This article was downloaded by: [K F Univ Graz], [Michael Bodi]
On: 12 September 2011, At: 07:39
Publisher: Taylor & Francis
Informa Ltd Registered in England and Wales Registered Number: 1072954 Registered office: Mortimer House, 37-41 Mortimer Street, London W1T 3JH, UK

Mathematical and Computer Modelling of Dynamical Systems

Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/nmcm20

Interaction of robot swarms using the honeybee-inspired control algorithm BEECLUST

Michael Bodi [a], Ronald Thenius [a], Martina Szopek [a], Thomas Schmickl [a] & Karl Crailsheim [a]

[a] Department of Zoology, Karl-Franzens Universitäät Graz, Universitäätsplatz 2, A-8010, Graz, Austria

Available online: 01 Aug 2011

To cite this article: Michael Bodi, Ronald Thenius, Martina Szopek, Thomas Schmickl & Karl Crailsheim (2011): Interaction of robot swarms using the honeybee-inspired control algorithm BEECLUST, Mathematical and Computer Modelling of Dynamical Systems, DOI:10.1080/13873954.2011.601420

To link to this article: http://dx.doi.org/10.1080/13873954.2011.601420

PLEASE SCROLL DOWN FOR ARTICLE

Full terms and conditions of use: http://www.tandfonline.com/page/terms-and-conditions

This article may be used for research, teaching and private study purposes. Any substantial or systematic reproduction, re-distribution, re-selling, loan, sub-licensing, systematic supply or distribution in any form to anyone is expressly forbidden.

The publisher does not give any warranty express or implied or make any representation that the contents will be complete or accurate or up to date. The accuracy of any instructions, formulae and drug doses should be independently verified with primary sources. The publisher shall not be liable for any loss, actions, claims, proceedings, demand or costs or damages whatsoever or howsoever caused arising directly or indirectly in connection with or arising out of the use of this material.

Mathematical and Computer Modelling of Dynamical Systems
i*First*, 2011, 1–14

Interaction of robot swarms using the honeybee-inspired control algorithm BEECLUST

Michael Bodi*, Ronald Thenius, Martina Szopek, Thomas Schmickl and Karl Crailsheim

Department of Zoology, Karl-Franzens Universität Graz, Universitätsplatz 2, A-8010 Graz, Austria

(*Received 29 April 2011; final version received 11 May 2011*)

In this work we investigated how robust a robot swarm acts against disturbances caused by another robot swarm, both using the BEECLUST algorithm, which is inspired by honeybee behaviour. For our investigation we simulated an environment with an ambient illuminance, a light spot and a shadow spot. In such an environment we tested two different castes of Jasmine III robots whereas each caste had to perform a different task. One swarm aggregates at places of high illuminance (light spot) and the other one at places of low illuminance (shadow spot). We show that small swarm populations can benefit from the presence of another robot swarm. Medium populated swarms are affected neither positively nor negatively. Large swarm populations act robust against disturbances caused by other robot swarms as long as no jamming effects occur. In this article we show that the BEECLUST algorithm provides all features for making collective decisions. Furthermore we show that the robustness of the BEECLUST algorithm allows us to control a heterogeneous robot swarm in environments which demand differing controller strategies and swarm intelligent behaviour.

Keywords: swarm robotics; collective decision; swarm intelligent

Downloaded by [K F Univ Graz], [Michael Bodi] at 07:39 12 September 2011

1. Introduction

Swarms of autonomous entities (animals, robots) show interesting features: huge numbers of agents act and interact in high densities and often perform their tasks in separated specialized cohorts [1]. The regulation of the agents' behaviours is often self-organized and swarm intelligent [2–4], features that are mainly achieved by various interlinked feedback loops within the swarm system. A prime example for such self-organized and swarm-intelligent systems would be the nectar source selection by honeybees or the trail formation in ants [5].

The requirements for identifying a system as swarm intelligent were defined by Millonas [6] and in other terms also by Sahin [7]:

(1) *Proximity principle*: The swarm should be able to do elementary space and time computation whereas in this case computation is understood as a direct behavioural reaction on a given environmental stimulus.

*Corresponding author. Email: michael.bodi@uni-graz.at

ISSN 1387-3954 print/ISSN 1744-5051 online
© 2011 Taylor & Francis
DOI: 10.1080/13873954.2011.601420
http://www.informaworld.com

(2) *Quality principle*: The swarm should be able to determine and react to the quality of an environmental stimulus.
(3) *Principle of diverse response*: The swarm should not narrow down its exploitation of a certain resource to much as an insurance against sudden changes in the environment.
(4) *Principle of stability*: The swarm should not shift its behaviour on every small environmental change.
(5) *Principle of adaptability*: On the other hand, the swarm should be able to change its behaviour to clear environmental changes. So an optimal solution for the swarm would be a good mixture of principles 4 and 5.

When creating artificial (e.g. robotic) swarm systems, the feedback networks which determine self-organized behaviour can quickly get rather complex and swarm behaviours can become hard to predict. This generates the demand for finding simple behavioural rules for individuals and also simple rules of interactions. Such rules can often be found in natural swarm systems [8–10]. This allows extraction of such regulation mechanisms and translation of these feedbacks to artificial swarm systems, which in turn show comparable behaviour like the natural sources of inspiration [11–13]. We recently demonstrated such a translation of natural algorithms to a swarm robotic algorithm by developing the BEECLUST [14–17].

Young honeybees prefer an ambient temperature of 36 °C which corresponds to the temperature of the brood nest in the beehive [18,19]. Recent experiments with young honeybees in a temperature gradient have shown that a single bee moves almost randomly through an arena (see Figure 1(a)) [15,20]. Some bees exhibit a slight bias towards the warm area during its random walk but it hardly ever stays there. In contrast, groups of bees (Figure 1(b)) show aggregation behaviour in or close to the warm spot. The bees still move almost randomly but the clusters which are accidentally formed in warm areas last longer than clusters formed in cooler areas. At the end of the experiment, almost all bees cluster in the warm area [15,20] (see Figure 1(c)). This interesting behaviour inspired us to develop the BEECLUST algorithm which is described in detail in Section 1.1.

