

Algorithms for Building Annular Structures with Minimalist Robots Inspired by Brood Sorting in Ant Colonies

Matt Wilson*, Chris Melhuish*, Ana B. Sendova-Franks†*, Samuel Scholes*

* - Intelligent Autonomous Systems Laboratory

† - School of Mathematical Sciences

Faculty of Computing, Engineering and Mathematical Sciences, University of the West of
England, Bristol, U.K.

matthew.wilson@uwe.ac.uk, chris.melhuish@uwe.ac.uk, ana.sendova-franks@uwe.ac.uk,
samuel2.scholes@uwe.ac.uk

Abstract

This study shows that a task as complicated as multi-object ‘ant-like annular sorting’ can be accomplished with ‘minimalist’ solutions employing simple mechanisms and minimal hardware. It provides an alternative to ‘patch sorting’ for multi-object sorting. Three different mechanisms, based on hypotheses about the behaviour of *Leptothorax* ants are investigated and comparisons are made. Mechanism I employs a simple clustering algorithm, with objects of different sizes. The mechanism explores the idea that it is the size difference of the object that promotes segregation. Mechanism II is an extension to our earlier two-object segregation mechanism. We test the ability of this mechanism to segregate an increased number of object types. Mechanism III uses a combined leaky integrator, which allows a greater segregation of object types while retaining the compactness of the structure. Its performance is improved by optimizing the mechanism’s parameters using a genetic algorithm. We compare the three mechanisms in terms of sorting performance. Comparisons between the results of these sorting mechanisms and the behaviour of ants should facilitate further insights into both biological and robotic research and make a contribution to the further development of swarm robotics.

Index terms – collective behaviour transition, genetic algorithm, ant, sorting, swarm intelligence, swarm robotics

1 Introduction

This study is inspired by the brood-sorting behaviour of *Leptothorax* ants, which have evolved to live in the narrow cracks in rocks. Due to this almost two-dimensional environment it is possible to recreate fairly natural conditions for these ants between two glass slides. This provides the opportunity for detailed observations [Franks and Sendova-Franks, 1992]. The ants sort their brood so that ‘*different brood stages are arranged in concentric rings in a single cluster around the eggs and micro-larvae*’¹, (Figure 1). Franks and Sendova-Franks [1992] speculate that these patterns are produced to influence the priority in the tending of different brood items. In general, the older and larger brood items, that need more tending, are placed in bands further from the centre of the structure. The more peripheral its position, the earlier a brood item is likely to be fed.

¹ *A similar pattern of concentric circles develops on the combs of honeybees, with ‘three distinct concentric regions – a central brood area, a surrounding rim of pollen, and a large peripheral region of honey’ [Camazine, 1991].*

Inspired by the clustering behaviour of ants [Deneubourg *et al.*, 1991], various researchers have explored the idea of sorting objects with robots employing minimal rules. Early research was conducted solely through simulations [Deneubourg *et al.*, 1991; Lumer and Faieta, 1994]. However, more recent research explores mechanisms for real robots, capable of: clustering a single object type [Beckers *et al.*, 1994]; segregating two object types [Melhuish *et al.*, 1998] and sorting multiple object types in separate patches [Melhuish *et al.*, 2001].

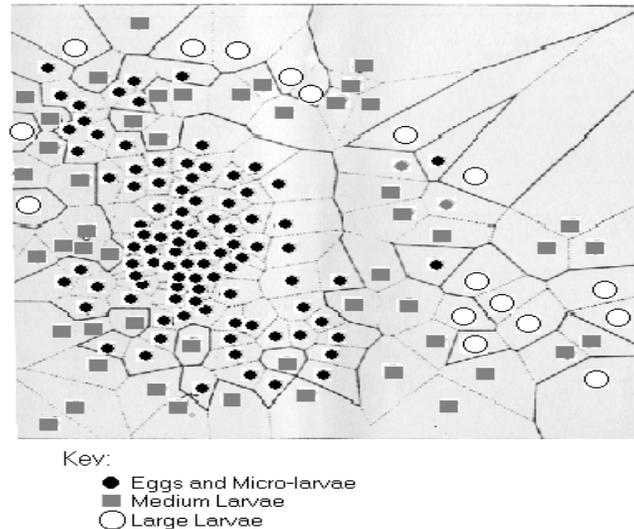


Figure 1: Illustration of a Sorted Brood Structure. *Shaded shapes have been superimposed onto a Dirichlet tessellation [Franks and Sendova-Franks, 1992], to make clear the positions of the different brood items. The figure shows a central core of eggs and micro-larvae, surrounded by a circular ring of medium larvae, with the large larvae tending to be furthest from the core. The structure is slightly distorted because of the rectangular shape of the nest, and the position of the nest entrance to the right.*

The present study explores the mechanisms that enable the sorting of any number of object types into an annular pattern, defined in Melhuish *et al.* [1998] as ‘forming a central cluster of one class of objects, and surrounding it with annular bands of the other classes, each band containing objects of only one type.’ We define a cluster as a collection of objects where no object is more than one object diameter away from any other object.

As with previous research [Beckers *et al.*, 1994; Melhuish *et al.*, 1998; Melhuish *et al.*, 2001], this study uses robots built from very simple mechanical components and sensors. Potentially, this allows for the production of more robust and cheaper robot units [Johnson and Bay, 1994]. Simple behavioural rules are employed so the robots are less complex. Rules have to be embodied and realized in a machine, and the more complicated the rules, the more complicated the hardware and software in the machine is likely to be. The more complicated the hardware and software required, the more there is to go wrong. By using less hardware and software the mean time between failures is reduced. If two rule sets generate the same behaviour it is better to choose the smaller rule set.

The mechanisms by which *Leptothorax* ants create brood structures are still not understood, although several hypotheses have been presented. One possibility is that brood sort structures form simply because objects are of different sizes. A similar effect occurs in muesli where the smaller particles percolate to the bottom of the packet [Barker and Grimson, 1990]. Another possibility is that ants deliberately introduce spacing between brood items. The amount of spacing is influenced by the size of the brood item, detected by the amount of waste gas the brood item produces. A spacing mechanism has been explored using real robots [Melhuish *et al.*, 1998] and produced one central circle with a concentric ring around the outside. However, Melhuish *et al.* [1998] tested this mechanism with only two object types of the same size.

In the study that follows these hypotheses are explored further in both a simulated and a real robot environment, where pucks (Frisbees™), are the objects used.

2 Performance Metric

In robotics, a structure resembling the sorted brood pattern in *Leptothorax* has been termed an annular pattern. To judge the quality of annular patterns produced, and to make comparisons between them, we devised a performance metric to quantify the degree of ‘annularity’ detectable in multi-object structures. The metric comprises four components. Each component compares a different characteristic of a given annular structure to the same characteristic in an ideal structure.

2.1 - Ideal Structure

We propose an ideal annular structure based on the definition of annular structure by Melhuish *et al.* [1998] where objects are separated according to type. The centre of the structure is formed by a compact circular cluster of one type of object, surrounded by perfect circular bands; each band formed from a different object type. Object types should be completely segregated so that no object lies in a band dominated by another type of object and the whole structure should be spread over the minimum possible area (Figure 2).

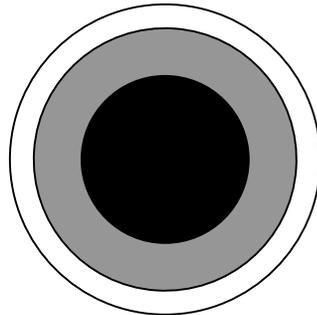


Figure 2: Illustration of an Ideal Structure. *It consists of a large number of small objects. An equal number of each object type results in all the bands having the same area as the central cluster.*

2.2 - Best Achievable Structure

The best achievable structure differs from the ideal structure, as it depends on the number and size of objects available for use. In the present study the objects used are circular pucks of the same size. The number and shape of the objects creates new issues and imposes limitations on possible structures that can be created. Figure 3 shows three attempts at forming the best achievable structure, and illustrates how obtaining complete and optimal circular bands may be mutually exclusive with achieving optimum compactness. For example, in Figure 3 (top left) full separation has been achieved, but compactness has not been optimized. In Figure 3 (top right) more of the available space has been utilized but it could be argued that the white objects impinge on the region occupied by the grey objects. Figure 3 (bottom) shows how the objects can be further compacted, but the shape of the structure and the separation of the object types have been further compromised.

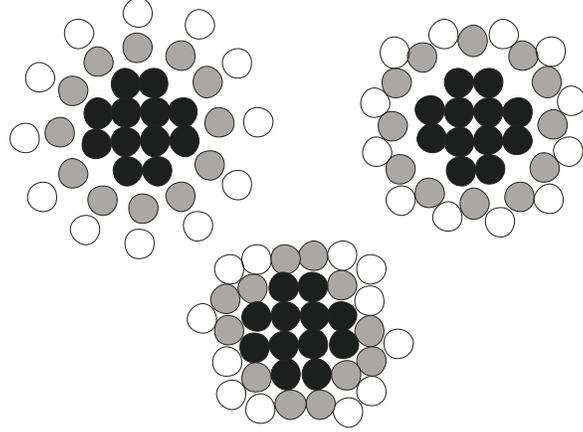


Figure 3: Three Different Structures Comprising 36 Objects, 12 of each Shade. a) *Top left – Optimum separation, with good shape and completeness.* b) *Top right – Compromises separation for compactness.* c) *Bottom – Excellent compactness.*

2.3 - Annular Sort Performance Metric

We postulate that an annular sort is a structure defined by:

- Separation
- Compactness
- Shape
- Completeness

We describe each of these components and present formulae for their calculation. For practical purposes, we judge 100% performance in a metric component as reaching the ideal for that component, and accept that achieving 100% performance simultaneously with all components is unachievable. The position of each individual object is taken at its centre; with its size unimportant. The centre of the whole structure is taken to be at the centroid of the largest cluster of the central object type.

The formulae in sections 2.3.1 to 2.3.4 make use of the following variables:

- r = object radius;
- l = arena side length;
- k = number of objects in the largest cluster of the central object type;
- m = number of types of object;
- c = type of object;
- n_c = number of objects of type c , $c = 1, \dots, m$;
- t = total number of objects;
- $x_{c,i}$ = radial distance to the centre of the structure for object i of type c , $i = 1, \dots, n_c$;
- \bar{x}_c = $\sum_{i=1}^{n_c} x_{c,i} / n_c$; mean distance to centre of the structure for objects of type c ;
- \bar{x} = $\sum_{c=1}^{c=m} \bar{x}_c / m$; mean distance to centre of the structure;
- q_c = the lower quartile of distances $x_{c,i}$ for type c ;
- p_c = the upper quartile of distances $x_{c,i}$ for type c ;
- $\theta_{c,i}$ = angle subtended between neighbouring objects of type c and the centre of the structure taken in a clockwise order.

