Segregation in Swarms of e-puck Robots Based On the Brazil Nut Effect

Jianing Chen, Melvin Gauci, Michael J. Price and Roderich Groß
Natural Robotics Lab
Department of Automatic Control and Systems Engineering
The University of Sheffield, UK
{j.n.chen, m.gauci, r.gross}@sheffield.ac.uk, michaelprice@theiet.org

ABSTRACT

When a mixture of particles with different attributes undergoes vibration, a segregation pattern is often observed. For example, in muesli cereal packs, the largest particles—the Brazil nuts—tend to end up at the top. For this reason, the phenomenon is known as the Brazil nut effect. In previous research, an algorithm inspired by this effect was designed to produce segregation patterns in swarms of simulated agents that move on a horizontal plane.

In this paper, we adapt this algorithm for implementation on robots with directional vision. We use the e-puck robot as a platform to test our implementation. In a swarm of e-pucks, different robots mimic disks of different sizes (larger than their physical dimensions). The motion of every robot is governed by a combination of three components: (i) attraction towards a point, which emulates the effect of a gravitational pull, (ii) random motion, which emulates the effect of vibration, and (iii) repulsion from nearby robots, which emulates the effect of collisions between disks. The algorithm does not require robots to discriminate between other robots; yet, it is capable of forming annular structures where the robots in each annulus represent disks of identical size.

We report on a set of experiments performed with a group of 20 physical e-pucks. The results obtained in 100 trials of 20 minutes each show that the percentage of incorrectly-ordered pairs of disks from different groups decreases as the size ratio of disks in different groups is increased. In our experiments, this percentage was, on average, below 0.5% for size ratios from 3.0 to 5.0. Moreover, for these size ratios, all segregation errors observed were due to mechanical failures that caused robots to stop moving.

Categories and Subject Descriptors

I.2 [**Artificial Intelligence**]: Robotics

General Terms

Algorithms, Experimentation, Reliability

Keywords

Brazil nut effect, Collective intelligence, Emergent behavior,

Appears in: *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2012)*, Conitzer, Winikoff, Padgham, and van der Hoek (eds.), 4-8 June 2012, Valencia, Spain.
Copyright © 2012, International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). All rights reserved.

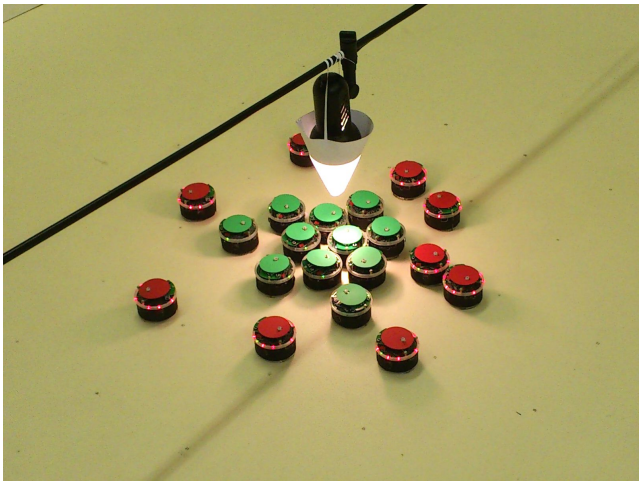


Figure 1: A segregation pattern in a swarm of 20 e-puck robots. The robots have organized into a center-periphery pattern around a light bulb. Robots with green and red top markers emulate disks of radius 8 cm and 16 cm, respectively. Each robot's motion is governed by a combination of three components: (i) attraction towards the light bulb, (ii) random motion, and (iii) repulsion from nearby robots.

Multi-robot systems, Segregation, Self-organization

1. INTRODUCTION

Segregation is a process whereby objects or individuals separate into distinct groups. It can be observed on various scales, ranging from the molecular to the macroscopic scale.

In this paper, we consider forms of segregation that are driven by self-organized processes [6]. We focus on the problem of making a swarm of physical robots self-organize into an annular structure. The robots are all identical in hardware; yet, by executing different behaviors, they are able to form a center-periphery pattern, as shown in Figure 1. We restrict our study to segregation between two groups of robots, mainly because of the limited number of physical robots available. However, the algorithm we use can, in principle, form annular structures with an arbitrary number of groups (and thus layers).

The formation of annular structures, and of center-periphery patterns in particular, might be useful in a range of

applications. Examples include reconfigurable nested membrane structures in biomedical applications and dynamically constructed defense structures in military applications.

A number of studies have looked at spatial segregation using simulated robotic agents. For example, Şahin *et al.* [8] implemented a control law based on a probabilistic framework. Kumar *et al.* [13] implemented a control law based on artificial potential functions. In these studies, segregation is the result of "individual choices that discriminate" [18]. In contrast, our robots are anonymous, and thus unable to discriminate between each other.

Other studies have looked at spatial segregation in the context of macroscopic self-assembly [11]. Bowden *et al.* [5] observed center-periphery structures when millimeter scale objects of two different heights interacted with each other by lateral capillary forces. Ngouabeu *et al.* [16] observed segregation phenomena in a system of vibrating and non-vibrating mechatronic modules that float on the surface of water.

Segregation phenomena observed in ant colonies [10] have inspired the implementation of control laws for robots that organize two distinct groups of items into center-periphery patterns [20, 14] (see also [1]). Unlike these works, our robots segregate themselves and are unable to discriminate between robots (or items) of different groups.

