HAL open science

The k-Unanimity Rule for Self-Organized Decision-Making in Swarms of Robots

Alexander Scheidler, Arne Brutschy, Eliseo Ferrante, Marco Dorigo

▶ **To cite this version:**

Alexander Scheidler, Arne Brutschy, Eliseo Ferrante, Marco Dorigo. The k-Unanimity Rule for Self-Organized Decision-Making in Swarms of Robots. IEEE Transactions on Cybernetics, 2016, 46 (5), pp.1175-1188. 10.1109/TCYB.2015.2429118 . hal-01403718

HAL Id: hal-01403718
https://hal.science/hal-01403718

Submitted on 6 Dec 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Université Libre de Bruxelles

Institut de Recherches Interdisciplinaires et de Développements en Intelligence Artificielle

The k-Unanimity Rule for Self-Organized Decision Making in Swarms of Robots

Alexander Scheidler, Arne Brutschy, Elisseo Ferrante and Marco Dorigo

IRIDIA – Technical Report Series

Technical Report No.
TR/IRIDIA/2011-023

October 2011

IRIDIA – Technical Report Series
ISSN 1781-3794

Published by:
IRIDIA, *Institut de Recherches Interdisciplinaires et de Développements en Intelligence Artificielle*
Université Libre de Bruxelles
Av F. D. Roosevelt 50, CP 194/6
1050 Bruxelles, Belgium

Technical report number TR/IRIDIA/2011-023

The information provided is the sole responsibility of the authors and does not necessarily reflect the opinion of the members of IRIDIA. The authors take full responsibility for any copyright breaches that may result from publication of this paper in the IRIDIA – Technical Report Series. IRIDIA is not responsible for any use that might be made of data appearing in this publication.

Noname manuscript No.
(will be inserted by the editor)

The k-Unanimity Rule for Self-Organized Decision Making in Swarms of Robots

Alexander Scheidler · Arne Brutschy · Elisseo Ferrante · Marco Dorigo

Received: date / Accepted: date

Abstract

In this paper we propose a collective decision making method for swarms of robots. The method enables a robot swarm to select, from a set of possible actions, the one that has the fastest mean execution time. A central aspect of the proposed method is the fact that consensus on the fastest action emerges from the initially heterogeneous beliefs of the robots. We study two analytical models of the proposed decision making method to understand the dynamics of the consensus formation process. Moreover, we verify the applicability of the method in a real swarm robotics scenario. To this end, we conduct three sets of experiments that show that a robotic swarm can collectively select the shortest of two paths. Finally, we use a Monte Carlo simulation model to study and predict the influence of different parameters on the accuracy of the method.

1 Introduction

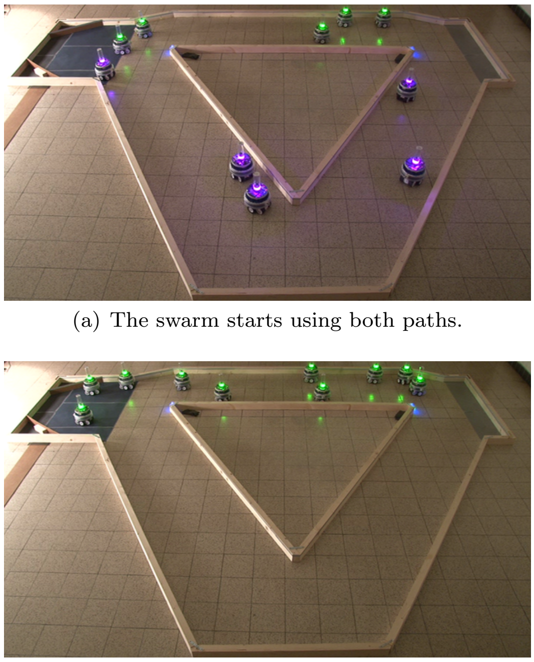
Swarm robotics deals with the control of large groups of relatively simple robots so that they perform tasks that go beyond their individual capabilities (Dorigo and Sahin, 2004; Şahin, 2005). The interactions among the robots in a swarm robotics system are based on simple behavioral rules that exploit only local information. The lack of global knowledge or of a central controller imposes challenging problems for the design of such a system. One important research question is how robot swarms can make intelligent decisions. The need to make good decisions arises in many applications of swarm robotics systems. For example, robot swarms need to select the best location containing objects to be retrieved (Gutiérrez et al., 2009), or they need to decide if a certain subtask is finished (Parker and Zhang, 2010), or they need to select the shortest between a set of paths from a source to a destination (Ducatelle et al., 2010, 2011).