Such bio-inspired algorithms are frequently investigated in the field of autonomous robotics in the recent years and are developed using a big variety of techniques: Halloy

Downloaded by [K F Univ Graz], [Michael Bodi] at 07:39 12 September 2011

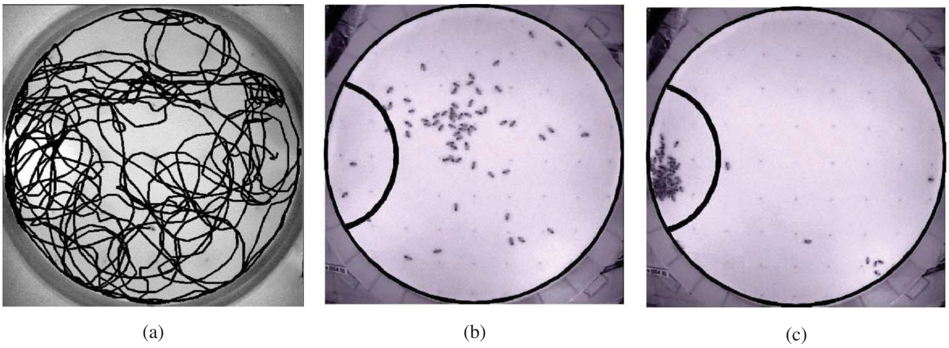

Figure 1. Aggregation behaviour of young honeybees. The ambient temperature in the arena is 30 °C. On the left side is a temperature optimum with 36 °C (marked with a black line in (b) and (c)). (a) Trajectory of a single bee. The single bee does not show a clear preference to the temperature optimum. (b) A group of bees are released in the centre of the arena ($t = 0$ min). (c) At the end of the experiment, most of the bees are clustered in the temperature optimum ($t = 30$ min).

Downloaded by [K F Univ Graz], [Michael Bodi] at 07:39 12 September 2011

et al. [21] experimented with robots which were able to integrate in insect societies. The field of bio-inspired control algorithms for large populated robot swarms has also been investigated intensively: In the I-Swarm project [22], techniques for controlling a swarm of 1000 microrobots were investigated. The Swarm-Bot project [23,24] investigated the self-organizing abilities of a robot swarm, consisting of simpler, insect-like robots. In the Symbrion project [25], the concept of a swarm of reconfigurable robots is investigated. The Replicator project [26] plans to develop an autonomous robot swarm for industrial purposes.

Compared to single robot systems, such robot swarms have the advantage to work robustly, because the removal or malfunction of single individuals affects the behaviour of the whole swarm only slightly. Robot swarms are also able to act 'swarm intelligent' which permits a group of individuals to make decisions to reach a common goal in a decentralized manner. In the following, we demonstrate how the BEECLUST algorithm scales with swarm sizes and how cooperation can emerge (without being pre-programmed) between several interacting robot swarms.

1.1. The BEECLUST algorithm

In recent works it has been shown that a swarm of Jasmine III robots using the BEECLUST algorithm are able to find locations of maximum illumination using a few simple rules [15] (see Figure 2):

(1) The robots move straight through the arena. Whenever a robot detects an obstacle, it stops and checks whether the obstacle is another robot or a wall by watching for emitted IR signals.
(2) If the detected obstacle does not emit IR signals it is a wall. The robot turns randomly and continues with Step 1.
(3) If the detected obstacle does emit IR signals it is another robot. The robot measures the local illuminance.

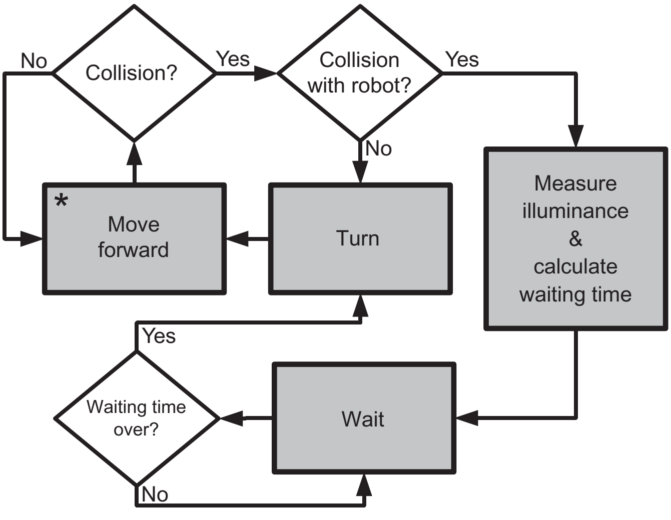

Figure 2. Finite-state machine of the BEECLUST controller. Boxes represent the different behavioural states of a robot. Asterisk indicates the starting state of the algorithm. Diamonds represent control structures (if–else decisions).

Downloaded by [K F Univ Graz], [Michael Bodi] at 07:39 12 September 2011

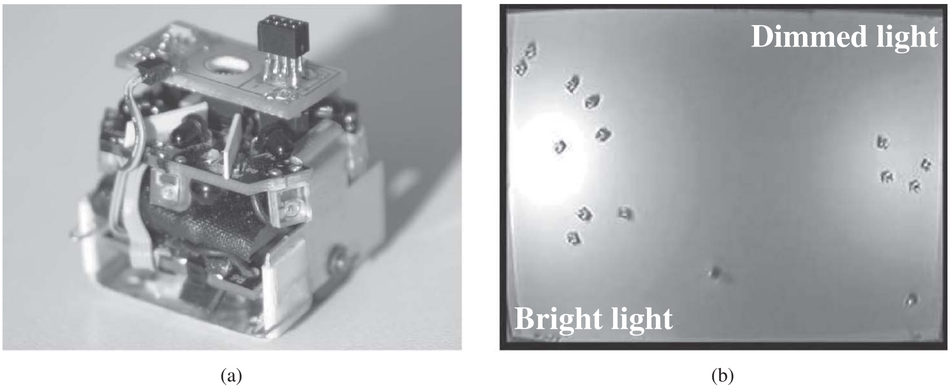

Figure 3. (a) A single Jasmine III robot. (b) A swarm of Jasmine III robots in a real-life experiment. Most of the robots are aggregated in the bright area of the arena. Reprinted from [15].