The Brazil nut effect [17] refers to the segregation that occurs when shaking a mixture of granular material of different sizes. Barker and Grimson [3] explain it as follows: "During the periods when shaking loosens the packing, individual small particles can move into voids beneath large particles and so prevent them from returning to their previous positions. It is far less probable that several small particles will move together so as to create a void that can be occupied by a single large particle. The net effect is that the smaller particles occupy the lower positions during the active part of the shaking process and then become trapped there when the grains fix into a new arrangement".

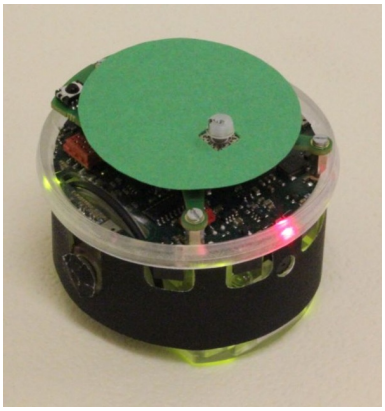
In previous research, a segregation algorithm based on the Brazil nut effect was developed and tested in computer simulation [12]. This algorithm assumed that every robot can instantly measure the relative position of all the robots in its vicinity. Here, we show how this algorithm can be modified to allow for an implementation using directional vision. This implies that (i) robots have to revolve in order to obtain an omni-directional picture and (ii) the algorithm has to cope with misperceptions, for example, due to visual occlusion (see Figure 5). We report on a series of experiments using the modified algorithm that show near error-free segregation in a swarm of 20 physical robots.

2. METHODS

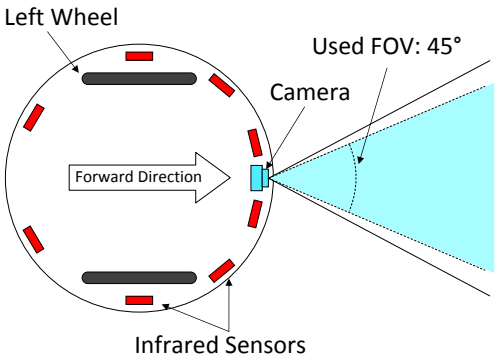
2.1 e-puck Robot

We use a mobile robot called *e-puck* (see Figure 2), which was developed for educational and research purposes [15]. It has a circular body of approximately 7.5 cm diameter, and weighs approximately 150 g. The e-puck is a differential-wheeled robot, having an inter-wheel distance of 5.1 cm.

In order to facilitate visual detection of robots by each other, we fitted every e-puck with a black paper skirt. Moreover, in order to allow for tracking of different groups of robots using an overhead camera system, we fitted the e-pucks with color-coded top markers. Figure 2(a) shows an e-puck fitted with a skirt and a green marker.



(a)



(b)

Figure 2: The e-puck robot. (a) An e-puck fitted with a black skirt and a green top marker. (b) Top-view schematic of an e-puck, indicating the locations of its wheels, camera [including the field of view (FOV)] and infrared sensors.

The e-puck has an RGB color camera located at its front. The camera has a resolution of 640 × 480 pixels (width × height), but the image taken was subsampled to 40 × 15 pixels. The e-puck also has eight infrared (IR) sensors, which are distributed around its body. Here, they are used in a passive mode, in order to detect the angular position of a light bulb within the arena. Figure 2(b) shows a top view of an e-puck, indicating the locations of the wheels, camera and IR sensors.

The robot has an IR receiver which allows IR signals to be sent to it, for example using an IR remote control. Here, we make use of this receiver in order to issue a starting signal to all of the robots at the beginning of every trial.

2.2 Controller

The controller used here is based on the one presented in [12]. Some modifications had to be made in order to port the algorithm onto physical e-puck robots. In the following, we describe the algorithm used here and highlight the modifications made.

The robots emulate a mixture of differently-sized disks subjected to vibration on a 2-dimensional plane. In partic-

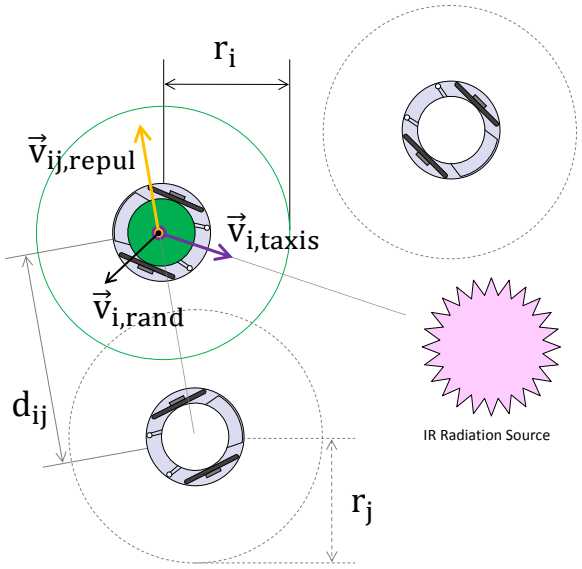


Figure 3: The three behavioral components of robot i. Vector $\vec{v}_{i,\text{taxis}}$ points towards the estimated location of the infrared radiation source. Vector $\vec{v}_{i,\text{rand}}$ points in a random direction. Vector $\vec{v}_{ij,\text{repul}}$ is due to the repulsion effect on robot i by robot j. Robot i is repelled by robot j if it perceives the virtual body of robot j as intersecting with its own virtual body. As robot i has no means of measuring the virtual radius of robot j (r_j), it assumes that $r_j = r_i$.

ular, robot i emulates a disk of radius r_i, whose motion is governed by a combination of three components (see Figure 3):

1. $\vec{v}_{i,\text{taxis}}$: attraction towards a point common to all the disks, which emulates the effect of a gravitational pull,
2. $\vec{v}_{i,\text{rand}}$: random motion, which emulates the effect of vibration and
3. $\vec{v}_{i,\text{repul}}$: repulsion from nearby disks, which emulates the effect of collisions.