When engineering decision making methods for swarm robotic systems, the distributed nature of swarms must be taken into account. Good decisions must be achieved using only local interactions among the robots. Moreover, the decision making methods must be efficient, robust with respect to robot failures, and scale well with the swarm size. Natural swarms have been a major source of inspiration for the design of collective decision making methods that possess these desired properties. Particularly, social insects are known for their ability to make good collective decisions in a self-organized way. For example, ants are able to decide on the shortest path from their nest to a food source (Goss et al., 1989) and bees are able to select a good nest site from a number of candidates (Seeley, 2010; Diwold et al., 2011). The underlying mechanisms are very different from each other but share a similarity, namely, that they are based on positive feedback. Positive feedback is a mechanism by which good decisions are reinforced more than bad ones. For example, ants find short paths by means of pheromones that are reinforced more on shorter paths. Bees, on the other hand, reinforce good candidate nest sites by a mechanism that recruits more scout bees for good sites.

IRIDIA, Institut de Recherches Interdisciplinaires et de Développements en Intelligence Artificielle Université Libre de Bruxelles
Av F. D. Roosevelt 50, CP 194/6
1050 Bruxelles, Belgium
E-mail: ascheidler@iridia.ulb.ac.be

In this paper, we introduce a collective decision-making method for swarms of robots that is based on positive feedback. In particular, the method enables a swarm of robots to select, from a set of possible actions, the one that has the fastest mean execution time. In the proposed method, every robot has its own *opinion* about which is the fastest action. During the course of the decision making process, robots observe the opinions of other robots and can, based on these observations, decide to change their opinion. Positive feedback eventually leads to consensus on one single opinion shared by the whole swarm. Moreover, due to a bias induced by the different execution times, with high probability the consensus is on the opinion representing the fastest action. The proposed method works in a fully self-organized and decentralized way. Moreover, there is no need to explicitly measure action execution times. The method is based solely on the local observation of the opinions of other robots. Therefore, the method can be applied in swarms of very simple robots that lack sophisticated communication capabilities.

The contributions of this paper are the following. We propose a new decision making method for swarms of robots (Section 2). We study two analytical models of the decision making method to understand the dynamics of the consensus formation process (Section 3). We verify the applicability of the method in a real application scenario where a robotic swarm has to collectively select the shortest of two paths (see Fig. 1 for a brief explanation of the scenario). Additionally, we use a Monte Carlo simulation model to study the influence of the different parameters on the decision making method. The experimental setup is presented in Section 4 and the obtained results are given in Section 5. We discuss the related literature in Section 6 and conclude the paper in Section 7.



(a) The swarm starts using both paths.

(b) The swarm found the short path.

Fig. 1 Pictures of the real robot experiments conducted to test the decision making method. The goal of the swarm is to find the shortest path between the left and the right end of the arena. The swarm must decide between the two actions "travel top path" and "travel bottom path". (a) The swarm starts with 5 robots having the opinion that the action "travel top path" takes less time while the other 5 robots have the opinion that action "travel bottom path" is the fastest. In the nest zone (black patch on the left) robots can observe the opinion of other robots and, based on their observations, switch their opinion. (b) Eventually, the swarm finds consensus on the action "travel top path", which is the fastest action and therefore corresponds to the shortest path.

2 A decision making method based on the k-Unanimity rule

In this paper we consider the following problem. Let a swarm of N robots be capable of executing two different actions A or B. Each action takes a stochastic amount of time to be executed. The overall goal of the swarm is to collectively choose the action that has the fastest mean execution time. We call this action the *fastest action*. Without loss of generality, in the reminder of the paper action A is the fastest action. Note that in this paper we only consider swarms that can choose between two actions.[1]

Every robot executes the action it believes to be the fastest action. This belief is called the *opinion* of the robot. Opinions are denoted with the same letters as the corresponding actions, that is, opinion A corresponds to action A and opinion B corresponds to action B.

Robots can observe the opinion of other robots when they meet. The robots store the latest k observed opinions in their memory. That is, if a robot observes an opinion, it removes the oldest opinion from it's memory and replaces it with the new opinion. Moreover, in between two action executions, robots can switch their opinion according to the so-called *k-Unanimity rule*, defined as follows:

[1] Simulation experiments suggest that the method also works for larger sets of actions without modifications.