(4) Depending on the local illuminance, the robot calculates a waiting time: the higher the local illuminance, the longer the waiting time. After the waiting time is over, the robot turns randomly and continues with Step 1.

Our recent studies showed that the BEECLUST algorithm allows a robot swarm to aggregate collectively at a light spot (see Figure 3(b)) [15]. This collective ability is achieved by robots that move purely random, and which have only one illuminance sensor that does not allow a directed gradient exploitation. Also no memory of past illuminance values is involved in the algorithm. The measurements of the environment occur rarely, only after robot-to-robot collisions. Although the individual 'intelligence' of single robots is kept quite low, the swarm as a total was found to act very intelligently, even in dynamic environments. The reasons for this emergent ability are the feedback loops that arise within the robot swarm, as it is explained in detail in [12,27]. Based on the work of [15], we investigated how the robustness of the BEECLUST algorithm is affected by the swarm population and how it is influenced by a second group of robots in the arena performing a different task at the same time.

2. Methods

The experiments in this work were performed with simulated Jasmine III robots. The Jasmine III robot (Figure 3(a)) has a size of about 30 × 30 × 25 mm (L × W × H). It has two wheels as actuators which can be turned in both directions. For distance measurement and for collision avoidance the Jasmine III has six IR sensors. The sensors range is approximately 60 mm. For light measurement the Jasmine III has two light sensors on top. A detailed description of the robot hardware can be found in [15,16] and building instructions are given in [24].

For our experiments we used SMARS (Small Robot Simulator), a simulation environment programmed in NetLogo [28], which was built for performing experiments with simulated Jasmine III robots (see Figure 4). In this simulation environment we simulated two different 'castes' of Jasmine III robots which were implemented with the physical properties of the real robots. The only difference between the two castes is that one caste waits longer at places of high illuminance (light finders) and the other one at places of low illuminance (shadow finders). All other properties of the robots are equal.

We generated a virtual arena with the dimension of 49 × 29 spatial units. These spatial units are called patches and have a base area of 3 × 3 cm^2, which is equal to the base

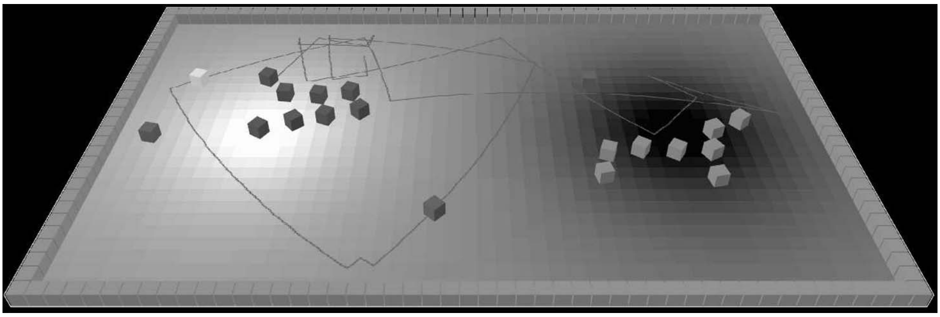

Figure 4. Screenshot of two robot swarms acting in the simulation environment SMARS. The white spot is the optimum for the 'light finders' with 1000 lux and the black spot is the optimum for the 'shadow finders' with 0 lux. The ambient illuminance is 500 lux. Robots are represented by boxes which indicate their caste: dark boxes are 'light finders' and bright boxes are 'shadow finders'. The line shows a trajectory of a randomly chosen robot for demonstration purposes.
Note: SMARS, Small Robot Simulator.

Downloaded by [K F Univ Graz], [Michael Bodi] at 07:39 12 September 2011

area of a Jasmine III robot. In this arena we implemented an environment with an ambient illuminance of 500 lux. We added a light spot with 1000 lux and a shadow spot with 0 lux (see Figure 5).

2.1. *The simulation*

In our simulation each infrared sensor s is modelled as a set of two values:

$$s = \{\alpha, d\}, \tag{1}$$

whereby α is the perception angle of the sensor and d its range. The sensors are implemented as a list of sensors S

$$S = \{s_1, \ldots, s_6\}. \tag{2}$$

For each sensor s of S, an object O and an object R are generated; O is the closest object (including other robots) and R is the closest robot. When robots detect an obstacle they emit an infrared signal and switch to a perception mode in which they wait for other infrared signals. (In this way a robot discriminates an obstacle from another robot because obstacles reflect IR light but do not emit IR light by themselves.) First we calculate whether a robot in the sensor cone could possibly be occluded by another object (expressed by i) by determining the distance of the robot c_R and the distance to the occluding object c_O:

$$i = \begin{cases} \text{true}, & \text{if } c_O < c_R \\ \text{false}, & \text{else} \end{cases}. \tag{3}$$

In a next step we calculate whether a robot is visible behind an obstacle (expressed by j). Therefore the angles β_O and β_R are defined for each O and each R in the sensor cone.

$$j = \begin{cases} \text{true}, & \text{if } |\beta_R - \beta_O| < arctan\frac{\frac{a_O}{2}}{c_O} \\ \\ \text{false}, & \text{else} \end{cases}, \tag{4}$$