Hereafter, the disk a robot represents is also referred to as the *virtual body* of the robot. The radius of the disk is also referred to as the *virtual radius* of the robot.

The behavior is implemented using the motor schema paradigm [2]. In every control cycle, robot i calculates the aforementioned three vectors. These are then combined as follows:

$$\vec{v}_i = \vec{v}_{i,\text{taxis}} + c_{\text{rand}}\vec{v}_{i,\text{rand}} + f(\vec{v}_{i,\text{repul}}). \quad (1)$$

Vector $\vec{v}_{i,\text{taxis}}$ is always a unit vector. Vector $\vec{v}_{i,\text{rand}}$ is also a unit vector but a parameter c_{rand} is used to weight its magnitude. Vector $\vec{v}_{i,\text{repul}}$ can have a large magnitude because it is computed as a sum of possibly many vectors (for details, see Section 2.2.3); therefore, its magnitude is capped by function $f(\cdot)$. Here, we use $c_{\text{rand}} = 0.6$ and a maximum allowed magnitude of 6.4 units for $\vec{v}_{i,\text{repul}}$. These settings follow suggestions from simulation results[1] [12].

[1]The algorithm in [12] uses an additional parameter to weight $\vec{v}_{i,\text{repul}}$. This is not used here because the repulsion mechanism has been modified. The weightings used here are identical to [12] when one considers the maximum allowed magnitude of $\vec{v}_{i,\text{repul}}$.

After constructing motion vector $\vec{v}_i$, robot i first turns to point in its direction, and then moves forward for a fixed duration. The speed at which it moves forward is proportional to the magnitude of the vector, so that the maximum magnitude possible (i.e., $1+0.6+6.4 = 8$ units) corresponds to the maximum speed of the robot (12.8 cm/s).

The length of the control cycle used here is 5 s, which is substantially longer than that used in simulation (0.1 s). The main reason for this is that the e-puck robots are equipped with directional cameras, whereas the simulated robots had omni-directional perception [12]. In each cycle, the robot spends around 2.4 s in revolving to obtain an omni-directional image, 1.3 s in turning to point in the direction of $\vec{v}_i$, and 1.3 s in moving forward.

In the following, we detail how vectors $\vec{v}_{i,\text{taxis}}$, $\vec{v}_{i,\text{rand}}$ and $\vec{v}_{i,\text{repul}}$ are computed.

2.2.1 *Attraction to Center of Gravity*

The algorithm requires a point of attraction in the environment to emulate the effect of a gravitational pull. Each robot is required to estimate the angular position of this point (the distance to it is not needed).

In our experimental setup, we use an infrared radiation source—a light bulb—as the point of attraction. In order to estimate its angular position, each robot makes use of its eight infrared sensors. In every control cycle, the three sensors giving the highest readings are selected. Each reading is then represented as a vector pointing from the center of the robot to the physical location of the sensor, with a magnitude proportional to the sensor's reading. The three vectors are summed, and the resulting vector is normalized to have a unit magnitude, giving $\vec{v}_{i,\text{taxis}}$.

2.2.2 *Random Motion*

The random motion vector $\vec{v}_{i,\text{rand}}$ is taken to be a unit vector pointing in a random direction in the interval $(0, 2\pi]$. This direction is taken with respect to the robot's orientation at the beginning of the control cycle.

2.2.3 *Repulsion*

In principle, each robot should be repelled by every other robot whose virtual body overlaps with its own virtual body. This would require the robots to know the virtual radii of nearby robots. However, as shown in [12], segregation can still be effectively achieved if every robot assumes for all other robots a constant virtual radius, which is a parameter that needs to be fixed a priori. Here, we propose and use an alternative, parameter-free heuristic: robot i assumes that the virtual radius of all other robots is equal to its own, that is, r_i.

In our implementation each robot uses its camera to estimate the angular position of and distance to nearby robots. In every control cycle, a robot turns through one revolution in eight steps of 45° each. In each step, its camera takes a picture. From the center of this picture, a horizontal line of 32 pixels is extracted (corresponding to a field of view of 45°). The pixel lines extracted from the eight images are concatenated to give a panoramic view of the scene (see Figure 4). The concatenated image is traversed horizontally