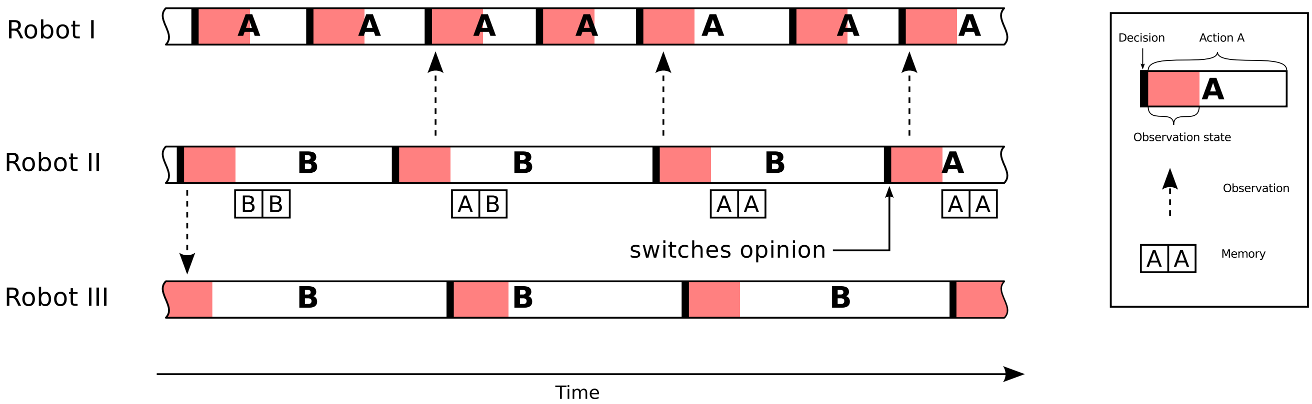


Fig. 2 Example observation and opinion switching from the point of view of Robot II. Robot I has opinion A and Robot II and Robot III have opinion B. Action B takes longer to be executed than action A on average. Due the fixed time length of the observation state, Robot I is in observation state more often than Robot III. Robot II switches opinion to A after observing A $k = 2$ times in a row.

> A robot switches to opinion X if and only if all k observations stored in its memory are of opinion X.

Applied repeatedly over time, the k-Unanimity rule eventually drives the swarm to consensus, that is, to a state in which all robots have the same opinion. If consensus is reached, no robot can change its opinion any further and we say that the swarm completed the decision making process. The k-Unanimity rule leads to consensus because it induces positive feedback on the opinion that is in the majority. For example, if opinion B is held by most robots, then it is more likely that another robot switches from A to B than that a robot switches from B to A. Consequently, with high probability, the swarm moves towards consensus on opinion B. In the next section we study this property of the k-Unanimity rule by means of an analytical model and calculate the exact switching probabilities depending on the number of robots with the different opinions.

We assume that the robots have no a priori knowledge about the execution times of the two actions. The swarm starts initially unbiased, that is, both opinions are held by an equal number of robots. This means that there is no majority opinion that can be amplified by the k-Unanimity rule. Instead, because both opinions are observed with the same probability, the opinion changes of the robots are random. However, due to this random opinion switches the symmetry between the two opinions is broken in favor of one opinion and eventually the k-Unanimity leads to consensus. Consequently, unbiased swarms find consensus on a random opinion.

However, our goal is not achieving consensus on a random opinion but on the opinion that is associated with the fastest action. We therefore introduce a mechanism to break the symmetry between the two opinions in favor of the opinion associated with the fastest action. The symmetry breaking mechanism is based on the introduction of a so-called *observation state*. The observation state restricts the observation possibilities of the robots. In particular, robots are only allowed to observe and to be observed when they are in the observation state. Robots enter the observation state only once per execution of an action. Moreover, the duration of the observation state is fixed and is the same for all robots regardless of their opinion. Due to the fixed length of the observation state, robots executing the fastest action are in observation state more often (see the example given in Fig. 2). Since robots with opinion A spend a larger fraction of their time in the observation state, the probability to observe opinion A over opinion B is higher. In (initially) unbiased swarms this breaks the symmetry between the two opinions in favor of opinion A. Consequently, with higher probability, the system will evolve consensus on opinion A.