2.3.1 - Separation

The separation component of our performance metric is computed by calculating the radial distance from the centre of the structure for each object. Sorting these distances according to type and then ordering them provides a lower quartile and an upper quartile of the distribution of distances for each object type. The distance between an object type's lower and upper quartile is taken to be that type's 'home zone'.

Our approach, to the calculation of this component of our performance metric, tests every object to see if its position infringes on the 'home zone' of another object type.

The following calculation then provides the separation component quantity:

1. For the central type objects: Count the number of objects, C_{glq}^1 , which have a radial distance to the centre greater than the lower quartile range of any other type.
2. For the outer-most type objects: Count the number of objects, O_{luq}^m , which have a radial distance less than the upper quartile of any other type.
3. For the intermediate type objects: For each of the intermediate object types, determine the number of intermediate objects I_{glq}^c which have a radial distance greater than the lower quartile of any other object type from the centre.
4. For the intermediate type objects: For each of the intermediate object types, determine the number of intermediate objects I_{luq}^c which have a radial distance greater than the lower quartile of any other object type from the centre.
5. Sum the count from part 1 and 2 with half the count from part 3 and 4. Compute this as a fraction of the total number of pucks available. Express as a percentage where 100% denotes full separation. Thus separation is expressed as:

$$100 * (1 - ((C_{glq}^1 + O_{luq}^m + (\sum_{c=2}^{c=m-1} (I_{glq}^c + I_{luq}^c) / 2)) / \sum_{c=1}^{c=m} n_c))$$

Figure 4 shows two extreme examples of separation: poor separation. Figure 4(a) represents a 27% separation component. In comparison Figure 4(b) illustrates good separation with a 100% separation component.

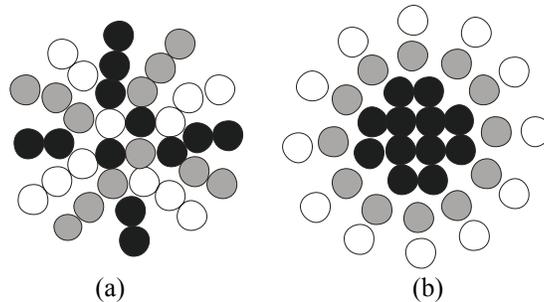


Figure 4: Extreme Separation and Shape Examples. (a) *Poor Separation (27%), Poor Shape (22%).* (b) *Good Separation (100%), Good Shape (93%).*

2.3.2 - Compactness

The ideal annular structure should be perfectly compact so that it fits into the smallest possible circular area. The problem of packing a number of identical circles into the smallest possible circular area is a well-known geometrical problem for which exact solutions have been found for up to twenty circles [e.g. Kravitz, 1967; Goldberg 1971; Reis, 1975]. For packing more than 20 identical smaller circles various iterative computer algorithms have been developed on the basis of the idealised movement of billiard balls inside a circular table [Lubachevsky and Graham 1997]. Here we use Donovan's algorithm¹ to estimate the minimum radius of the packing circle. This allows an estimate to be made of the best achievable mean radial distance to the centre of the structure. Figure 5 shows the packing achieved by Donovan's algorithm for 36 circles.

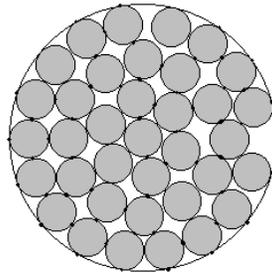


Figure 5: Circles in a Circle: Packing Arrangement for 36 Circles. *Using Donovan's algorithm.*

The computation of the compactness component involves the following steps.

- Firstly, we calculate the mean radial distance \bar{x} to the centre of the structure for the total number of objects.
- We express the compactness as a percentage of the differences between this mean radial distance and the best radial distance, calculated with Donovan's algorithm, and the radial distance for a 'poor' structure. This limiting distance is set at 1.5 arena side lengths and represents the largest mean distance observed during trial experiments.

$$\text{Therefore Compactness} = 100 * (1 - (\bar{x} - \text{optD}(t)) / (1.5 - \text{optD}(t)))$$

where $\text{optD}(t)$ simply represents the optimum mean Euclidean distance to the centre of the structure for the total number of objects when they are packed using Donovan's algorithm.

2.3.3 - Shape

The ideal annular structure requires a compact circular central cluster of objects of the same type. This central structure should be surrounded by bands of objects of other types, each band forming a perfect circle.

The shape component of the performance metric consists of two sub-components: a cluster performance component for the central object type, and a measure of deviation from a common radius for the other object types.

The clustering performance component is calculated by finding the size of the largest cluster, as defined by Melhuish *et al.* [1998], of the central object type (type 1) and converting this to a percentage of the total number of available objects of this type.

Each of the other object types should form circular bands around the central type, and each object of a particular type should lie on the same radius as every other object of that type. Considering each object type

¹ *The algorithm employs a form of 'billiard balls' packing described at www.packomania.com.*

in turn, a common radius is calculated by taking the mean radial distance from the centre of the structure. The sum of the absolute deviation from this radius is then the sum of the Euclidean distances to each of the objects from this radius. A percentage is calculated by placing this sum deviation between zero and one common radius, which also has a normalizing effect. 100% represents a perfect band shape.

Taking the mean, by summing the clustering percentage sub-component with each of the band shape percentage sub-components, provides the shape component of the performance metric. The metric is therefore defined as:

Shape Metric = (Cluster Percentage + Sum of performances for each band)/ number of object types

$$= (100k/n_1 + \sum_{c=2}^{c=m} (100 * (1 - ((\sum_{i=1}^{i=n_c} |x_{c,i} - \bar{x}_c|) / (n_c * \bar{x}_c)))))/m;$$

where the contribution of the central cluster made from type 1 objects is $100k/n_c$ and the contribution from each of the bands surrounding the central cluster is:

$$100 * (1 - ((\sum_{i=1}^{i=n_c} |x_{c,i} - \bar{x}_c|) / (n_c * \bar{x}_c)))$$

Figure 4 shows two extreme examples of shape: a poor shape with a shape component of 22% is shown in Figure 4(a) while figure 4(b) illustrates a good shape with a shape component of 93%.

2.3.4 - Completeness

The completeness component of the performance metric aims to determine the degree to which objects are evenly spread within a circular structure. This is a statistical problem of circular data analysis [Fisher, 1993].

Figure 6 illustrates the meaning of completeness. Both structures contain 12 objects, and each structure would be assigned a 100% value for the shape component of the metric because the objects in each structure lie on a common radius. However, in contrast to figure 6(a), the objects in 6(b) are uniformly distributed around the radius and therefore this circle is said to be more complete.

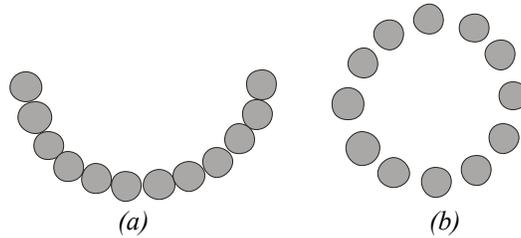


Figure 6: Extreme Completeness Examples. a) *poor completeness (12.5%).* b) *good completeness (100%)*

To judge completeness we are interested in the spread of points around the circle. Spread statistics, such as variance and standard deviation, make use of the mean in their calculation. The traditional mean is not well 'defined' in the case of circular data. It would not make sense for the circular average of 0.2π and 1.8π to be $0.5(0.2\pi + 1.8\pi) = \pi$. Therefore, Rao [1969] introduced a test for uniformity, based on equal spacing.

Under uniformity, $E[\theta_{c,i}] = 2\pi/n_c$. For this study, we adapt Rao's test. In a perfect annular sort, the angles between pairs of neighbouring objects, taken from the centre of the structure, should be identical for all pairs of objects. This perfect angle is calculated by dividing 2π radians by the number of objects. Each

object type is taken in turn and the completeness formula calculates the absolute difference of the angle between two neighbouring objects and this perfect angle value. The completeness percentage for an object type is the average of these angular differences expressed as an angular percentage between zero and the angle representing the worst possible case of 2π when there is only one object. A mean of all object types is taken to provide the completeness metric. This metric is limited to the spatial arrangement in a 2D environment. For example, vertical stacking of objects, providing no angular difference between objects is not allowed.

$$\begin{aligned} \text{Completeness for type } c \text{ is therefore: } & \left(\sum_{i=1}^{i=n_c} |\theta_{c,i} - 2\pi/n_c| \right) \\ \text{Completeness percentage for type } c = & 100 * (1 - \text{Sum of deviations} / \text{worst possible sum}) \\ & = 100 * (1 - (\sum_{i=1}^{i=n_c} |\theta_{c,i} - 2\pi/n_c|) / 2\pi) \\ \text{Completeness metric expressed for all types objects (where } & c = 1, \dots, m) \\ & = 100 * (\sum_{c=1}^{c=m} (1 - (\sum_{i=1}^{i=n_c} |\theta_{c,i} - 2\pi/n_c|) / 2\pi) / m) \end{aligned}$$

Figure 6 shows two extreme examples of completeness: In Figure 6(a) the completeness component is 12.5% representing poor completion. Figure 6(b), in contrast, shows good completeness with a completeness component of 100%.

2.4 – Representing the Metric

The components of the metric, namely; separation, compactness, shape and completeness are brought together in the ‘performance diagram’ illustrated in figure 7. Such diagrams are used throughout this paper to illustrate the aspects of the quality of an annular sort. In the simulated experiments, the values in the diagrams represent averages over 50 runs. The centre of the structure is taken to be the centre of the biggest cluster of black pucks (the central object type). On the left, the radial displacement plots show the distribution of distance from the centre of the structure for each object type; the smallest and largest radial displacement and the inter-quartile range represented by the box. The diagram on the right represents all the components of the performance metric on a bar chart with error bars. Standard errors tend to be low. Errors below 0.5% are not visible on the bar chart.