Downloaded by [K F Univ Graz], [Michael Bodi] at 07:39 12 September 2011

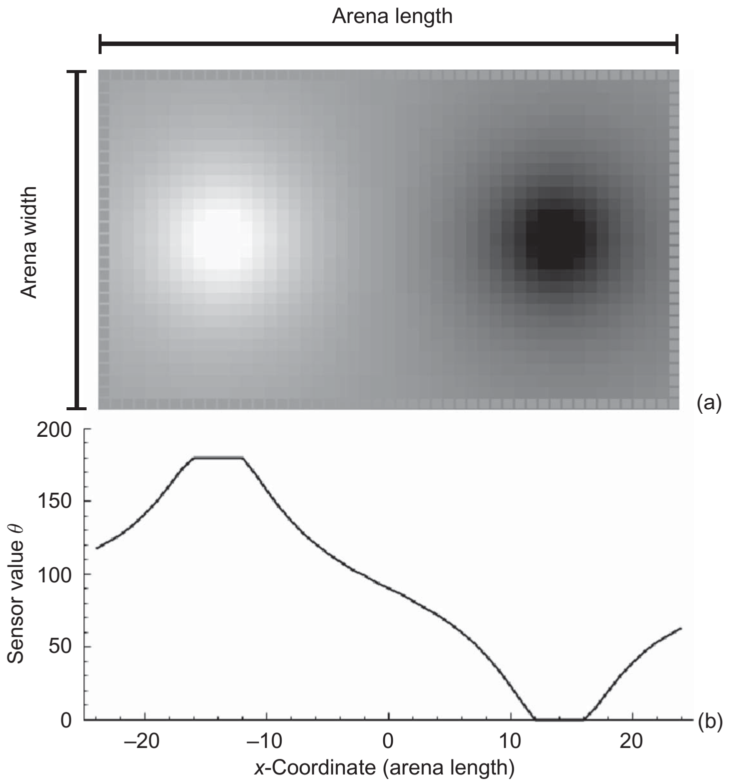

Figure 5. (a) Overhead screenshot of the arena in SMARS. The bright area indicates the light spot and the dark area indicates the shadow spot. Arena length = 49 patches, arena width = 29 patches. (b) Sensor values detected by the robots. Samples are taken from patches with $y = 0$, which is the vertical centre of the arena.

whereby a is the diameter of the seen object.

There is a direct proportion between the illuminance and the sensor value whereby the maximal sensor value (180) is reached at 1000 lux and above. In our model sensor values of the light sensors are a linear function of local illuminance, as it is explained by Equation (5):

$$e_{x,y} = \begin{cases} E_{x,y}\theta, & \text{if } e_{x,y} \leq E_{\max} \\ e_{\max}, & \text{else} \end{cases}, \tag{5}$$

whereby $e_{x,y}$ represents the sensor value of the robot's light sensor at the position x, y in the arena and $e_{\max}$ the maximum sensor value of a robot. $E_{x,y}$ is modelled as the illuminance at the position x, y and $E_{\max}$ as the maximal perceived light value. θ refers to the translation coefficient which converts measured light values to sensor values. For simulating the turning behaviour of a robot, the turnangle γ and the turnradius r have to be calculated:

$$\gamma = \frac{360p}{2r\pi}, \tag{6}$$

$$r = \frac{r_R}{1 - \frac{p}{q}}, \tag{7}$$

whereby p is the distance which is covered by the outer actuator and q is the distance which is covered by the inner actuator. r_R represents the diameter of a robot. The turning of a robot takes place in three steps: first the robot turns $\gamma/2$, then the robot moves forward. In the last step the robot turns $\gamma/2$ again.

2.2. *The controller*

As mentioned above, each robot which meets another robot calculates its waiting time individually, depending on the local illuminance but regardless of the other robots' caste. The waiting time of the 'light finders' is modelled as

$$w^L_{x,y} = \frac{w_{\max}}{1 + \exp^{-(e_{x,y}+o)\sigma}}, \tag{8}$$

and for 'shadow finders' it is modelled as

$$w^S_{x,y} = \frac{w_{\max}}{1 + \exp^{-(e_{\max}+e_{x,y}+o)\sigma}}, \tag{9}$$

whereby $w^L_{x,y}$ refers to the waiting time at the position x, y for 'light finders' and $w^S_{x,y}$ for 'shadow finders'. $w_{\max}$ is modelled as the maximum waiting time, $e_{x,y}$ is the robot's actual sensor value, $e_{\max}$ is the maximum sensor value, o is the vertical offset and σ is the steepness. Equations (8) and (9) are both illustrated in Figure 6. The given values for the constants can be seen in Table 1.

2.3. *The experiments*

To find the optimal density of a Jasmine III swarm, we first tested groups of differing population sizes only consisting of 'light finders'. We defined a target zone for the 'light finder' caste consisting of all patches having a light value between 600 and 1000 lux (this is 40% of the maximum light value in the arena). For each swarm population, we repeated the experiment six times for 4 simulated minutes. We defined two different measurable criteria

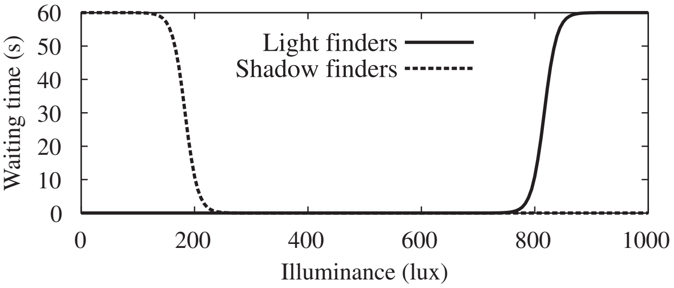

Figure 6. Illustration of the dependence between the local illuminance $e_{x,y}$ and the waiting time $w_{x,y}$. The solid drawn function was implemented in the 'light finders', and the dashed drawn function was implemented in the 'shadow finders'.