Note that the proposed method does not guarantee that always the fastest action is found. Swarms might still reach consensus on the slowest action B. This is due to the inherent randomness of encounters and the resulting opinion switches. Particularly, at the start of the decision making process, it can happen that, by chance, the number of robots with opinion B becomes larger than the number of robots with opinion A. This results in an increased probability to observe opinion B. If the number of robots with opinion B exceeds a certain level, the probability to observe B will be higher than the probability to observe A. Consequently, in this case, the swarm will find consensus on B than on A with higher probability. (In the next section we will determine the critical level at which the swarm might decide on the slowest action B.)

3 The analytical model

In the following, we analytically study the dynamics of the decision making process induced by the k-Unanimity rule. The investigated theoretical models help to understand the dynamics of the decision making method and predict its behaviour for idealized conditions, that is, in the absence of noise induced by imprecision of sensors or robot failures.

Consider a swarm of N robots that use a memory of size k. We denote the number of robots with opinion A at a given time by n. Let $x = n/N$ be the fraction of these robots. Without loss of generality the average execution time of action A is assumed to be 1. The average duration of action B is λ with $\lambda \geq 1$, that is, action B takes at least as long as action A. It follows that, within a unit time, on average n robots with opinion A and $(N-n)/\lambda$ robots with opinion B finish their actions. The probability that a robot finishing an action has opinion A is thus given by:

$$p = \frac{n}{n + (N-n)/\lambda} = \frac{x}{x + (1-x)/\lambda} \ . \tag{1}$$

In our model, robots apply the k-Unanimity rule directly after the execution of an action. Subsequently, they immediately start a new action. As such, we do not model the observation state explicitly. Moreover, we assume that the observation memory is always filled with opinions sampled accordingly to the actual rates at which the robots finish their actions. More precisely, we assume that each of the memorized opinions equals opinion A with probability p and opinion B with probability $(1-p)$. Note that this way of modeling neglects the fact that in a real robot swarm the stored opinions in a robot's memory might have been observed at different times (between several action executions). However, since the latest observation is removed when a new observation is made, the stored opinions at a certain time must have been observed in the near past and represent a snapshot of the current rates at which the robots finish their actions. Hence, the simplification that robots observe k new opinions at once after every action execution is valid for our analytical model.

If a robot applies the k-Unanimity rule, at the level of the swarm this can have three different effects. In particular, the number of robots with opinion A can be increased by one, can be decreased by one, or no change can occur. We now determine the probabilities for these three events. The overall number of robots with opinion A increases if a robot with opinion B observes opinion A k times in a row. The probability that a robot observes opinion A k times is given by p^k and the probability that a robots that applies the k-Unanimity rule has opinion B is $1-p$. Hence, $w^+ = (1-p)p^k$ gives the probability that the application of the k-Unanimity rule increases the number of robots with opinion A. Similarly, the probability that the number of robots with opinion A is decreased is $w^- = p(1-p)^k$. The probability that the number of robots with opinion A does not change upon an application of the k-Unanimity rule is $w^* = 1 - w^+ - w^-$.

The calculation of the probabilities w^+ and w^- reveals why the k-Unanimity rule for $k \geq 2$ is able to amplify an existing opinion bias and to eventually lead to consensus on one opinion. If, for example, opinion A has a higher probability to be observed than opinion B (i.e., $p > 0.5$) it follows that $w^+ > w^-$. This means that if we pick a random robot that applies the k-Unanimity rule, the probability that this robot switches from B to A is higher than the probability that it switches from A to B. Therefore, the number of robots with opinion A increases with higher probability upon every application of the k-Unanimity rule. Consequently, the swarm is driven to consensus on A.

3.1 Continuum model

In this section we study the dynamics of consensus formation by means of a continuum model. This means that we model how the average fraction of robots that prefer opinion A evolves over time. We do not take a specific swarm size N into account. Recall that within a unit time $n + (N-n)/\lambda$ robots finish their actions and apply the k-Unanimity rule. This corresponds to a fraction of $x + (1-x)/\lambda$ of the swarm. We can thus model the evolution of the expected fraction of robots with opinion A as

$$\dot{x} = (w^+ - w^-)[x + (1-x)/\lambda] \ . \tag{2}$$

Fig. 3 visualizes Equation (2). It shows for a given fraction x of robots with opinion A the expected change within a unit time. The zeros of $\dot{x}$, that is, the stationary solutions of Equation (2), are the (stable) consensus states $[x = 0]$ and $[x = 1]$ and the (unstable) equilibrium point $[x = 1/(1+\lambda)]$. The latter marks the *critical fraction* that separates the flow to the consensus states. We denote the critical fraction by x_c. Note that x_c depends only on λ. The model predicts that if the fraction x of robots with opinion A exceeds x_c, then x will steadily increase until consensus on A is reached