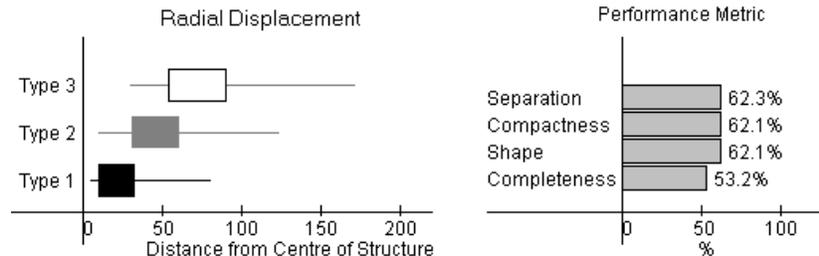


Figure 7: Example of a Performance Diagram. Left – the radial displacement range plot showing maximum and minimum values represented by the line and the interquartile range represented by the box. Right – the components of the performance metric

2.5 - Judging the Quality of Annular Sorting

In the remaining part of this study we employ the above performance metrics to judge and compare the amount of ‘annular sorting’ existent in a given structure of objects. To declare that a mechanism shows evidence of ‘annular sorting’ it must be able to produce structures, which on average have more ‘annularity’ than is evident in the initial distribution of the pucks, and more ‘annularity’ than is evident in

structures created by a simple clustering algorithm. Therefore, for comparative purposes, experiments were conducted with different numbers of object types employing variants of Melhuish's version [Melhuish, 1999] of Becker's clustering algorithm [Beckers *et al.*, 1994]. The full rule set for this algorithm, referred to as the canonical three rule algorithm by Melhuish [1999] is:

Rule 1:

If (gripper pressed and obstacle ahead) then

Make random turn away from obstacle

Rule 2:

If (gripper pressed and no obstacle ahead) then

Reverse a small set distance which causes the puck to be dropped

Make random turn left or right

Rule 3:

If (gripper not pressed) then

Go forward

It should be noted that 'gripper pressed and obstacle ahead' implies that the robot has hit a wall or another robot and gripper pressed and no obstacle ahead means the robot is carrying a puck and has hit another puck. Different puck types are not distinguished by the algorithm.

2.6 - Primary and Secondary Performance Metric Components

Of the four components of the metric, namely; separation, compactness, shape and completeness, we argue that the separation and compactness components of the performance metric are the most useful in the study of sorting mechanisms presented in this paper. This is because shape and completeness components of the performance metric occur naturally – simply as a result of clustering. The robots disturb centrally-placed clusters evenly from all sides and this tends to lead to a fairly circular shape. Because robots deposit pucks evenly around the cluster, completeness also tends to happen naturally. Separation and compactness are more directly affected by the parameters of each of the mechanisms and are therefore considered primary performance metric components, whereas shape and completeness are considered as secondary performance components.

3 Annular Sort Mechanisms

The results of three different mechanisms are presented in this section:

Mechanism I : Object Clustering using Objects of Different Size

Mechanism II : Extended Differential Pullback

Mechanism III : Combined Leaky Integrator

The implementation of all three mechanisms requires robots that are able to move, if unimpeded, in approximately straight lines in an arena. They have sensors that enable them to detect walls and other robots (but not discriminate between them), and have the ability to 'carry' one puck at a time. The robots drop pucks next to other pucks with which they collide, or drop pucks at a distance from other pucks, by first pulling back before releasing the held puck. The inside surface of the puck can be white, black or employ rings of black and white.

Using an infrared sensor the robots are able to detect the reflectivity of the puck, and hence can discriminate between the type of the object they are carrying in the scoop. The robots we use for the present

study are known as Ubots [Melhuish *et al.*, 1998]. In the Ubot simulation the objects are moved by blind bulldozing agents, with the simulation designed to extract the salient features from the physical system. These features include: modelling robot-robot interactions, robot-environment interactions, and includes noise in the detection of the puck types, at the same rate as the mistakes made by the infrared sensors of the real robots (approximately 1%). The simulation also models friction in the environment, with disturbance of pucks inversely proportional to their mass. Puck-puck and puck-wall interaction take into account the size of the pucks.

The experiments carried out, both simulated and real, involved six robots and 15 pucks of each type. The simulated experiments were run for 500,000 iterations, the equivalent of 4 hours of real time. Longer runs were tried, but this did not improve performance as a stable structure had already been created. An example of the evolution of the four components of the performance metric over time in a simulated experiment is shown in Figure 8. Fifty runs of each simulated experiment were conducted and results in the performance diagrams represent the average of the 50 runs.

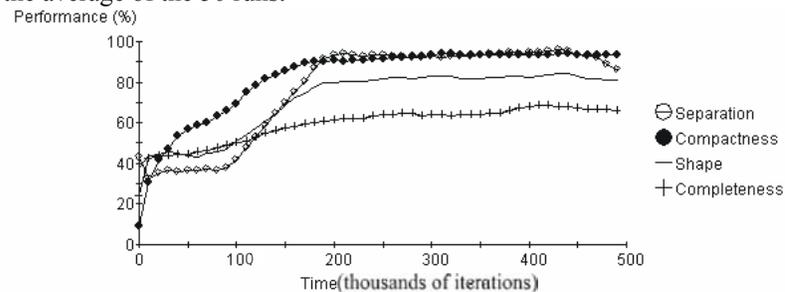


Figure 8: Typical Evolution of the 4 Components of the Performance Metric over Time in the Simulated Experiments. This example is taken from one run of the extended differential pullback mechanism using three different object types.

3.1 - Mechanism I: Object Clustering using Objects of Different Size

3.1.1 - Method

This sorting mechanism is based on the hypothesis that the brood of *Leptothorax* ants is organised in concentric annuli as a result of the size difference of the brood items. In their article ‘The Physics of Muesli’, Barker and Grimson [1990] discuss the properties of particulate solids. The central theme of the article is the idea that under gravity or a centripetal force, particles separate according to their size. As an illustration, they consider a packet of muesli where the largest particles rise to the top under repeated shaking. There are two simultaneous mechanisms at work here. Together they create the following effects:

- Small particles filter down through the network of pores created by the packing of the larger particles, under the force of gravity.
- The shaking process allows small particles to move into voids below the larger particles, preventing these from returning to their previous lower position.

This section explores whether clustering of objects of different sizes creates an annular structure. This idea was first explored in simulation by McCoy [1991]. However, the simulation she used did not model any attributes of a realistic physical environment and results are sketchy. Our simulation experiments of the Ubot in the present study provide a better environment than McCoy’s simulation to test whether object size contributes to object segregation. We do not use real robots here because the scoop of the Ubot is made for pushing and pulling pucks of a particular size. Using objects of a different size would require a major change in hardware design.

3.1.2 - Simulation Results

The aim of these experiments was to find out whether it is possible, using the canonical clustering algorithm described above, to create segregation within object types of different size. The first object types used are pucks of standard size, proportionally the same as those used in the real robot arena. The second, third and fourth object type are pucks that are, respectively, two, three or four times the diameter of the standard size puck. Figure 9 gives a snapshot of the arena after a typical run of 500,000 iterations, with object type 4 shown as crosses for clarity, while Figure 10 shows the performance metric diagrams based on the average results of 50 runs.

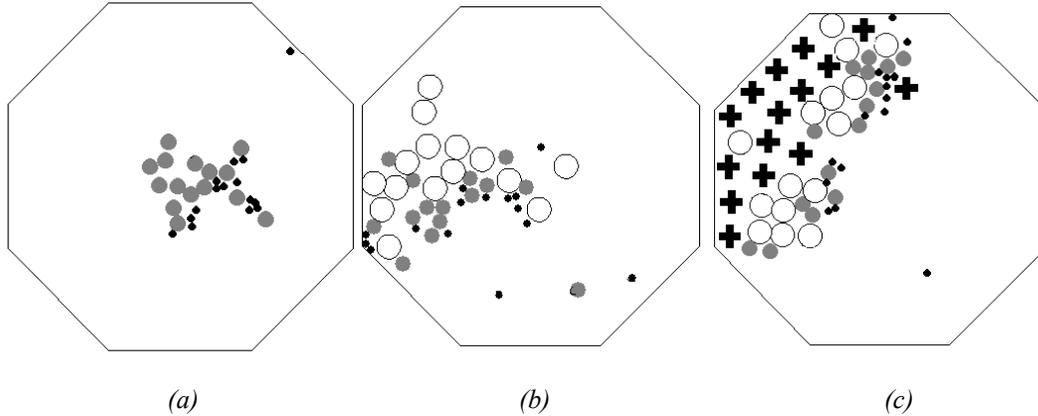
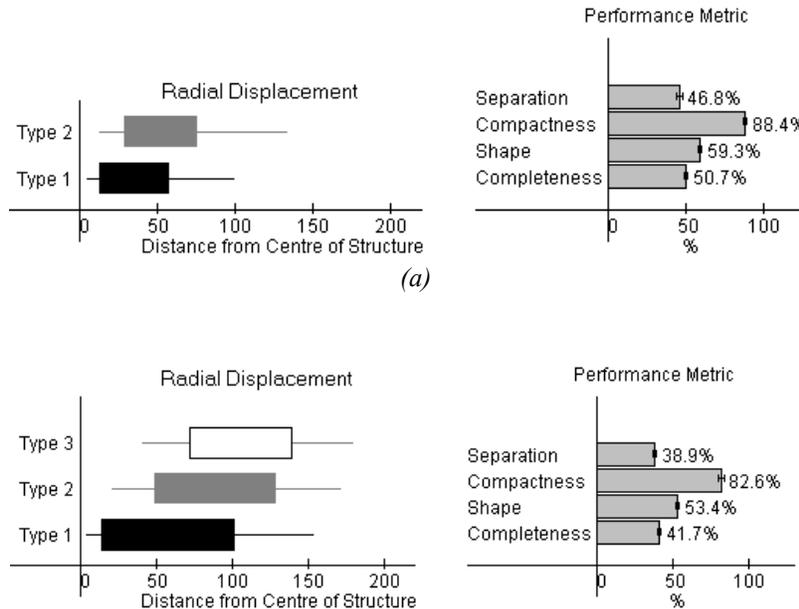


Figure 9: Examples of the Simulated Arena after 500,000 Iterations. (a) Two object types. (b) Three object types. (c) Four object types.

Separation for the experiments is shown on both the radial displacement diagrams and in the separation component of the performance metric in Figure 10. The average separation component for two colours, using different sized objects was 46.8% (Figure 10(a)) in comparison to 30.2% for same object size clustering; for three colours it was 38.9% (Figure 10(b)) compared to 26.6% for same size clustering and for four colours it was 34.9% (Figure 10(c)) compared to 22.9%. The results show there is a slight improvement in separation when the canonical-three rule algorithm is employed using different sized objects, rather than objects of the same size.