Downloaded by [K F Univ Graz], [Michael Bodi] at 07:39 12 September 2011

Table 1. Constants and their values used in Equations (1)–(3) in our simulation runs.

Constants	Values	Dimensions
$w_{\max}$	60	s
$e_{\max}$	180	Dimensionless
$E_{\max}$	1000	lux
θ	0.18	Dimensionless
o	147	Dimensionless
σ	0.5	Dimensionless
r_R	30	mm

Downloaded by [K F Univ Graz], [Michael Bodi] at 07:39 12 September 2011

for aggregation success: aggregation quality and aggregation speed. To get an overview of the aggregation quality of our robot swarm, we monitored the ‘light finders’ in the state ‘wait’ located in the target zone within the last minute of every repetition. To analyse the aggregation speed, we monitored the point of time when 50% of the ‘light finder’ population aggregated within the target zone (A_{50}). To test the robustness of a robot swarm against disturbances caused by another swarm performing another task, we added groups of variable numbers of ‘shadow finders’ and observed how they affect the aggregation behaviour of the ‘light finders’.

3. Results

Our results show that there is an optimal density of the tested robot swarm for the used environmental conditions. This optimal density was found at a swarm population of 9 individuals (see Figure 7). A swarm consisting of 9 individuals is able to aggregate, on average, 80% of its individuals in the target zone.

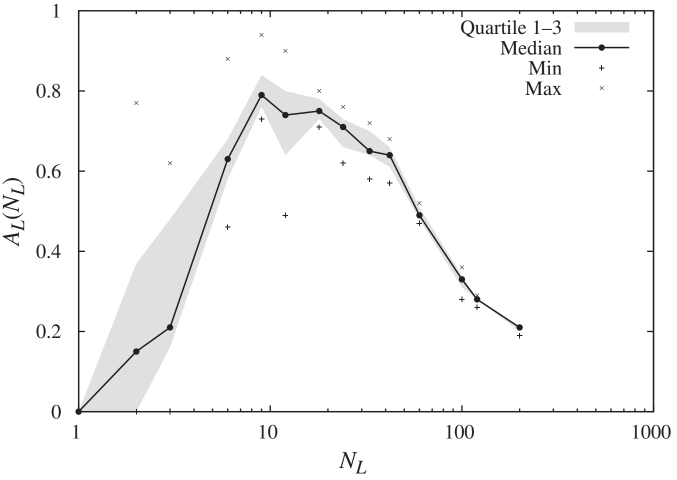

Figure 7. Fraction of aggregated ‘light finders’ $A_L(N_L)$ in target zone over the last minute of observation. The optimal swarm population for the given environment was found at 9 robots. The tested swarm populations were 2, 3, 6, 9, 12, 18, 24, 33, 42, 60, 100 and 120 individuals; $n = 6$ repetitions per experiment.

Downloaded by [K F Univ Graz], [Michael Bodi] at 07:39 12 September 2011

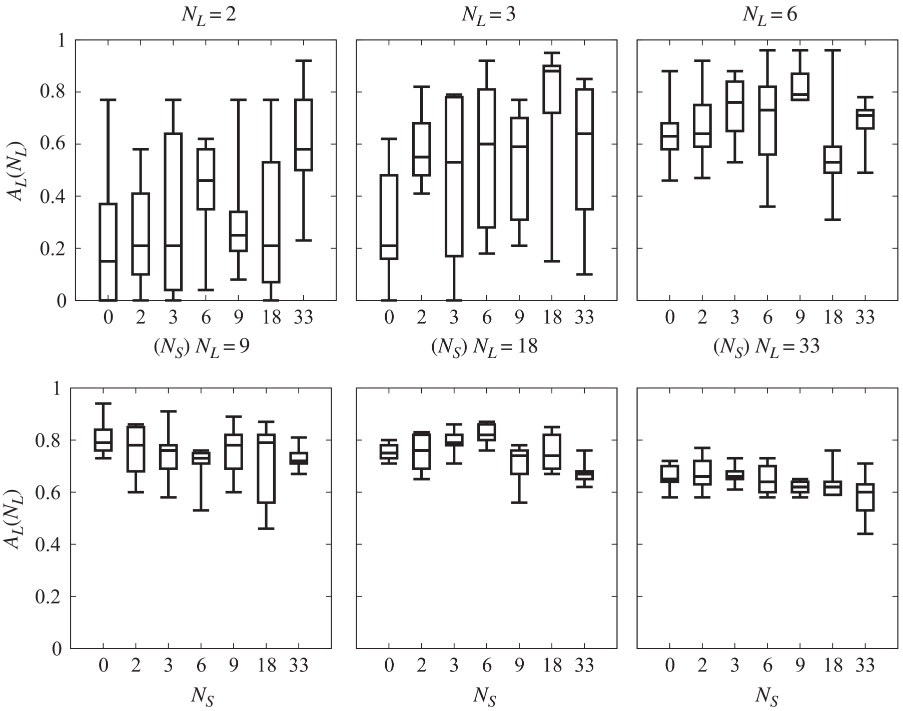

Figure 8. Fractions of 'light finders' aggregated in the target zone within the last minute of observation (median ± quartiles, min and max). N_L is the total number of 'light finders' and $n = 6$ repetitions per experiment.

In Figure 8 we demonstrate that in small robot populations ($N_L = 2,\ 3$), the aggregation quality increases with the number of other robots in the arena. Medium populated robot swarms ($N_L = 6{-}18$) have a constant fraction of robots aggregated under the light source, independent of the number of other robots in the arena. Larger swarms ($N_L = 33$) show a decreasing fraction of aggregated robots with increasing number of other robots in the arena.