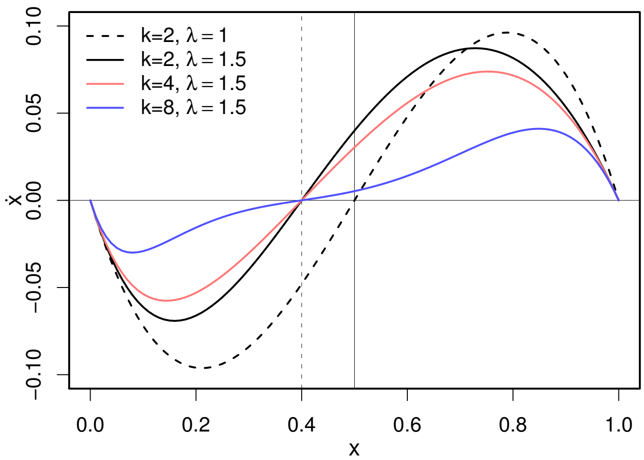


Fig. 3 Rate of change of the fraction of robots with opinion A. The vertical solid line marks unbiased swarms ($x = 0.5$). The dashed line marks the critical fraction $x_c = 0.4$ for $\lambda = 1.5$.

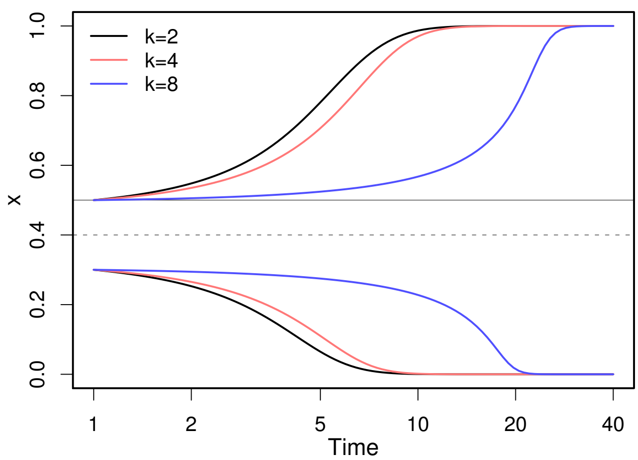


Fig. 4 Example trajectories of the continuum model (2) for $\lambda = 1.5$ and different values of memory size k. The upper trajectories start at $x = 0.5$ and the lower trajectories start at $x = 0.3$. The dashed line marks the critical fraction $x_c = 0.4$.

($\dot{x} > 0$ for $x > x_c$). For $x < x_c$ consensus on B will be found. Hence, as expected, the k-Unanimity rule induces positive feedback and amplifies an existing bias in the observed opinions. Note that the critical fraction x_c marks the state in which the probability to observe opinion A is $p = 0.5$. In other words, at the critical fraction both actions are executed at the same rates and there is no bias in the observed opinions. Beside the critical fraction x_c the point $x = 0.5$ is of particular interest. It corresponds to a swarm in which both opinions are present in equal proportions. We call a swarm in this state unbiased. The vertical solid line in Fig. 3 marks this point.

For equal execution times ($\lambda = 1$, Fig. 3, dashed line), the critical fraction is $x_c = 0.5$. If the system moves from the critical fraction towards the consensus states the rate of change $\dot{x}$ first increases. This is because the probability increases to observe the opinion that is in the majority k times in a row. However, near the consensus states the rate of change decreases as the number of remaining robots that have to be convinced becomes smaller.

If action B takes longer than action A ($\lambda > 1$) the critical fraction is shifted towards smaller values. For example, for $\lambda = 1.5$ the critical fraction is $x_c = 0.4$ (vertical dashed line). However, it still holds that for any $x > x_c$ consensus is found on A. Consequently, the model predicts that unbiased swarms (i.e., $x = 0.5$) always find consensus on A, the fastest action.