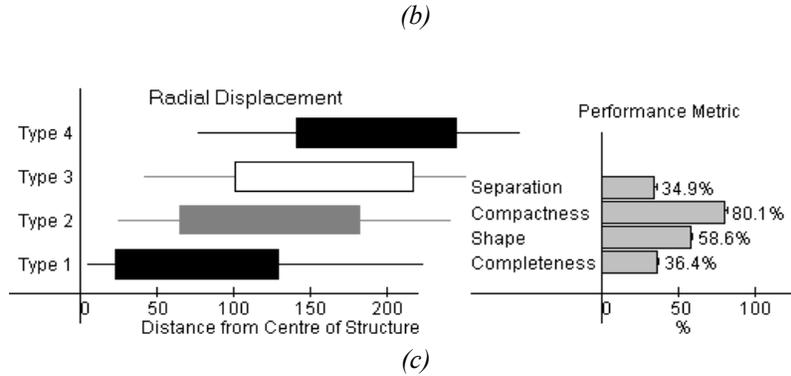


Figure 10: The Performance Metric Diagram. The calculations are based on the average of 50 runs, stopped after 500,000 iterations. (a) Two object types. (b) Three object types. (c) Four object types.

3.2 Mechanism II: Extended Differential Pullback

3.2.1 – Method

In section 3.1 we have shown that applying the canonical three-rule clustering algorithm [Melhuish, 1999] to object types of different size, does not create sufficient ‘muesli effect’ to produce a good annular structure within a cluster of objects. This section explores an alternative mechanism. Observations of *Leptothorax* ants’ nests have led to a new hypothesis on how *Leptothorax* ants sort their brood. This hypothesis is based on two assumptions. The first assumption is that brood items have a constant metabolic rate, which means that smaller items produce fewer waste products than larger items. The second assumption is that the ants have the sensory apparatus to detect the volume of metabolic by-product produced by a brood item. This allows the ants to estimate the size of the brood items.

An animal’s sensory apparatus tends to work within a range between a lowest and highest detection threshold. Below the lowest threshold the stimulus will not trigger an action potential, while above the highest threshold, additional stimuli will not result in any further effects on the nervous system. Within limits most animals get around this problem by recalibrating sensors when a high concentration of stimulant is present. However, this creates a new problem; the sensor operating range can shift toward higher stimulant concentrations. It is possible that *Leptothorax* ants attempt to modify the environment to keep the metabolic by-product sensors within an optimum range. This means that small brood items are clustered together to produce the same volume of gas as large brood items that are spaced out. Spacing according to size allows by-product regulation, a consequence of which is the familiar pattern of concentric annuli.

Prior to the present study, the best attempt for basic robots to create an annular sorted structure within a collection of different object types was implemented by Melhuish *et al.* [1998]. The segregation mechanism extends the canonical three-rule algorithm [Melhuish, 1999] to allow object spacing according to type. The first type of object (red¹ pucks) is dropped next to other pucks without spaces introduced. However, the second type of object (yellow pucks) is pulled back from other pucks, before being dropped. This mechanism allows a physical implementation of the *Leptothorax* brood-sorting hypothesis introduced earlier. In this way the red pucks represent eggs which only produce small volumes of waste products, while the yellow pucks could represent larvae producing more waste products, and therefore spaced away from other brood items. For practical purposes the objects used are not of different sizes. However, this should not be important because the underlying idea of the above brood sorting hypothesis relies on spacing and not size. When tested on real robots and later in simulation the pullback algorithm produced what was described as ‘a central cluster of red pucks surrounded by a halo of yellow pucks’ [Melhuish *et al.*, 1998]; a description equivalent to a two colour annular structure. It was therefore reasoned that it might

¹ Although the terms red and yellow are used for convenience, these refer to the puck rims. The robots discriminate between puck types by the greyscale markings on the puck surfaces.

be possible to create annular structures with more than two object types simply by using different pullback distances for the different types of puck. Puck types, which should end up nearer the periphery of the annular structure, are subjected to greater pullback distances than those types that end up nearer the centre. The rules for this extended pullback mechanism are:

Rule 1: *If (gripper pressed and obstacle ahead) then*
 Make a random turn away from the obstacle and drop the puck

Rule 2: *If (gripper pressed and no obstacle ahead) {*
 If (Type 1 puck carried)
 Drop puck
 else
 Pull back the distance according to puck type and drop the puck

Rule 3: *If (gripper not pressed)*
 Go forward

3.2.3 - Results

The extended differential pullback mechanism was tested with both real robots and in simulation, with the pullback distances set to be proportional to the object type:

Type 1: 0.
Type 2: 20 Units = 2.5 puck diameters.
Type 3: 40 Units = 5 puck diameters.
Type 4: 60 Units = 7.5 puck diameters.
Type 5: 80 Units = 10 puck diameters.

The values of the pullback distances were chosen by a process of trial and error using the simulation, with the aim to obtain a balance between compactness and separation.

3.2.3.1 - Simulation Results

The first experiment explores differential pullback with two object types, which is similar to the segregation experiments of Melhuish [1999]. The mechanism is then extended to three, four and five different object types (Figure 11).

Figure 12 shows a reasonably good separation is achieved for two and three different object types, with the separation at a reduced level for four and five different types. For greater number of types, the radial displacement diagrams show that the outer object types are less well separated than the inner ones. When the experiments were conducted in a bigger sized arena, a greater degree of separation was achieved. This suggests the arena is not large enough to accommodate the pullback distances necessary to produce good separation of more than three object types using this mechanism.

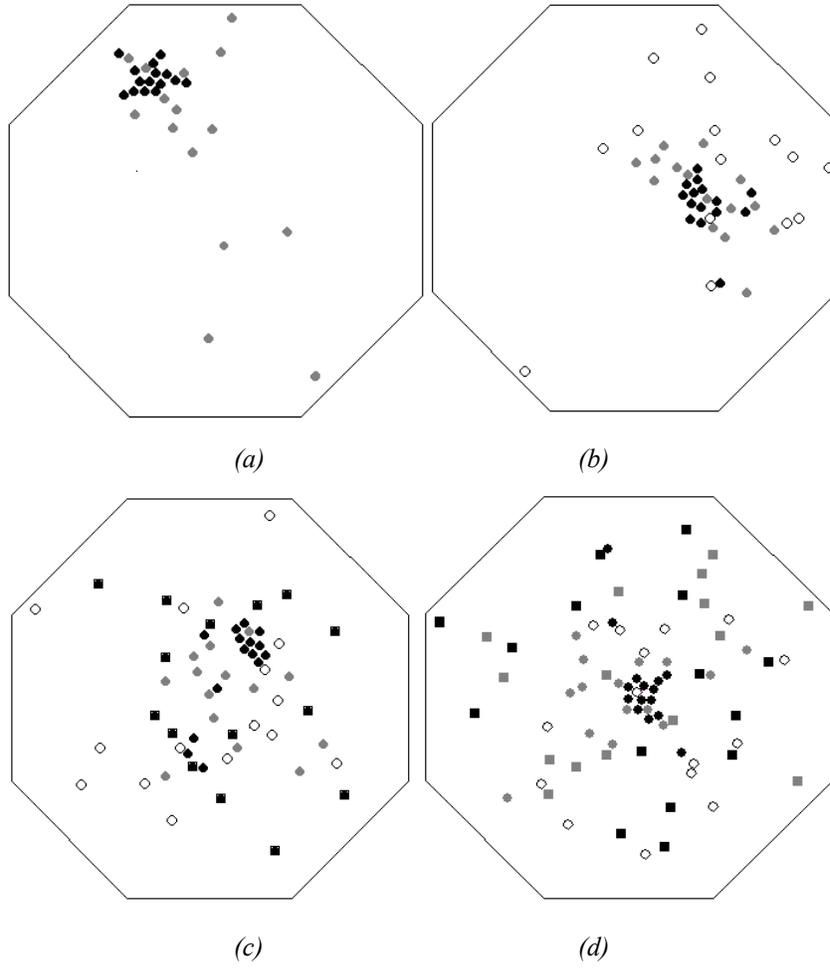
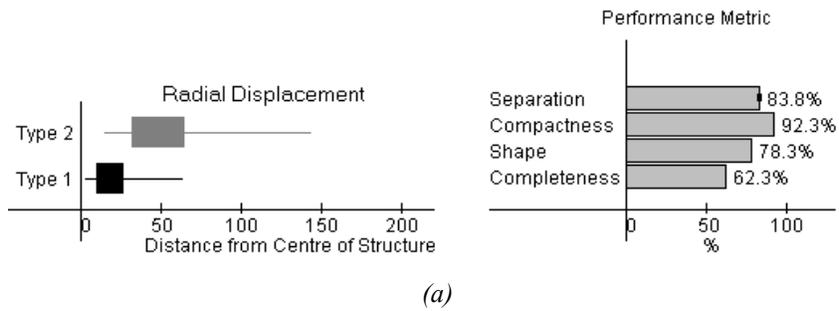
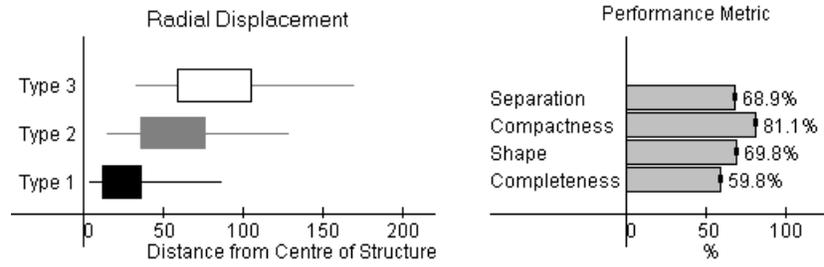
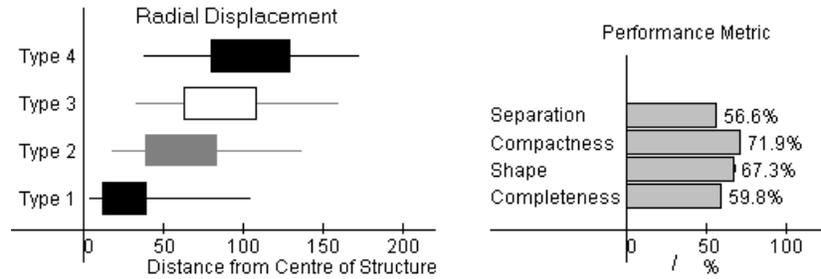


Figure 11: Examples of the Simulated Arena after 500,000 Iterations. *a) Two object types. (b) Three object types. (c) Four object types. (d) Five object types.*

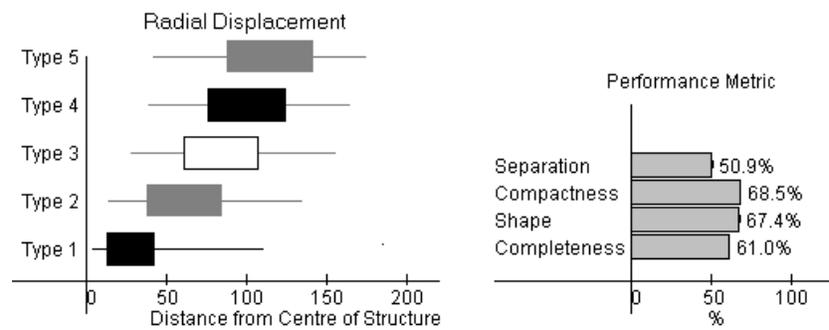




(b)



(c)



(d)

Figure 12: The Performance Metric Diagram. The calculations are based on the average of 50 runs and stopped after 500,000 iterations. (a) Two object types. (b) Three object types. (c) Four object types. (d) Five object types.