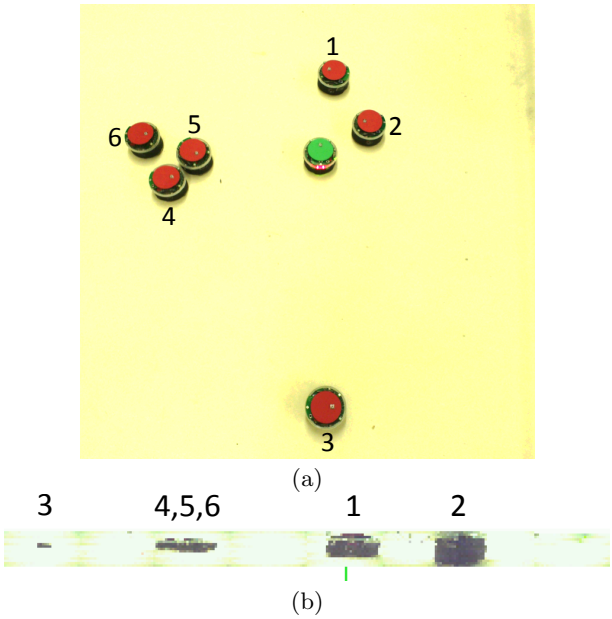


Figure 4: Image processing. (a) Overview of a scene with seven robots. (b) The corresponding concatenated image (here, with the original 15 pixel height) formed by the green robot as it takes eight images in one revolution. Note how the green robot sees the red robots 4, 5 and 6 as a single object that appears closer (see also Figure 5).

to scan for nearby robots. This is achieved by identifying blocks of dark pixels. Each block represents a perceived robot j. The angular position of that robot is estimated from the position of the block. Vector $\vec{v}_{ij,\text{repul}}$ points in the direction away from robot j. The distance to the robot, d_{ij}, is estimated from the width of the block. The amount of repulsion from a perceived robot j is proportional to the perceived amount of intersection. Formally,

$$||\vec{v}_{ij,\text{repul}}|| = \begin{cases} k\,(2r_i - d_{ij}) & d_{ij} < 2r_i; \\ 0 & d_{ij} \geq 2r_i, \end{cases} \tag{2}$$

where $k = 0.2$.

The total repulsion on robot i, $\vec{v}_{i,\text{repul}}$, is giving by summing the individual repulsion vectors for all blocks.

The vision based implementation differs from [12] in that two types of misperceptions can occur: (i) it is possible for several robots to be perceived as a single block of pixels [see Figure 5(a)]; (ii) it is possible for a robot to occlude one or more robots completely [see Figure 5(b)]. In order to compensate for these misperceptions, our repulsion mechanism places more emphasis on robots that are perceived to be close [see Equation (2)]. This is in contrast with the mechanism used in simulation [12], where the amount of repulsion is constant regardless of the distance to a perceived robot.

2.3 Experimental Setup

We use n to denote the number of robots in the swarm. Furthermore, we use m to denote the number of groups, and n_k to denote the number of robots in group k, $k \in \{1, 2, \ldots, m\}$. The robots in group k all have virtual radius $r^{(k)}$. Recall that r_i denotes the virtual radius of robot i.

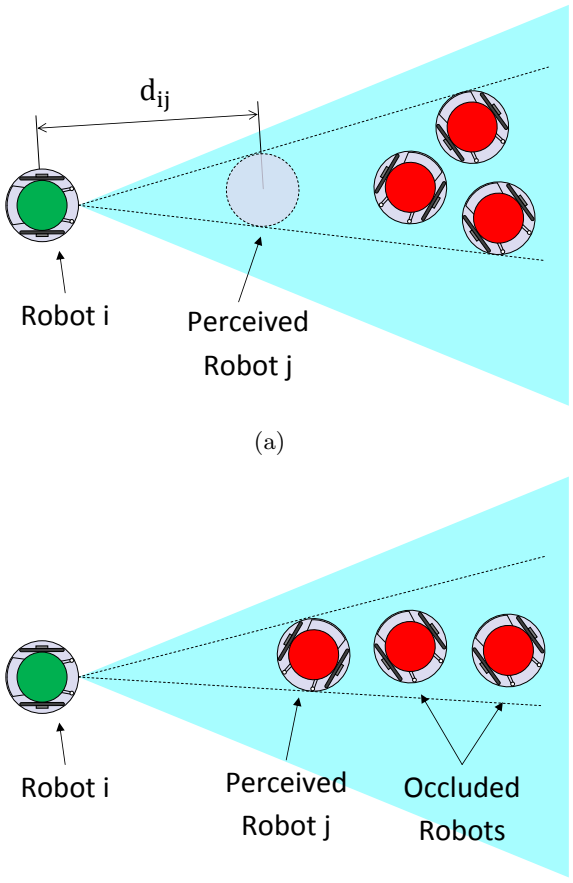


Figure 5: Possible misperceptions. (a) Robot i sees three overlapping robots as a single object, j. It incorrectly perceives a single robot at distance d_{ij}. (b) Robot i can not see the two robots occluded by robot j.

Thus, $r_i = r^{(k)}$, if robot i is in group k.

We consider a system with $n = 20$ robots and with $m = 2$ different groups. The virtual radius of robots from group k is chosen as follows:

$$r^{(k)} = ab^{k-1}, \tag{3}$$

where a is the size (in cm) of the smallest disk and b is the minimum size ratio between disks of different groups. We use $a = 8\,\text{cm}$ and $b \in \{1, 2, 3, 4, 5\}$.

Ideally, we expect the robots to organize into an annular structure, where the disks of radius $r^{(k)}$, $k \in \{1, 2, \ldots, m\}$, are fully contained within the area of the annulus formed by the concentric circles of radii $(k-1)g$ and kg in the center of the environment. Parameter g represents the "thickness" of the annulus and can be controlled by group size n [12].