Concerning the aggregation speed , we show that small robot populations ($N_L = 2{-}3$) do no aggregate under the light source as long as no other robots ($N_S = 0$) are present in the arena (Figure 9). The aggregation speed increases with the number of other robots ($N_S > 3$). Medium populated robot swarms ($N_L = 6{-}18$) aggregate with the same speed under the light source, independent of the number of other robots in the arena. Larger robot swarms ($N_L > 18$) show a decreasing aggregation speed with increasing number of other robots in the arena. To allow the investigations of the observed enhancements of aggregation, which the focal robot swarm can draw from the presence of the other swarm, we developed two numerical expressions of aggregation enhancement: ΔA_L, which expresses the increase of the aggregation fractions in the presence of 33 'shadow finders' compared to runs were 'light finders' acted alone (see Equation (10)). In addition, ΔA_{50} expresses the increase in the speed of aggregation when such simulation runs were compared (see Equation (11)).

Downloaded by [K F Univ Graz], [Michael Bodi] at 07:39 12 September 2011

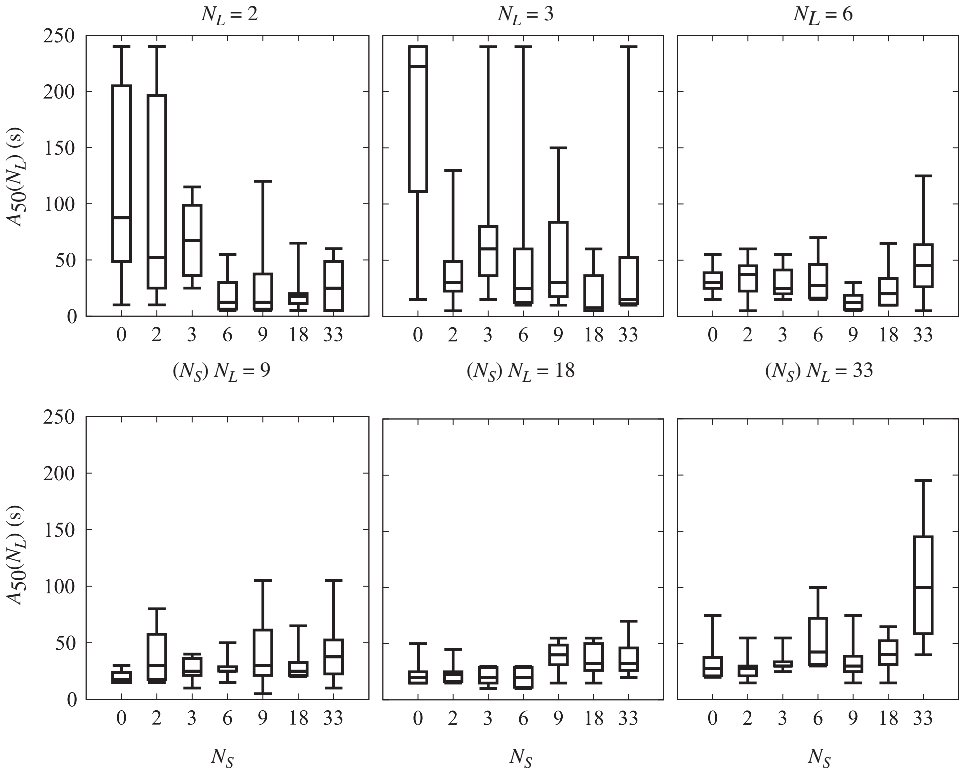

Figure 9. Time in which 50% of the 'light finder' swarm aggregated in the target zone ($A_{50}N_L$) at different population sizes of 'shadow finders' N_S (median quartiles, min and max). N_L is the number of 'light finders' and $n = 6$ repetitions per experiment.

$$\Delta A_L(N_L) = A_L(N_S = 33) - A_L(N_S = 0). \tag{10}$$

Figure 10, where $\Delta A_L(N_L)$ was evaluated for different swarm sizes of 'light finders', shows that small populations show an enhancement of their ability to aggregate by increasing number of other robots, whereas larger populations are not affected, whereby $\Delta A_L(N_L)$ represents the difference between the fraction of 'light finders' in the target zone at a 'shadow finder' population of 33 $A_L(N_S = 33)$ and a 'shadow finder' population of 0 $A_L(N_S = 0)$.

To investigate also the advancements in aggregation speed, which are caused by the presence of other robot swarm, we defined an aggregation enhancement ΔA_{50} according to

$$\Delta A_{50}(N_L) = A_{50}(N_S = 0) - A_{50}(N_S = 33), \tag{11}$$

whereby $A_{50}(N_S = 0)$ is the point of time when 50% of the 'light finders' aggregate under the light source by a population of 0 'shadow finders' and $A_{50}(N_S = 33)$ a population of 33 'shadow finders'. In Figure 11 we show that small populations of 'light finders' aggregate faster by increasing number of 'shadow finders'. Medium populated swarms only take

Downloaded by [K F Univ Graz], [Michael Bodi] at 07:39 12 September 2011

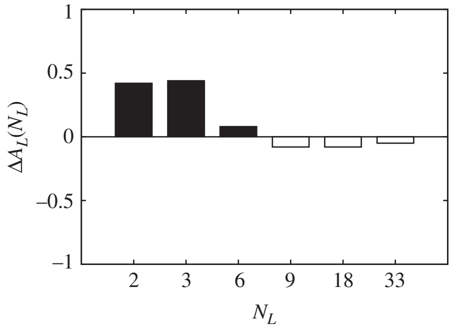

Figure 10. The enhancement of aggregation $\Delta A_L(N_L)$ which is the difference of fraction between $N_S = 0$ and $N_S = 33$, for $N_L = 2$–33, is shown. For details on the formulation of $\Delta A_L(N_L)$, see Equation (10).