3.2.3.2 - Real Robot Results

To verify the results achieved using the extended pullback mechanism in simulation, the mechanism was tested using real robots in the physical arena with three types of puck. Three runs of the real robot experiment were conducted. Figure 13(a) shows the layout of the arena at the start of each of the runs and Figure 13(b), (c) and (d) show the arena after 4 hours of running time in three different trials. By comparing Figure 11(b) with the photographs in Figures 13(b), (c), (d), it can be seen that the real robots produce comparable results to simulated robots. Analyzing the structures in these photos using the performance metric allows the performance diagram in Figure 14 to be produced. It is comparable to figure 12(b). Although the average performance metric values taken from the real robot experiments are slightly different from those taken from the simulated robots (see figure 12(b)), the difference is not considerable, given that only three runs were conducted with the real robots.

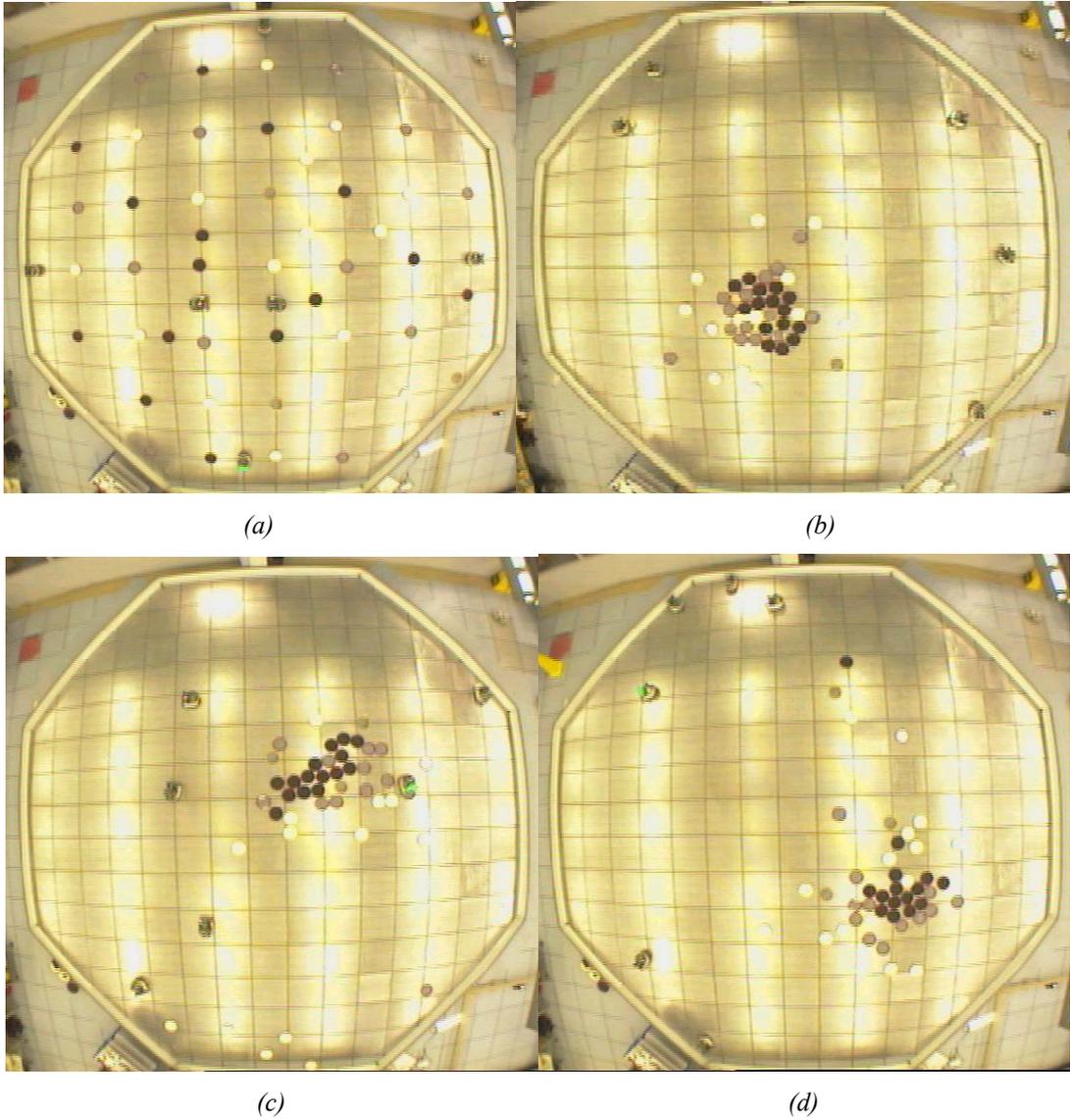


Figure 13: Real Robot Experiment. *a) Start of Run. b) Run 1: After 4 hours. c) Run 2: After 4 hours. d) Run 3: After 4 hours.*

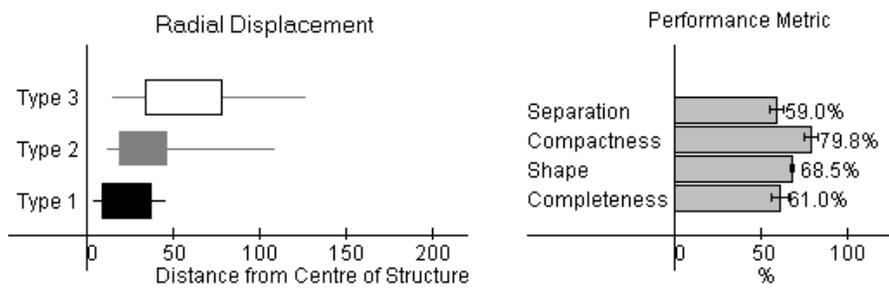


Figure 14: The Performance Metric Diagram. *The calculations are based on the average of 3 runs of the real robots, after 4 hours.*

3.3 - Mechanism III: Combined Leaky Integrator

3.3.1 - Method

Using the differential pullback mechanism with a large number of object types results in a lack of compactness (Figures 11 and 12). The combined leaky integrator mechanism is an attempt to remedy this. The differential pullback mechanism with its fixed pullback distance was able to segregate objects efficiently and it was reasoned that these larger pullback distances were effective during initial stages of the sorting process, but once the objects were well sorted then smaller pullback distances could compact the structure. To implement this idea the combined leaky integrator mechanism is introduced. Degrading integrators rapidly ‘forget’ and only recent information is taken into account. The integrator provides information about the dispersal of each object type. Thus object types not recently seen are likely to be clustered, whereas object types that are currently seen are probably still dispersed throughout the arena.

Describing the mechanism more completely, each individual robot holds a separate counter (leaky integrator) for each type of object. These counters are analogous to containers for liquid. Initially the containers are half full. When a robot drops a puck and the container associated with the dropped puck type is not full, an amount of liquid is added to that container. To avoid ‘saturation’ the volume of liquid in each container slowly drains away.

The combined leaky integrator mechanism operates in a similar way to the extended differential pullback mechanism described in Section 3.2, but pullback distances are adaptive and based on the volume of liquid in the integrators. When using three puck types, the type 1 (black) pucks are not pulled back; the type 2 (grey) pucks are pulled back a distance based on the sum total of the liquid in the grey and the black integrators; and the type 3 (white) pucks are pulled back a distance based on the sum total of the liquid in the black, grey and white integrators. Figure 15 illustrates how pullback distance relates to the quantity of liquid in each of the integrators.

The complete rule set for the combined leaky integrators mechanism is given below:

- Rule 1:** *If (gripper pressed and obstacle ahead) then*
Make a random turn away from the obstacle
- Rule 2:** *If (gripper pressed and no obstacle ahead)*
If (Type 1 puck carried)
Drop Puck and add 15 units to type 1 integrator
If (Puck type f is carried)
Reverse distance proportional to the sum of integrators 1 to f
Drop puck and add 15 units to type f integrator
- Rule 3:** *If (gripper not pressed)*
Go forward
- Rule 4:** *If (time counter reaches threshold)*
Deduct 1 unit from all integrators and reset time counter

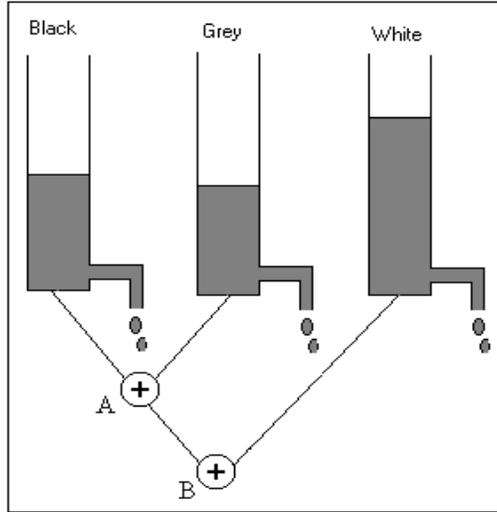


Figure 15: Illustration of the Leaky Integrator Mechanism.

$$\text{Pullback for grey} = (\rho_b h_b + \rho_g h_g) g$$

$$\text{Pullback for white} = (\rho_b h_b + \rho_g h_g + \rho_w h_w) g,$$

where ρ_b - density of liquid for black; h_b - height of liquid for black; g = gravity.