An approximation of the ideal pattern can be obtained by choosing n_k as follows [12]:

$$n_k = \frac{\frac{2k-1}{(r^{(k)})^2}}{\sum_{j=1}^{m} \frac{2j-1}{(r^{(j)})^2}} n. \tag{4}$$

Table 1: Overview of configurations studied.

radius factor b	n_1	$r^{(1)}$	n_2	$r^{(2)}$
1.0	5	8.0 cm	15	8.0 cm
2.0	11	8.0 cm	9	16.0 cm
3.0	15	8.0 cm	5	24.0 cm
4.0	17	8.0 cm	3	32.0 cm
5.0	18	8.0 cm	2	40.0 cm

In our physical implementation, the robots moved in a square arena of sides 2.5 m. A light bulb was placed over the center of the arena, acting as the infrared radiation source, that is, the point of attraction.

The initial placement of the robots was done as follows: a square grid of 6×6 points was marked on the arena floor, centered around the light bulb, with all points being 20 cm apart. For each trial, 20 points were chosen randomly without replacement. Additionally, for each robot, the orientation was selected randomly from four possibilities: north, south, east and west.

Each trial was recorded from start to finish with an overhead camera system.

2.4 Performance Metric

To measure the quality of segregation, we calculate the *segregation error* (SE) as defined in [12]. Consider two robots i and j and let $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ denote their positions. Furthermore, let $\boldsymbol{o}$ denote the position of the 'center of gravity' in the same co-ordinate system, that is, the point to which all robots are attracted.

The pair of robots (i, j) contributes to the segregation error if one of the robots has a larger virtual radius *and* is closer to $\boldsymbol{o}$ than the other one. It does not contribute to the segregation error if either the robots have identical virtual radii, or if the robot with a smaller virtual radius is closer to $\boldsymbol{o}$ than the other one. Formally,

$$e_{ij} = \begin{cases} 1 & (r_i < r_j) \wedge (\|\boldsymbol{x}_i - \boldsymbol{o}\| \geq \|\boldsymbol{x}_j - \boldsymbol{o}\|)\,; \\ 1 & (r_i > r_j) \wedge (\|\boldsymbol{x}_i - \boldsymbol{o}\| \leq \|\boldsymbol{x}_j - \boldsymbol{o}\|)\,; \\ 0 & \text{otherwise.} \end{cases}$$

The segregation error is given by summing e_{ij} over all pairs of robots, and normalizing by (only) the number of errors possible. Formally,

$$\mathrm{SE} = \frac{\sum_{i=1}^{n} \sum_{j=1}^{n} e_{ij}}{n^2 - \sum_{k=1}^{m} n_k^2}, \tag{5}$$

where $\mathrm{SE} \in [0, 1]$. Randomly placed robots will have a segregation error of 0.5 on average. An error of 1.0 is achieved if the robots are in an 'inverse Brazil nut' configuration, that is, if for all (i, j) s.t. $r_i < r_j$, $\|\boldsymbol{x}_i - \boldsymbol{o}\| \geq \|\boldsymbol{x}_j - \boldsymbol{o}\|$.

3. RESULTS

We considered $m = 2$ groups of robots. Robots of the first group represented disks of radius $r_1 = 8$ cm, whereas robots of the second group represented disks of radius $r_2 = 8b$ cm, $b \in \{1, 2, 3, 4, 5\}$. As reported in [19], the size ratio b is a critical variable—increasing it results in a decrease in the segregation error.

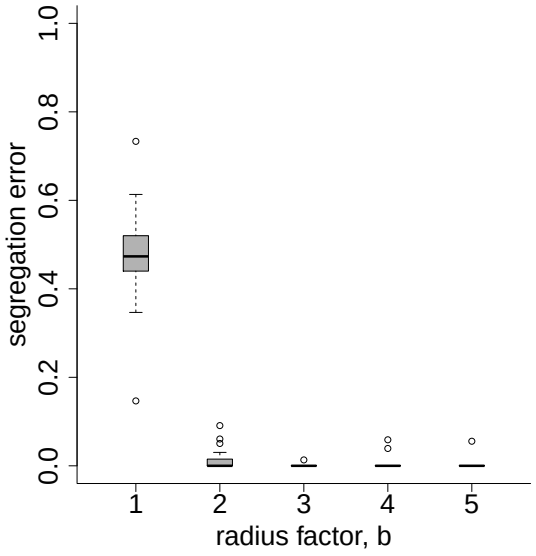


Figure 7: Box-and-whisker plot showing the segregation error observed in experimental trials with 20 e-puck robots for different radius factors (20 trials per radius factor). Each box comprises of observations ranging from the first to the third quartile. The median is indicated by a horizontal bar within the box. The whiskers extend to the farthest data points that are within 1.5 times the inter-quartile range. Outliers are indicated as circles.

For each value of b, we performed 20 trials with $n = 20$ robots each, that is, we ran 100 experimental trials in total. Every trial lasted for 20 minutes. Table 1 shows the number of robots in each group [see Equation (4)].

Figure 6 shows a sequence of snapshots taken during three typical trials with radius factor $b = 1$, 2 and 4.

3.1 Influence of Size Ratio on Segregation Error

Figure 7 shows a box-and-whisker plot [4] of the segregation errors for the different radius factors (b).

For $b = 1$, all e-pucks represented disks of identical size. Consequently, the segregation error (47.3%) was, on average, similar to the expected error for purely randomly distributed e-puck robots (50%). In no trial was perfect segregation observed.

