st ed., Sebastopol, O' Reilly Media

Brooks R.A. (1991) ‘Intelligence without reason’. *Computers and Thought,* IJCAI-91, April 1991

Brooks R.A. and Flynn A.M. (1989) ‘Fast, Cheap and Out of Control: a Robot Invasion to the Solar System’. *Journal of the British Interplanetary Society*, 42:478-485

Dynamo Project [online] http://satirist.org/learn-game/projects/soccer/dynamo.html [accessed 01 January 2010]

E-puck [online] http://www.e-puck.org/ [accessed 01 January 2010]

Franceschini N., Pichon J.M., Blanes C. (1992) ‘From Insect Vision to Robot Vision’. *Philosophical Transactions: Biological Sciences*, 337:283-294

Gamershell: Flock [online] Available on: http://www.gamershell.com/tv/17844.html [accessed 03 December 2009]

Gibson J.J. (1998) ‘Visually Controlled Locomotion and Visual Orientation in Animals’. *Ecological Psychology*, 10:3:161-176

Hobby Engineering [online] http://www.hobbyengineering.com/SectionS.html#CatSIR [accessed 01 January 2010]

Horswill I. D. and Brooks R. A. (1988) ‘Situated Vision in a Dynamic World: Chasing Objects’. *AAAI-88*, 796-800

Kirsh D. (1991) ‘Today the Earwig, Tomorrow Man’. *Artificial Intelligence* 47:161-184

Levy S. (1992) *Artificial Life*. London, Penguin Group

Mackworth A. K. (1992) ‘On seeing robots’. *Computer Vision: Systems, Theory and Applications*, Singapore, World Scientific Press, pp. 1-13.

MgTaylor [online] Available on: http://www.mgtaylor.com/mgtaylor/jotm/spring97/flock.htmz [accessed 03 December 2009]

Moeslinger Ch., Schmickl T., Crailsheim K. (2009) ‘A Minimalist Flocking Algorithm for Swarm Robots’. *Proceedings of the European Conference on Artificial Life* (ECAL 09), Springer LNCS (in press)

Pfeifer R. (1996) ‘Building “Fungus Eaters”: Design Principles of Autonomous Agents’. *From Animals to Animats. Proceedings of the 4th International Conference on Simulation of Adaptive Behaviour.* Cambridge, Massachusetts, The MIT Press

Reynolds C.W. (1987) ‘Flocks, Herds, and Schools’. *Computer Graphics* 21(4), p 25–34
Robot Shop [online] http://www.robotshop.ca/infrared-light-sensors.html [accessed 06 January 2010]

ROBOTmaker 1 [online] http://www.robotmaker.co.uk/ [accessed 17 Dec 2009]

ROBOTmaker 2 [online] http://www.robotmaker.co.uk/IRCF/IRCF_benefits_Features.htm [accessed 17 Dec 2009]

Seattle Robotics [online] http://www.seattlerobotics.org/guide/infrared.html [accessed 06 January 2010]

Seeley T.D., Camazine S., Sneyd J.D. (1991) ‘Collective Decision Making in Honey Bees: How Colonies Choose Among Nectar Sources’. *Behavioural Ecology and Sociobiology* 28:277-290

Society of Robots [online] http://www.societyofrobots.com/schematics_infraredemitdet.shtml [accessed 06 January 2010]

Swarm bots 1 [online] http://www.swarm-bots.org/ [accessed 06 January 2010]

Swarm bots 2 [online] http://www.swarm-bots.org/index.php?main=3&sub=33 [accessed 06 January 2010]

Swarm bots 3 [online] http://www.swarm-bots.org/index.php?main=3&sub=31&conpage=sbot [accessed 06 January 2010]

Swarm bots 4 [online] http://www.swarm-bots.org/index.php?main=3&sub=35 [accessed 06 January 2010]

Swarmanoid [online] http://www.swarmanoid.org/project_description.php [accessed 06 January 2010]

YouTube: Weapon Choice [online] http://www.youtube.com/watch?v=-siT8boc9BU [accessed 03 December 2009]

YouTube: e-puck 1 [online] http://www.youtube.com/watch?v=ngGDHwUJxpQ [accessed 18 Dec 2009]

YouTube: e-puck 2 [online] http://www.youtube.com/watch?v=6MBjPfgyKvM [accessed 18 Dec 2009]

Webb B. (1996) ‘A Cricket Robot’. *Scientific American*, December 1996, 62-67