For larger values of k (memory size) the rate of change $\dot{x}$ decreases. Therefore, swarms need more time to converge for larger k. Moreover, for larger k, the shape of the curves changes. More precisely, the rate of change approaches 0 near the critical fraction. The reason is that the larger k the harder it is to observe the same opinion k times in a row. This is particularly the case near the critical fraction ($p \approx 0.5$).

In Fig. 4 example trajectories for $\lambda = 1.5$ and different memory sizes k are depicted. The upper trajectories start at $x = 0.5$ (which corresponds to unbiased swarms). As already explained, unbiased swarms converge to consensus on A. Furthermore, the time it takes to converge grows rapidly with k (note that the x-axis is log-scaled). The bottom trajectories start at $x = 0.3$. They show that if too few robots have opinion A the swarm finds consensus on B. The dashed vertical line marks the critical fraction $x_c = 0.4$. The continuum model predicts that a swarm that starts at this point will not develop consensus. However, in contrast to this prediction, swarms of finite size must eventually reach consensus since the two consensus states are the only absorbing states. In the next section we propose a model that takes this consideration into account.

3.2 Master equation approach

The continuum model only captures the average behaviour of the swarm. This is due to the fact that continuum models investigate how the fractions of robots in the different states evolve. As they consider swarms of infinite number of robots, continuum models often help to understand certain macroscopic properties of the swarm behavior. However, they fail to capture effects caused by fluctuations due to random decisions of robots in finite swarms. For example, for a given initial fraction x of robots with opinion A our continuum model predicts only one of the two consensus states as final outcome. However, in real swarms as long as both opinions are present both consensus states are still reachable.

In the following, we propose a model of our decision making method that also takes into account the fluctuations that occur in finite swarms. For a given n (current number of robots with opinion A) we define E_n as the probability to eventually reach consensus on A. Note that although the evolution of a swarm towards consensus is a dynamic process, there is no time dependence in E_n.

In the following we estimate the N probabilities E_n. If n robots have opinion A, due to the application of the k-Unanimity rule the number of robots with opinion A might increase to $n+1$. The probability of this event is w^+. Consequently, the probability to find consensus on A if the application of the k-Unanimity rule increases the number of robots with opinion A from n to $n+1$ is given by w^+E_{n+1}. Considering also the two remaining outcomes of the application of the k-Unanimity rule we obtain a so-called master equation:

$$E_n = w^+E_{n+1} + w^-E_{n-1} + w^*E_n \ . \tag{3}$$

Solving this master equation would mean to derive a non-recursive, closed form for E_n. However, it is much easier to approximate its solution by a continuous function $E(x)$. This function is defined for $x \in [0, 1]$ and at the points $x = n/N$ ($n \in [0, \dots, N]$) its value is $E(x) = E_n$. We rewrite the master equation in terms of $E(x)$ and do a second order Taylor expansion:

$$\begin{aligned} E(x) &= w^+E\left(\frac{n+1}{N}\right) + w^-E\left(\frac{n-1}{N}\right) + w^*E\left(\frac{n}{N}\right) \\ &= w^+E(x+1/N) + w^-E(x-1/N) + w^*E(x) \\ &= w^+\left[E(x) + \frac{1}{N}\frac{\partial E(x)}{\partial x} + \frac{1}{2}\frac{1}{N^2}\frac{\partial^2 E(x)}{\partial x^2}\right] + \\ &\quad w^-\left[E(x) - \frac{1}{N}\frac{\partial E(x)}{\partial x} + \frac{1}{2}\frac{1}{N^2}\frac{\partial^2 E(x)}{\partial x^2}\right] + \\ &\quad w^*E(x) \ . \end{aligned} \tag{4}$$

Because $w^+ + w^- + w^* = 1$ the term $E(x)$ can be eliminated in (4) and we derive the second order differential equation

$$\begin{aligned} 0 &= \left[w^+ - w^-\right]\frac{1}{N}\frac{\partial E(x)}{\partial x} + \left[w^+ + w^-\right]\frac{1}{2}\frac{1}{N^2}\frac{\partial^2 E(x)}{\partial x^2} \\ &= 2N\left[\frac{(1-p)^{k-1} - p^{k-1}}{(1-p)^{k-1} + p^{k-1}}\right]\frac{\partial E(x)}{\partial x} + \frac{\partial^2 E(x)}{\partial x^2} \ . \end{aligned} \tag{5}$$

Clearly, if all robots have opinion B then the probability to reach consensus on A is zero ($E_0 = 0$). On the other hand, if all robots have opinion A then the probability to converge to opinion A is one ($E_N = 1$). This defines the boundary conditions $E(0) = 0$ and $E(1) = 1$ for our approximation.