For these experiments the values of ρ_b and g for each integrator are set to 1. The pullback distance for the grey pucks is proportional to the sum of the pressures at the bottom of the black and grey containers ($\rho_b h_b + \rho_g h_g$), shown at A (Figure 15), and the pullback distance for the white pucks are proportional to the sum of the pressures at the bottom of the black, grey and white containers ($\rho_b h_b + \rho_g h_g + \rho_w h_w$), shown at B (Figure 15).

For three puck types, the combined leaky integrator mechanism works as follows:

At the start of a run, all the pucks are uniformly dispersed and the robots are equally likely to see each type of puck. There will be a tendency for all three integrators to be full and the pullback distances for the grey and white pucks will be large. Pulling back these puck types, allows a central cluster of black pucks to easily form.

As the black cluster forms, the black pucks are less accessible and the frequency with which robots carry black pucks decreases. Now there is a tendency for less liquid to be present in the black integrator, reducing the pullback distance for grey pucks. This reduced pullback distance draws the grey pucks closer to the central black cluster and forms the first annular band around the central cluster. The reduction in the amount of liquid in the black integrator reduces the pullback distance for the white pucks too but to a much lesser extent than for the grey pucks.

With the black and grey pucks both less accessible because they are now in a single cluster, there tends to be less liquid in their associated integrators and the pullback distance for white pucks tends to reduce even more. These pucks are therefore drawn closer to the structure, creating the second annular band.

Using the combined leaky integrator for two different object types has no advantage over simple differential pullback. The first experiment with the leaky integrator mechanism was therefore conducted in simulation using three different types of puck and hence each robot has three leaky integrators, represented

by counters. In the original experiment, all three counters initially started at 100 units. Every time a puck was dropped the appropriate counter increased by 15 units and every 150 time iterations (equivalent to the robot moving approximately half way across the arena) each counter decreased by one unit. The counter value was not allowed to exceed 200. An incremental value of 15 was chosen by trial and error. It tends to keep fluid in each integrator while rarely forcing an integrator to reach saturation.

The results of these experiments did not show an improved performance in terms of either compactness or separation over the differential pullback mechanism. This is because the parameter values were set using intuition and a rough trial and error approach. The pullback distances were calculated by combining the selected integrators in equal proportions. Equality was arbitrarily chosen and is not necessarily the best combination of integrators. In addition, weighting the contribution made by each integrator to the pullback distance could improve the performance of the mechanism. It is also likely that the values for the maximum integrator sizes along with the incremental values are not optimal.

The number of different parameters required for a given number of object types m is $m(m+1)$. To find a good combination of parameter values, a GA (genetic algorithm) [De Jong, 1975; Holland 1975] was employed in combination with the simulation. Each set of parameters represented a different individual whose fitness function was calculated by running the simulation using the individual's set of parameters. The parameters represented were translated from 8 bit binary into real numbers. The maximum integrator size was allowed to vary between 1 and 1000 and the incremental value to vary between 1 and 100. Leaky integrators were then combined by allowing a proportion of each integrator to contribute to an object type's pullback distance. The GA parameters were set as follows:

- Number of Individuals = 50
- Selection Method: Tournament
- No. of fitness tests per Individual = 2
- Elitism = 0.1
- Mutation Rate = 0.001
- Crossover Rate = 0.8

The fitness function used in these experiments was the separation component of the performance metric. Each individual was tested twice against the simulation with the mean value taken as the fitness. The time allowed, number of robots and number of pucks were as in the previous experiments. Because 100 runs of the simulation were required per generation each of the following experiments took a 1.4 GHz Athlon computer several weeks to complete, and, due to time constraints, only 2 tests per individual were conducted.

3.3.2 - Simulation Results

Figure 16(a) shows a graph of the mean and best separation during the evolution of a solution for three different object types. After about 20 generations, the GA had found good solutions and no improvement in fitness seemed to be occurring. At this point the difference between the mean and the best performance may simply be due to the random fluctuation in separation performance between trials.

Following evolution, the best individual was tested for 50 runs of the simulation. The results are presented in the performance diagram at Figure 16(c). The results show the new evolved set of parameters give a significant improvement in separation and compactness over both the extended differential pullback mechanism and the previous leaky integrator runs (Figure 12(b)). This demonstrates that the combined leaky integrator mechanism can be powerful when parameter values are well chosen. Following the success of the GA with three different object types, further runs were conducted with four and five different object types (Figures 17 and 18). In both cases the GA evolves parameter sets which show improvements in separation and compactness over the extended differential pullback mechanism (Figure 12(c) and (d)).

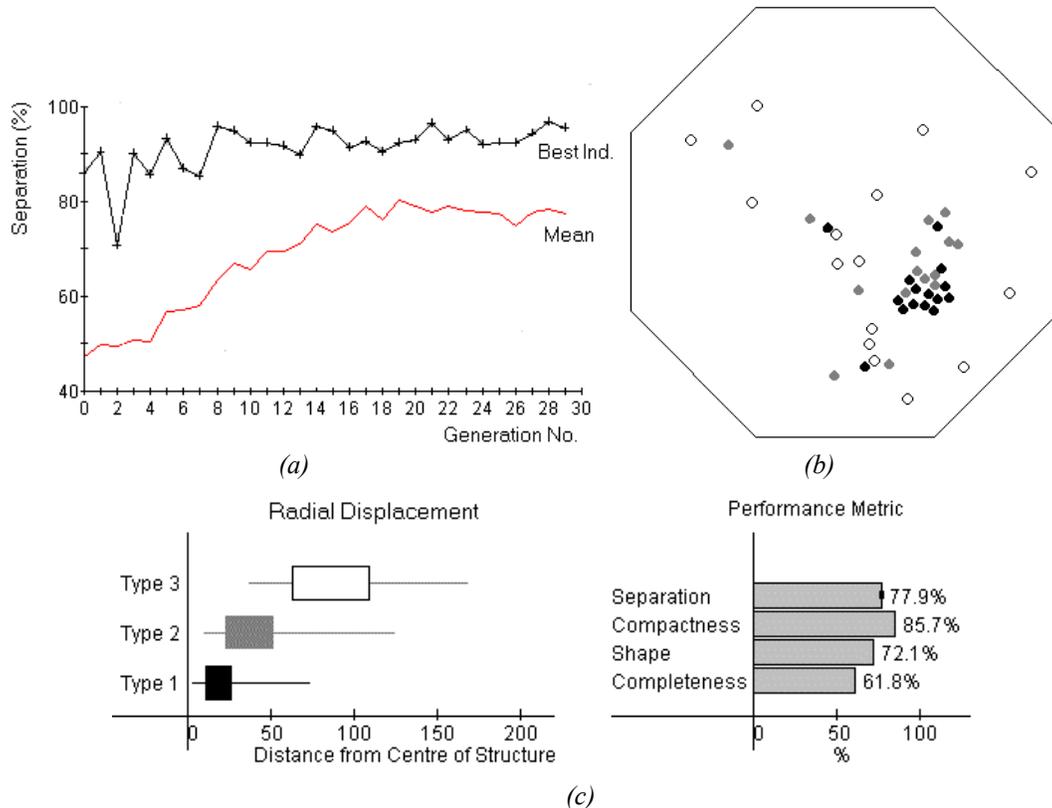
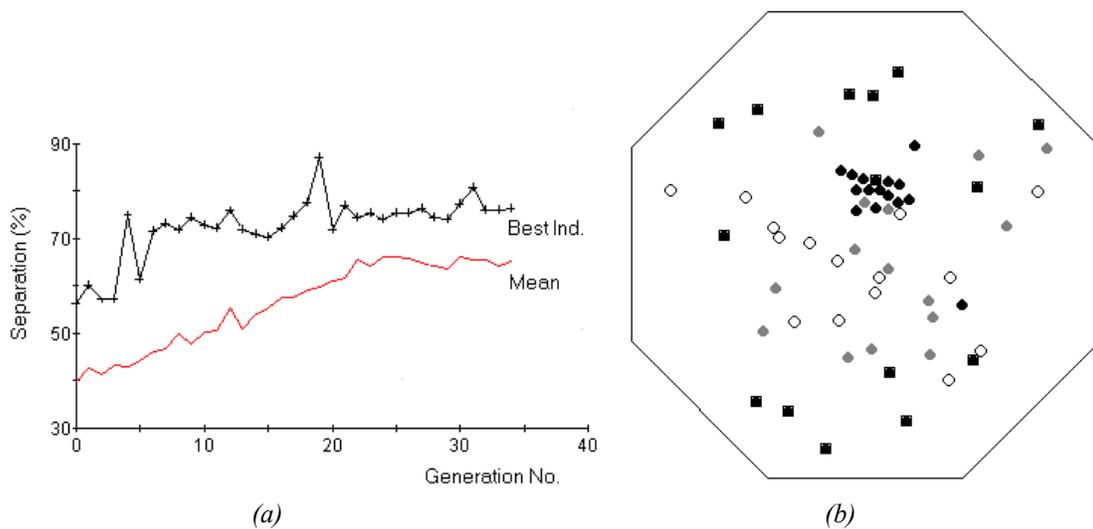


Figure 16: Results - Optimizing Combined Leaky Integrator Parameter for Three Object Types. (a) Best and average fitness during an evolutionary run for three object types, using separation as a fitness function. (b) An example of the simulated arena after 500,000 iterations testing the best individual from the evolutionary run. (c) Performance diagram for 500,000 iterations testing the best individual from the evolutionary run.



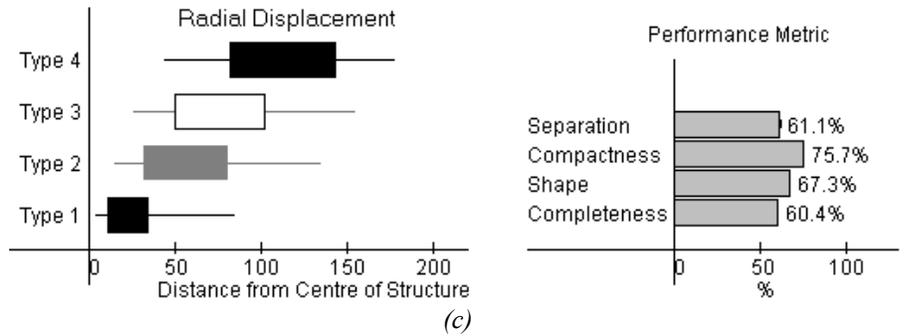


Figure 17: Results - Optimizing Combined Leaky Integrator Parameter for Four Object Types. (a) Best and Average Fitness during an evolutionary run for four different object types using separation as a fitness function. (b) An example of the simulated area after 500,000 iterations testing the best individual from the evolutionary run. (c) Performance diagram for 500,000 iterations testing the best individual from the evolutionary run.