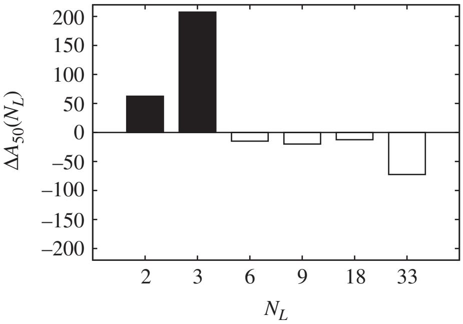

Figure 11. The acceleration of aggregation $A_{50}(N_L)$ in seconds which is the difference of TA_{50} between $N_S = 0$ and $N_S = 33$, for $N_L = 2$–33, is shown. For details on the formulation of $A_{50}(N_L)$, see Equation (11).

small disadvantage by an increasing number of 'shadow finders' and large 'light finder' populations take strong disadvantage.

4. Discussion

As shown in Figure 7, a robot swarm using the BEECLUST algorithm is predicted by our model to have an optimal swarm population of nine individuals for the tested arena size and illuminance patterns. Below this value there is a steep increase in the aggregation performance. This could be caused by positive feedback loops, which are typical for swarm behaviour: the more individuals, the better is the performance of the whole swarm. At a swarm population above the optimum of nine individuals, the aggregation performance decreases again. On the one hand , the robots block each other because the robot density gets too high for the arena size; on the other hand, the light spot gets saturated. We expect a different finding in a differently dimensioned environment because the optimal population size is dependent on the robot density.

Concerning the robustness of the swarm, we found that small 'light finder' populations gain benefit from increasing numbers of 'shadow finders' in aggregation quality as

Downloaded by [K F Univ Graz], [Michael Bodi] at 07:39 12 September 2011

well as in their aggregation speed. This results from a higher total number of robots in the arena; more collisions happen and lead to an increasing frequency of light measurements. Medium populated swarms act robust in the presence of another robot swarm concerning their ability to aggregate as well as concerning their aggregation speed. Both of these observables are affected only slightly. The advantage of more measurements and the disadvantage of jamming cancel each other out. Large swarms take disadvantage from an increasing number of other robots, because the robot density gets too high and jamming effects occur. The appearance of advantages in aggregation caused by other robots in the arena is caused by the increase of robot collisions which go along with a higher robot density in the arena. This leads to an increasing number of light measurements due to an increasing number of robot collisions. But if the density of robots in the arena is too high, jamming effects occur. This leads to a lower quantity of aggregation (Figures 8 and 10) and lower aggregation speed (Figures 9 and 11).

With the work at hand we show that the BEECLUST algorithm is not only capable of working robustly in a homogeneous robot swarm, but also suitable in heterogeneous robot groups with differing controller strategies, even without discrimination of caste affiliation of other robots. Those gained benefits allow us to use the BEECLUST algorithm in environments which may demand heterogeneous controller strategies, for example, loading energy: Robots with low energy aggregate at the fuelling station, whereas robots with enough energy accomplish their mission.

The topic of aggregation of swarms of agents is a common benchmark in the field of swarm robotics: Dorigo *et al.* [23] used an evolving neural network consisting of 12 neurons for robot aggregation. Compared to this work the BEECLUST algorithm is quite simple, because it gets along with just a few basic rules and therefore it needs less processing power and memory. Other experiments with cockroach-like robots have been performed in simulations as well as in real world [29,30]. The main difference to our work presented here is the ability of cockroach-like robots to measure the number of other nearby robots. This requires a unique ID and communication between the individuals. Our robots, which use the BEECLUST algorithm, neither need to discriminate between different individuals within the swarm nor need to know how many individuals are located in a nearby aggregation spot. Furthermore, the ‘cockroach-derived algorithm’ is aimed to emulate cockroach behaviour as closely as possible while the BEECLUST is aimed to collectively choose an optimal aggregation site. Therefore, the ‘cockroach-derived algorithm’ is more complex than the BEECLUST.

In summary, the BEECLUST algorithm is simple and therefore it has little requirements. It does not require any communication in a biological sense. The robots are able to distinguish other robots from obstacles in the environment via their physical properties and characteristics, but they do not need to know ‘which’ and how many other robots are around. Above all, our study presented here shows how robust the BEECLUST algorithm works and how versatile its applicability is despite its simplicity.

In future we will investigate how the distribution of light in the arena (gradient or discrete) influences the aggregation behaviour. Furthermore, we will investigate how the distance between the two optima influences the aggregation behaviour.

Acknowledgements

We thank Sibylle Hahshold and Gerald Radspieler for providing us with the data of honeybee experiments and for building the honeybee temperature arena. This work was supported by EU-IST-FET project SYMBRION, no. 216342; EU-ICT project ‘REPLICATOR’, no. 216240; and Austrian Science Fund (FWF) research grants P15961-B06 and P19478-B16.

References

[1] B.R. Johnson, *Organization of work in the honeybee: A compromise between division of labour and behavioural flexibility*, Proc. R. Soc. Lond. B 270 (2003), pp. 147–152.

[2] G. Beni, *From swarm intelligence to swarm robotics*, in *Proceedings of the First International Workshop on Swarm Robotics*, E. Şahin and W.M. Spears, eds., LNCS, Santa Monica, CA, 2005, pp. 1–9.

[3] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*, Oxford University Press, Oxford, 1999.

[4] J. Kennedy and R.C. Eberhart, *Swarm Intelligence*, Morgan Kaufmann Series in Evolutionary Computation, Morgan Kaufmann, San Francisco, CA, 2001.

[5] S. Camazine, J.L. Deneubourg, N.R. Franks, J. Sneyd, G. Theraulaz, and E. Bonabeau, *Self-Organization in Biological Systems*, Princeton University Press, Princeton, NJ, 2001.

[6] M.M. Millonas, *Swarms, phase transitions, and collective intelligence*, in *Artificial Life*, C.G. Langton III, ed., Addison-Wesley, Reading, MA, 1994, pp. 417–445.