For $b > 1$, the median segregation errors are all 0. The mean segregation errors are 1.31%, 0.07%, 0.49% and 0.28% for $b \in \{2, 3, 4, 5\}$, respectively. For $b = 2$, error free segregation was observed in 14 out of 20 trials. For $b \in \{3, 4, 5\}$, error free segregation was observed in 19, 18 and 19 of the 20 trials, respectively. That is, in these trials, all 20 e-pucks were spatially distributed as intended.

In 4 out of 60 trials for $b \in \{3, 4, 5\}$ the segregation was not error free. This was due to mechanical failures that caused robots to stop moving. For example, a robot became stuck on the arena floor, or lost contact with its battery.

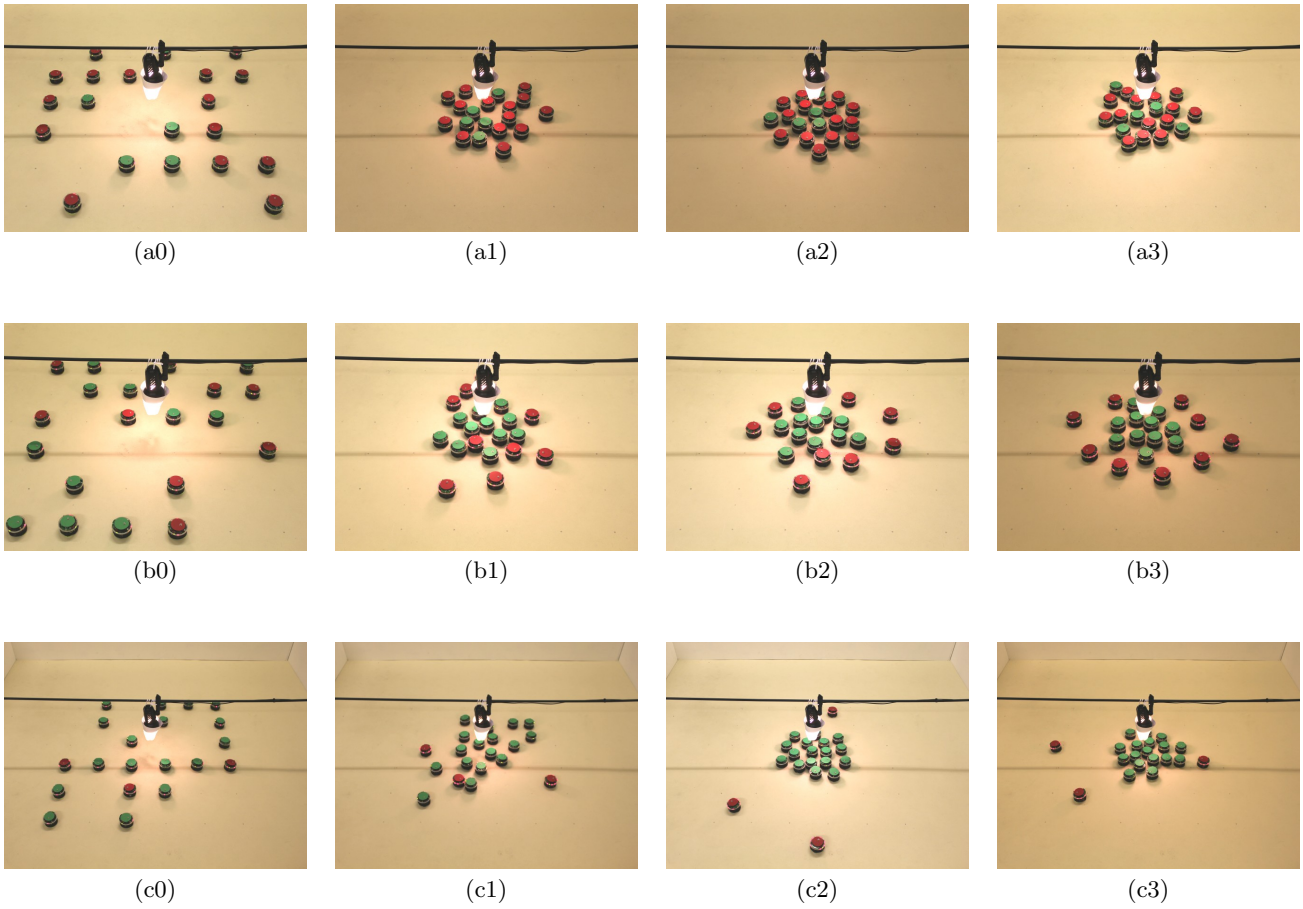


Figure 6: Sequences of snapshots taken during trials with radius factor b equal to 1 (top), 2 (center) and 4 (bottom). Robots with green markers represent disks of 8 cm radius. Robots with red markers represent disks of radius 8 cm (top), 16 cm (center) and 32 cm (bottom). The first and last images in each sequence (from left to right) show the initial and final configurations after 0 and 1200 s. The other two images show intermediate situations.

3.2 Influence of Size Ratio on Spatial Distribution

To understand better the effect of the size ratio (b), we analyzed the spatial distribution of robots of both groups. Figure 8 shows the distances of all robots from the center of the arena as observed at the end of the trial. The data is grouped according to the two groups of robots presenting disks of different sizes in addition to the radius factor used.

For $b = 1$, robots of both groups were similarly distributed in space. The mean distances from the center of 'smaller' robots (green marker) and 'larger' robots (red markers) were 16.9 cm and 17.5 cm, respectively.