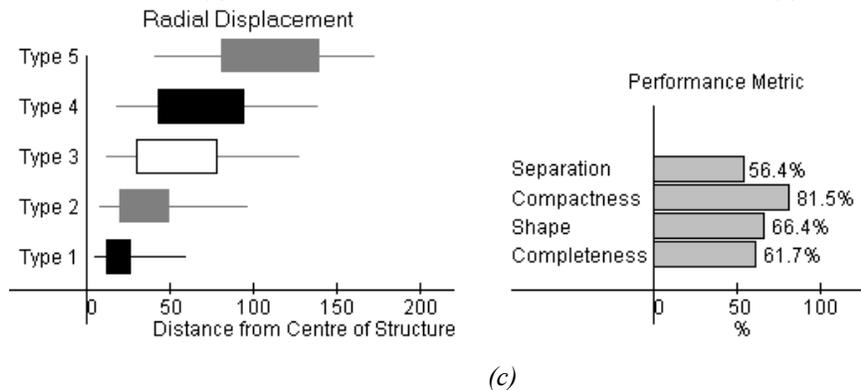
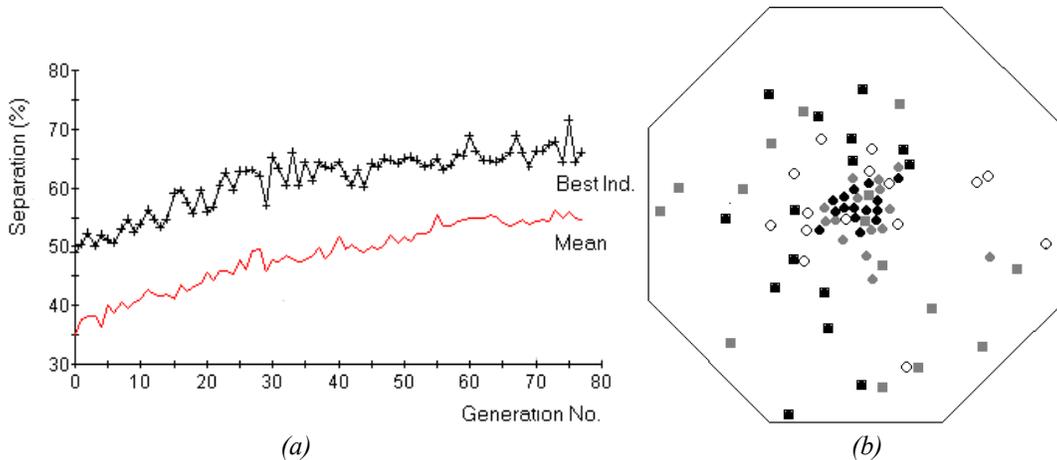


Figure 18: Results - Optimizing Combined Leaky Integrator Parameter for Five Object Types. (a) Best and Average Fitness during an evolutionary run for five object types using separation as a fitness function. (b) An example of the simulated area after 500,000 iterations testing the best individual from the evolutionary run. (c) Performance diagram for 500,000 iterations testing the best individual from the evolutionary run.

3.3.3 - Real Robot Results

To validate the simulation results, three runs of a real robot experiment were conducted with three different object types (Figure 19). Comparisons of the performance diagrams (Figure 20 and Figure 16(c)) show that real robots produce comparable results to simulated robots.

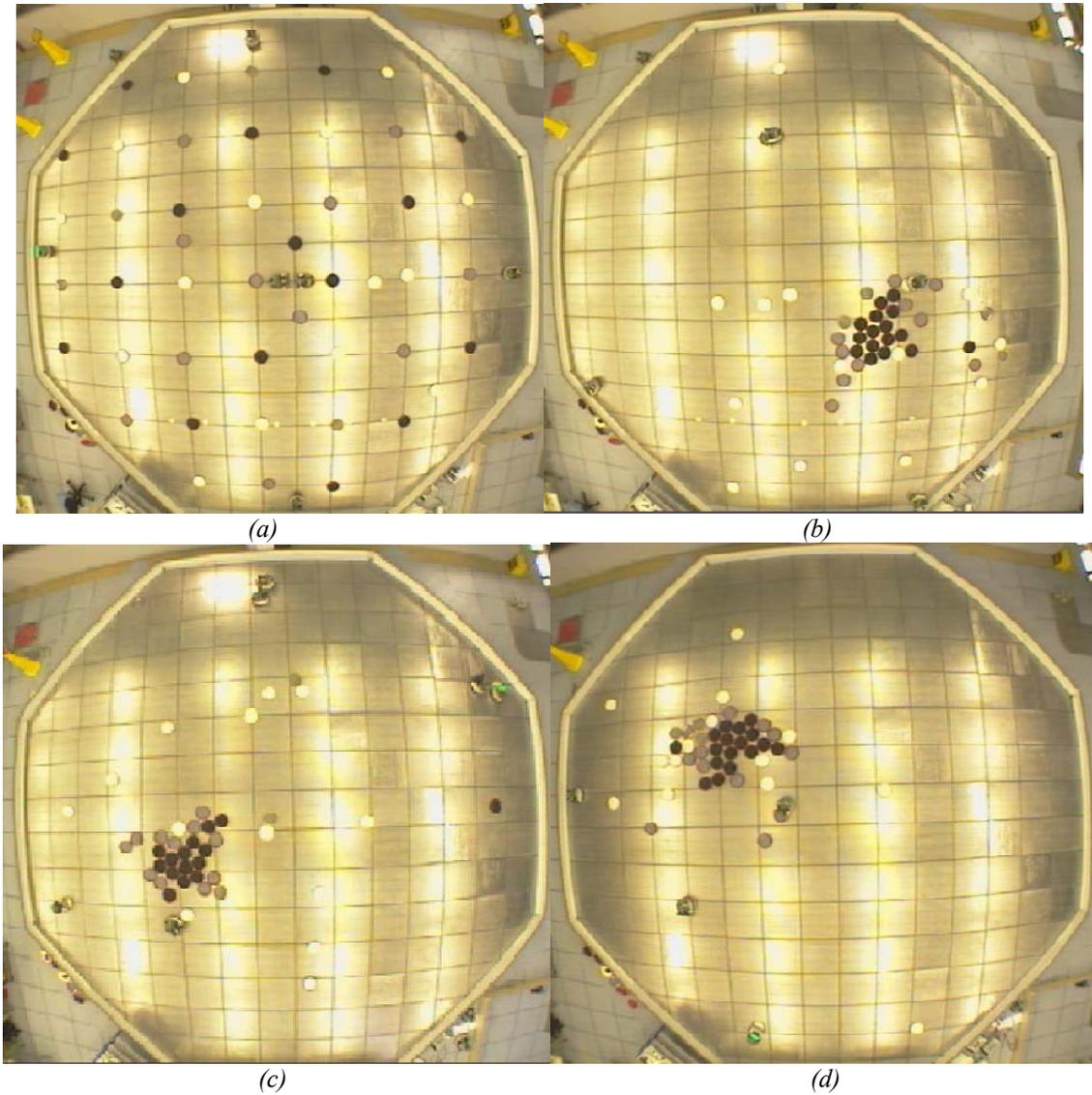


Figure 19: Real Robot Experiment: *Using three different object types. a) Start of Run. b) Run 1: After 4 hours. c) Run 2: After 4 hours. d) Run 3: After 4 hours.*

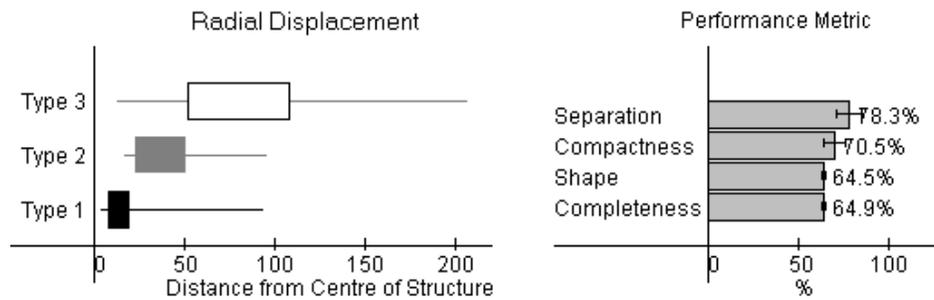


Figure 20: The Performance Metric Diagram. The calculations are based on the average of 3 runs of the real robot, after 4 hours.

4 Discussion

We have shown that a simple clustering mechanism using objects of different sizes may cause some separation of object types to occur without the need for a specific separation algorithm; the muesli effect may be at work within the cluster [Barker and Grimson, 1990]. It seems likely that it is the jostling of the cluster, under repeated collisions by the robots that facilitates this effect. There is, however, no force similar to a gravitational force, required for muesli sorting, toward the centre of the cluster. Here gravity is replaced by a clustering tendency provided by the robots, but this does not seem to provide the objects with a strong enough pull towards the centre of the cluster. This is very likely the reason why using different sizes of object types only produces weak separation of the types.

Figure 9(c) shows a cluster that has formed against the peripheral wall of the arena and here the muesli effect, in reverse, seems to have occurred, with the larger objects found nearer the wall. The explanation for this is that the objects are not moved by gravity and shaking, but physically by robots. Since the robots are more likely to collide with larger objects, they move larger objects more often. Therefore, larger objects are pushed towards the wall more often. Effectively they have a stronger wall attraction and smaller objects are squeezed out.

It may be the case that the muesli effect is in operation within the brood clusters of *Leptothorax* ants and that this is a contributing factor in the object separation taking place. However, its impact was not strong in our simulation. For engineering purposes, we should seek a better, more reliable, mechanism and one which works with objects of the same size.

Experiments conducted with the differential pullback mechanism were an attempt to find a better solution to the problem of creating an annular sort. The mechanism clusters all the objects, but provides spacing according to type. It may be that if the ants can detect the brood type they are carrying they space the objects in a similar way. It is from this spacing mechanism that annular structures emerge. It was found that this mechanism was successful at producing annular structures that are, at least, similar to those the *Leptothorax* ants create within their nests. These experiments show it is possible to produce such structures, using both simulated and real robots within a physical arena with a simple mechanism.