[7] E. Şahin, *Swarm robotics: From sources of inspiration to domains of application*, in *Swarm Robotics: State-of-the-Art Survey*, E. Şahin and W.M. Spears, eds., LNCS 3342, Springer-Verlag, Berlin, 2005, pp. 10–20.

[8] T.D. Seeley, S. Camazine, and J. Sneyd, *Collective decision-making in honey bees: How colonies choose among nectar sources*, Behav. Ecol. Sociobiol. 28 (1991), pp. 277–290.

[9] T. Schmickl and K. Crailsheim, *Costs of environmental fluctuations and benefits of dynamic decentralized foraging decisions in honey bees*, Adapt. Behav. Anim. Anim., Software Agents Rob. Adapt. Syst. 12 (2004), pp. 263–277.

[10] R. Jeanson, C. Rivault, J.L. Deneubourg, S. Blanco, R. Fournier, C. Jost, and G. Theraulaz, *Self-organized aggregation in cockroaches*, Anim. Behav. 69 (2005), pp. 169–180.

[11] G. Theraulaz, E. Bonabeau, S.C. Nicolis, R.V. Sole, V. Fourcassie, S. Blanco, R. Fournier, J.L. Joly, P. Fernandez, A. Grimal, P. Dalle, and J.-L. Deneubourg, *Spatial patterns in ant colonies*, Proc. Natl. Acad. Sci. U.S.A. 99 (2002), pp. 9645–9649.

[12] T. Schmickl, H. Hamann, H. Wörn, and K. Crailsheim, *Two different approaches to a macroscopic model of a bio-inspired robotic swarm*, Rob. Auton. Syst. 57 (2009), pp. 913–921.

[13] C.W. Reynolds, *Flocks, herds, and schools*, Comput. Graph. 21 (1987), pp. 25–34.

[14] T. Schmickl and H. Hamann, *BEECLUST: A swarm algorithm derived from honeybees*, in *Bioinspired Computing and Communication Networks*, Y. Xiao and F. Hu, eds., Routledge, New York, 2011, pp. 95–137.

[15] T. Schmickl, R. Thenius, C. Möslinger, G. Radspieler, S. Kernbach, and K. Crailsheim, *Get in touch: Cooperative decision making based on robot-to-robot collisions*, Auton. Agent Multi Agent Syst. 18 (2008), pp. 133–155.

[16] S. Kernbach, R. Thenius, O. Kornienko, and T. Schmickl, *Re-embodiment of honeybee aggregation behavior in an artificial micro-robotic swarm*, Adapt. Behav. 17 (2009), pp. 237–259.

[17] H. Hamann, T. Schmickl, H. Wörn, and K. Crailsheim, *Analysis of emergent symmetry breaking in collective decision making*, Neural Comput. Appl. (2011), pp. 1–12.

[18] H. Heran, *Untersuchungen über den Termperatursinn der Honigbiene (Apis mellifica) unter besonderer Berücksichtigung der Wahrnehmung strahlender Wärme*, Z. vgl. Physiol. 34 (1952), pp. 179–206.

[19] T. Ohtani, *Spatial distribution and age-specific thermal reaction of worker honeybees*, Hum. Nature 1 (1992), pp. 11–26.

[20] M. Szopek, G. Radspieler, R. Thenius, T. Schmickl, and K. Crailsheim, *Cooperative thermotaxis in honeybees: Group decisions in a complex temperature gradient*, Apidologie 40 (2009), p. 663.

[21] J. Halloy, G. Sempo, G. Caprari, C. Rivault, and J.L. Deneubourg, *Social integration of robots into groups of cockroaches to control self-organized choices*, Science 318 (2007), pp. 1155–1158.

[22] P. Valdastri, P. Corradi, A. Menciassi, T. Schmickl, K. Crailsheim, J. Seyfried, and P. Dario, *Micromanipulation, communication and swarm intelligence issues in a swarm microrobotic platform*, Rob. Auton. Syst. 54 (2006), pp. 789–804.

[23] M. Dorigo, V. Trianni, E. Sahin, R. Groß, T.H. Labella, G. Baldassarre, S. Nolfi, J.-L. Deneubourg, F. Mondada, D. Floreano, and L.M. Gambardella, *Evolving self-organizing behaviors for a swarm-bot*, Auton. Rob. 17 (2004), pp. 223–245.

[24] U. Stuttgart, Available at http://www.swarmrobot.org (Accessed 15 April 2011).

Downloaded by [K F Univ Graz], [Michael Bodi] at 07:39 12 September 2011

[25] SYMBRION, Project Website. Available at http://www.symbrion.eu (Accessed 15 April 2011).

[26] REPLICATOR, Project Website. Available at http://www.replicators.eu (Accessed 15 April 2011).

[27] H. Hamann, H. Wörn, K. Crailsheim, and T. Schmickl, *Spatial macroscopic models of a bio-inspired robotic swarm algorithm*, IEEE/RSJ 2008 International Conference on Intelligent Robots and Systems (IROS'08), IEEE Press, Los Alamitos, CA, 2008, pp. 1415–1420.

[28] U. Wilensky, *NetLogo, Center for Connected Learning and Computer-Based Modeling*, Northwestern University, Evanston, IL, 1999.

[29] S. Garnier, C. Jost, R. Jeanson, J. Gautrais, M. Asadpour, G. Caprari, and G. Theraulaz, *Collective decision-making by a group of cockroach-like robots*, Proceedings of the 2nd IEEE Swarm Intelligence Symposium (SIS), IEEE, Los Alamitos, CA, 2005, pp. 233–240.

[30] S. Garnier, F. Tache, M. Combe, A. Grimal, and G. Theraulaz, *Alice in pheromone land: An experimental setup for the study of ant-like robots*, Swarm Intelligence Symposium (SIS 2007), IEEE Press, Los Alamitos, CA, 2007, pp. 37–44.

Downloaded by [K F Univ Graz], [Michael Bodi] at 07:39 12 September 2011