As b increased, the distance between robots of different groups increased.

For robots representing small disks (of 8 cm radius), the mean distance from the center of the arena mainly depends on the number of disks of that size. The largest number of small disks was present for $b = 1$ (in this case, all 20 robots were identical). For $b \in \{2, 3, 4, 5\}$, the numbers were 11, 15, 17, and 18, respectively (see Table 1).

The mean distance of 'larger' robots from the center grew almost linearly with the radius factor, setting them spatially apart from the other group. This caused the segregation error to decrease.

3.3 Segregation Dynamics

Figure 9 shows the segregation error over time as observed in trials with radius factor $b = 4$. Initially, the segregation error rapidly decreased until it became zero after 3.5 mins in most of the trials.

4. CONCLUSIONS

In this paper, we studied spatial segregation in a swarm of physical robots. We described how to port an algorithm inspired by the Brazil nut effect from computer simulations [12] to the miniature mobile robot e-puck. The algorithm lets e-pucks mimic a mixture of disks under vibration.

We presented a series of experiments with 20 e-puck robots that confirm the efficacy of the algorithm. The e-pucks were programmed to simulate a system of two groups of disks. The desired target pattern was an annular structure around a common point of attraction, where the robots in each annulus represent disks of identical size. The percentage of incorrectly-ordered pairs of disks from different groups decreased as the size ratio of disks in different groups was in-

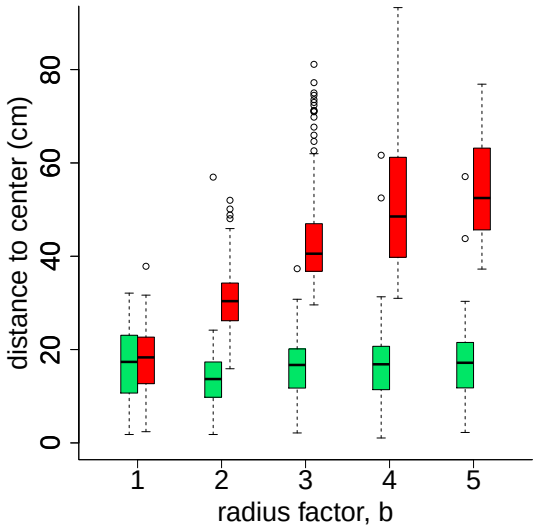


Figure 8: Box-and-whisker plot showing the distances of all robots from the center of the arena for groups of different radius factor (400 data points per radius factor). Green (light gray) boxes represent data from those robots that used the basic virtual radius, whereas red (dark gray) boxes represent data from those robots with the corresponding radius factor applied.

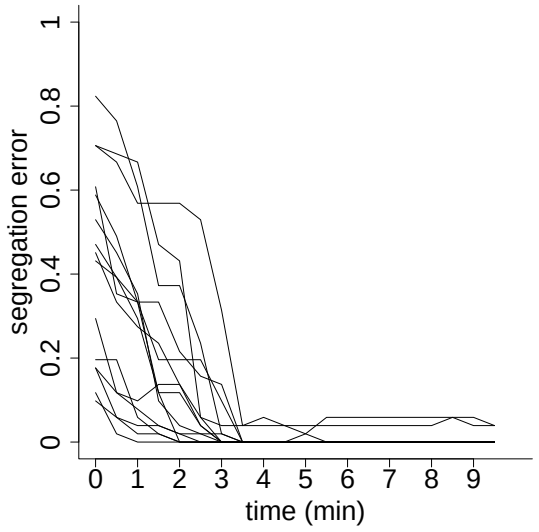


Figure 9: Segregation error over time for 15 experimental trials with 20 e-puck robots and radius factor $b = 4.0$. Data from the remaining five trials are not included because of some missing frames in the corresponding video recordings.

creased.[2] This percentage was, on average, below 0.5% for size ratios from 3.0 to 5.0. Moreover, for these size ratios, all segregation errors observed were due to mechanical failures that caused robots to stop moving. To the best of our knowledge, this is the first example of segregation in a swarm of physical robots with such a high level of accuracy.

The original algorithm in [12] assumed that every robot can instantly measure the relative position of all the robots in its vicinity. Here, we showed how this algorithm can be modified to allow for an implementation using directional vision. This implies that (i) robots have to revolve in order to obtain an omni-directional picture and (ii) the algorithm has to cope with misperceptions, for example, due to visual occlusion. We believe that the new algorithm is applicable to a wider range of robotic platforms when compared to the original algorithm. In principle, the new algorithm can be implemented on any wheeled robot with a camera or equivalent sensor to detect nearby robots. Note that the robot also needs to sense the angular position of a point of attraction in the environment (to emulate the effect of a gravitational pull). Here, we used a light bulb, which was perceived by the e-puck's infrared sensors. In principle, the light bulb could be perceived as well using the directional camera while the e-puck revolves to obtain the omni-directional picture.

The algorithm does not require the robots to communicate, nor does it require them to discriminate between each other. Therefore, the performance of the algorithm could possibly scale well with both the number of robots and the number of groups. Simulation results [12] support this claim; in these, the segregation error decreased exponentially with the size ratio and error free segregation was reported for 150 agents of three distinct groups.

In principle, the algorithm could form annular structures with an arbitrary number of nested layers as well as structures in three dimensions [9]. A present limitation, however, is that the robots' minimum sensing range could be required to increase exponentially with the number of layers.