From a more pragmatic engineering point of view, the major problem with the extended differential pullback mechanism is that it cannot be extended to a large number of object types because the arena size limits the amount of pullback possible. The combined leaky integrator mechanism explores a means of obtaining a greater level of compactness while maintaining separation. This mechanism uses adaptive pullback distances instead of the fixed pullback distances used by the previous mechanism. Initial results with the leaky integrator mechanism were disappointing. However, this is because parameters for the mechanism were arbitrarily set. To improve the performance of the mechanism, a genetic algorithm was used in conjunction with the simulation to find a better set of parameters and produce better separation within the sorted structure. For 3, 4 and 5 different object types solutions were found, which, when tested using 50 simulated runs, not only outperformed the previous results achieved with the leaky integrator

experiments, but also outperformed results achieved using the extended differential pullback mechanism. This shows the combined leaky integrator is a more powerful mechanism.

It is interesting to note that the radial displacement diagrams show that the leaky integrator mechanism, when optimized for separation¹, seems to favour structures where the outer object type is more separated than the inner object types (Figure 16(c)). This is probably due to the fact that, while inner bands protect other object types, it is from the outer-most band of an annular structure that pucks are most often removed by passing robots. This means that while the leaky integrators of other object types tend to be empty, a ‘volume of liquid’ is usually retained in the integrator of the outer puck. It is this integrator that maintains a higher pullback for the outer object type.

The leaky integrator provides an excellent example of a self-organising mechanism. It can be described well in terms of the four underpinning characteristics proposed by Theraulaz *et al.* [1998]: both positive and negative feedback are present; fluctuations are amplified and there are multiple robot-robot and robot-puck interactions. In terms of positive feedback, the bigger the cluster of central objects, the quicker the central cluster grows. There is also negative feedback. As each type of puck is ‘seen’ less often by the robots, the reduction in the volume contained in each integrator causes the pucks to be drawn closer to the cluster. This, in turn, creates a more compact structure, and the robots pick up pucks less often, further reducing the integrators. The ‘amount of liquid’ in the integrators fluctuates; when there is less liquid present, the pucks tend to be drawn closer to the structure which, in turn, often further reduces the amount of liquid in the integrator and hence amplifies the fluctuation. The phenomenon is also an example of cue-based stigmergy [Holland and Melhuish, 1999] in that communication about the structure of the pucks occurs through the environment and influences the task-achieving behaviour of the robots.

The leaky integrator is also a mechanism that does not completely exclude the possibility of disruptive individuals. The value of each integrator for each individual is based on the number of each type of puck recently seen and although there is a likelihood that the more clustered the pucks are, the less they will be seen; this is not always the case. The next puck that a robot collects is, effectively, a random choice of those freely available. It is therefore possible that a robot’s integrator will not always provide an accurate representation of the structure of the pucks. For example, one robot which has recently seen a lot of white pucks may start to pull them away from the structure while the other five robots continue to cluster them. This is analogous to the situation of a ‘rogue’ ant that seems to be working in a way that is counterproductive to the collective. However, the probabilistic nature of this behaviour can possibly also provide a way out of such behavioural cul-de-sacs.

The employment of six robots in the algorithms presented in this paper is likely to have reduced per capita performance due to the interference that characterises groups of more than three embodied agents [Schneider-Fontán and Mataric, 1998; Krieger *et al.*, 2000]. However, social insects and ants in particular do not have this problem. As the size of the colony increases, per capita productivity either stays the same [Denny, 2003] or increases [Jeanne and Nordheim, 1996] while task specialisation increases [Thomas and Elgar, 2003]. The answer probably lies in the details of individual movement and interaction at the local scale [Sendova-Franks and Franks, 1995; Sendova-Franks and Van lent, 2002]. Therefore, in the future we would like to learn the lessons from ants and apply them to swarm robotics. We think that this is a crucial issue for the future of the field.

¹ *Optimisation for compactness alone would produce a simple mixed cluster. An experiment was conducted to optimise for the sum of compactness and separation, but the results so far showed no increase in compactness while separation suffered greatly.*

5 Conclusion

This study examines different mechanisms which attempt to produce an annular structure, in a collection of objects of different types, similar to structures observed in the brood of *Leptothorax* ants.

The first two mechanisms explore unproven theories about the mechanism the ants use to sort their brood. The first mechanism relies on object size and segregation combined with a simple clustering algorithm. This separation is however weak and it is likely that this does not tell the whole story of how annular structures arise. The second mechanism extends previous research into object segregation and shows that with objects of the same size, segregation can be achieved by varying pullback distance. The problem with this mechanism is that the arena size limits the number of object sizes that can be sorted. The third mechanism introduces the concept of a combined leaky integrator, which, by adapting pullback distances, allows a greater number of object types to be compacted into the same arena size. When a genetic algorithm is used to optimize the parameters of this mechanism even better performance is achieved.

We are currently conducting experiments using *Leptothorax* ants to try to ascertain the rules the ants use to create annular structures within their brood [Sendova-Franks *et al.*, in press]. It will be interesting to see how the above mechanisms relate to ant behaviour.

Acknowledgments

We thank the EPSRC for the on-going funding of this project. We are also grateful to Larry Bull, Tim Swift, Mark Holbrook, Tony Allen and Ian Horsfield for help and advice.

References

- Barker, G.C. and Grimson, M.J. 1990. The physics of muesli. *New Scientist*, 26:37-4.
- Beckers, R., Holland, O. and Deneubourg, J-L. 1994. From local actions to global tasks: stigmergy in collective robotics. In *Proceedings of the Fourth International Conference on Artificial Life*. The MIT Press: Cambridge, MA., pp. 181-189.
- Camazine, S. 1991. Self-organizing pattern formation on the combs of honey bee colonies. *Behavioral Ecology and Sociobiology*, 28:61-76.
- de Jong, K.A. 1975. *Analysis of the Behavior of a Class of Genetic Algorithms*. Ph.D. thesis. University of Michigan, MI.
- Denny, A.J. 2003. *An Experimental Analysis of Polydomy in the Ant Leptothorax albipennis*. Ph.D. Thesis. University of Bristol, UK.
- Deneubourg, J-L., Goss, S., Franks, N., Sendova-Franks, A., Detrain, C. and Chretien, L. 1991. The dynamics of collective sorting: robot-like ants and ant-like robots. In *Proceedings of the First International Conference on Simulation of Adaptive Behavior*. The MIT Press: Cambridge, MA, pp. 356-365.
- Fisher, N.I. 1993. *Statistical Analysis of Circular Data*. Cambridge University Press: Cambridge, UK.
- Franks, N.R. and Sendova-Franks, A.B. 1992. Brood sorting by ants: distributing the workload over the work-surface. *Behavioral Ecology and Sociobiology*, 30:109-123.

- Goldberg, M. 1971. Packing of 14, 16, 17 and 20 circles in a circle. *Mathematics Magazine*, 44: 134-139.
- Holland, J.H. 1975. *Adaptation in Natural and Artificial Systems*. The University of Michigan Press: Ann Arbor, MI.
- Holland, O.E. and Melhuish, C. R. 1999. Stigmergy, self-organisation, and sorting in collective robotics. *Artificial Life*, 5:173-202.
- Jeanne, R.L. and Nordheim, E.V. 1996. Productivity in a social wasp: per capita output increases with swarm size. *Behavioral Ecology*, 7: 43-48.
- Johnson, P.J. and Bay, J.S. 1994. Distributed control of autonomous mobile robot collectives in payload transportation. Technical report. Virginia Polytechnic Institute and State University, U.S.A.
- Kravitz, S. 1967. Packing cylinders into cylindrical containers. *Mathematics Magazine*, 40: 65-70.
- Krieger, M.J.B., Billeter, J.-B. and Keller, L. 2000. Ant-like task allocation and recruitment in cooperative robots. *Nature*, 406:992-995.
- Lubachevsky, B.D. and Graham, R.L. 1997. Curved hexagonal packings of equal disks in a circle. *Discrete and Computational Geometry*, 18: 179-194.
- Lumer, E.D. and Faieta, B. 1994. Diversity and adaptation in populations of clustering ants. *In Proceedings of the Third International Conference on Simulation of Adaptive Behavior*. The MIT Press: Cambridge, MA., pp. 499-508.
- McCoy, E.J. 1991. *Brood Sorting by Ants and Related Spatial Statistics*. M.Sc. Dissertation. University of Bath, U.K.
- Melhuish, C. 1999. *Strategies for Collective Minimalist Mobile Robots*. Ph.D. Thesis. University of the West of England, Bristol, U.K.
- Melhuish, C., Holland, O. and Hoddell, S. 1998. Collective sorting and segregation in robots with minimal sensing. *In Proceedings of the Fifth International Conference on Simulation of Adaptive Behavior*. The MIT Press: Cambridge, MA., pp. 465-470.
- Melhuish, C., Wilson, M. and Sendova-Franks, A.B. 2001. Patch-sorting: multi-object clustering using minimalist robots. Proceedings of 6th European Conference, Prague, *Advances in Artificial Life*, 2159: 543-552.
- Reis, G.E. 1975. Dense packing of equal circles within a circle. *Mathematics Magazine*, 48: 33-37.
- Rao, J.S. 1969. *Some Contributions to the Analysis of Circular Data*. Ph.D. Thesis. Indian Statistical Institute, Calcutta, India.
- Schneider-Fontán, M. and Mataric, M.J. 1998. Territorial multi-robot task division. *IEEE Transactions on Robotics and Automation*, 5:815-822.
- Sendova-Franks, A.B. and Franks, N.R. 1995. Demonstrating new social interactions in ant colonies through randomisation tests: separating seeing from believing. *Animal Behaviour*, 50:1683-1696.
- Sendova-Franks, A.B. and Van lent, J. 2002. Random walk models of worker sorting in ant colonies. *Journal of Theoretical Biology*, 217:255-274.

Sendova-Franks, A.B, Scholes, S.R., Franks, N.R. and Melhuish, C. 2004. Brood sorting by ants: two phases and differential diffusion. *Animal Behaviour*, in press.

Theraulaz, G., Bonabeau, E. and Deneubourg, J-L. 1998. The origin of nest complexity in social insects. *Complexity*, 3: 15-25.

Thomas, M.L. and Elgar, M.A. 2003. Colony size affects division of labour in the ponerine ant *Rhytidoponera metallica*. *Naturwissenschaften*, 90:88-92.