5. ACKNOWLEDGMENTS

The research work disclosed in this publication is funded by the Marie Curie European Reintegration Grant within the 7$^{\text{th}}$ European Community Framework Programme (grant no. PERG07-GA-2010-267354).

M. Gauci acknowledges support by the Strategic Educational Pathways Scholarship (Malta). The scholarship is part-financed by the European Union – European Social Fund (ESF) under Operational Programme II – Cohesion Policy 2007–2013, "Empowering People for More Jobs and a Better Quality of Life".

The authors thank Andrew Hills for helpful comments on an earlier version of this paper.

6. REFERENCES

[1] M. Amos and O. Don. Swarm-based spatial sorting. *International Journal of Intelligent Computing and Cybernetics*, 1(3):454–473, 2008.

[2] R. C. Arkin. Motor schema-based mobile robot navigation. *Int. J. Robot. Res.*, 8(4):92–112, 1989.

[3] G. Barker and M. Grimson. The physics of muesli. *New Sci.*, 126(1718):37–40, 1990.

[2]Video recordings of the experimental trials can be found in the online supplementary material [7].

[4] R. A. Becker, J. M. Chambers, and A. R. Wilks. *The new S language. A programming environment for data analysis and graphics*. Chapman & Hall, London, 1988.

[5] N. Bowden, A. Terfort, J. Carbeck, and G. M. Whitesides. Self-assembly of mesoscale objects into ordered two-dimensional arrays. *Science*, 276(5310):233–235, 1997.

[6] S. Camazine, J.-L. Deneubourg, N. R. Franks, J. Sneyd, G. Theraulaz, and E. Bonabeau. *Self-organization in biological systems*. Princeton Univ. Press, Princeton, NJ, 2001.

[7] J. Chen, M. Gauci, M. J. Price, and R. Groß. Online supplementary material. `http://naturalrobotics.group.shef.ac.uk/supp/2012-001`, 2012.

[8] E. Şahin, T. Labella, V. Trianni, J.-L. Deneubourg, P. Rasse, D. Floreano, L. Gambardella, F. Mondada, S. Nolfi, and M. Dorigo. Swarm-bot: Pattern formation in a swarm of self-assembling mobile robots. In *Proc. of the 2002 IEEE Int. Conf. on Systems, Man and Cybernetics (SMC 2002)*, volume 4. IEEE Computer Society Press, Los Alamitos, CA, 2002.

[9] S. Foster and R. Groß. Forming nested 3D structures based on the Brazil nut effect. In *Proc. of the 12th Conf. Towards Autonomous Robotic Systems (TAROS 2011)*, volume 6856 of *Lecture Notes in Artificial Intelligence*, pages 394–395, Berlin, Germany, 2011. Springer-Verlag.

[10] N. R. Franks and A. B. Sendova-Franks. Brood sorting by ants: distributing the workload over the work-surface. *Behav. Ecol. Sociobiol.*, 30(2):109–123, 1992.

[11] R. Groß and M. Dorigo. Self-assembly at the macroscopic scale. *P. IEEE*, 96(9):1490–1508, 2008.

[12] R. Groß, S. Magnenat, and F. Mondada. Segregation in swarms of mobile robots based on the Brazil nut effect. In *Proc. of the 2009 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS 2009)*, pages 4349–4356. IEEE Computer Society Press, Los Alamitos, CA, 2009.

[13] M. Kumar, D. Garg, and V. Kumar. Segregation of heterogeneous units in a swarm of robotic agents. *IEEE T. Automat. Contr.*, 55(3):743–748, 2010.

[14] C. Melhuish, A. B. Sendova-Franks, S. Scholes, I. Horsfield, and F. Welsby. Ant-inspired sorting by robots: the importance of initial clustering. *J. R. Soc. Interface*, 3(7):235–242, 2006.

[15] F. Mondada, M. Bonani, X. Raemy, J. Pugh, C. Cianci, A. Klaptocz, S. Magnenat, J.-C. Zufferey, D. Floreano, and A. Martinoli. The e-puck, a robot designed for education in engineering. In *Proc. of the 9th Conf. on Autonomous Robot Systems and Competitions*, volume 1, pages 59–65. IPCB: Instituto Politécnico de Castelo Branco, 2009.

[16] A. M. T. Ngouabeu, S. Miyashita, R. M. Füchslin, K. Nakajima, M. Göldi, and R. Pfeifer. Self-organized segregation effect on self-assembling robots. In *Proc. of the 12th Int. Conf. on the Synthesis and Simulation of Living Systems (Artificial Life XII)*, pages 232–238. MIT Press, Cambridge, MA, 2010.

[17] A. Rosato, K. J. Strandburg, F. Prinz, and R. H. Swendsen. Why the Brazil nuts are on top: size segregation of particulate matter by shaking. *Phys. Rev. Lett.*, 58(10):1038–1040, 1987.

[18] T. C. Schelling. Models of segregation. *Am. Econ. Rev.*, 59(2):488–493, 1969.

[19] J. C. Williams and M. I. Khan. The mixing and segregation of particulate solids of different particle size. *The Chemical Engineer*, 269:19–25, 1973.

[20] M. Wilson, C. Melhuish, A. B. Sendova-Franks, and S. Scholes. Algorithms for building annular structures with minimalist robots inspired by brood sorting in ant colonies. *Auton. Robot.*, 17(2–3):115–136, 2004.