We can also model the expected time T_n until convergence. This is the time that a swarm of N robots in which n robots have opinion A needs to reach consensus. Recall that within a unit time $n+(N-n)/\lambda$ robots finish their actions. We can hence determine the expected time between two robots finishing their action (between two applications of the k-Unanimity rule) as

$$\delta t = \frac{1}{n + (N-n)/\lambda} = \frac{p}{xN} \ . \tag{6}$$

The master equation for the time until consensus is then given by

$$T_n = \delta t + w^+T_{n+1} + w^-T_{n-1} + w^*T_n \ . \tag{7}$$

Inserting (6) into (7) and applying the same steps as used to derive Equation (5) now leads to:

$$\begin{aligned} 0 = &\frac{2Np}{x[p(1-p)^k + (1-p)p^k]} + \\ &2N\left[\frac{(1-p)^{k-1} - p^{k-1}}{(1-p)^{k-1} + p^{k-1}}\right]\frac{\partial T(x)}{\partial x} + \frac{\partial^2 T(x)}{\partial x^2} \ . \end{aligned} \tag{8}$$

Clearly, if one of the two consensus states is reached the time to convergence is zero. Hence, the boundary conditions for the approximation of T_n are $T(0) = 0$ and $T(1) = 0$.

Fig. 5 shows solutions for $E(x)$ and $T(x)$ with respect to the given boundary conditions for $\lambda = 1$ and $\lambda = 1.5$ and different values of memory size k and swarm size N. Again the critical fraction x_c and the fraction $x = 0.5$ that relates to unbiased swarms are of particular interest.

Recall that for equal action execution times ($\lambda = 1$) the critical fraction is $x_c = 0.5$. The continuum model predicts that for $x > x_c$ consensus is found on A (see previous section). However, this only holds on average. As long as consensus is not reached there is a nonzero probability for both consensus states. As can be seen in Fig. 5a, even if robots with opinion A are in the majority, there is still a certain probability to reach consensus on B. For example, if in a swarm of 10 robots 6 robots prefer A (that is, $x = 0.6$) the probability that consensus is reached on B is still $E(0.6) \approx 0.28$. However, for larger swarms the probability to reach consensus on

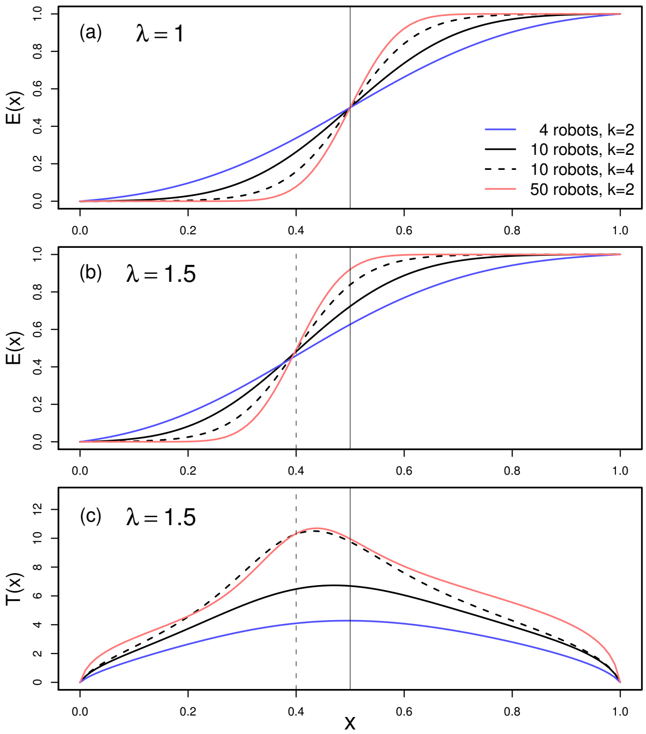


Fig. 5 Influence of the initial fraction x of robots that start with opinion A. (a) Approximation of $E(x)$, the probability to find consensus on opinion A, for equal action execution times ($\lambda = 1$). (b) $E(x)$ if action B takes longer than action A ($\lambda = 1.5$) (c) Expected time to consensus for $\lambda = 1.5$

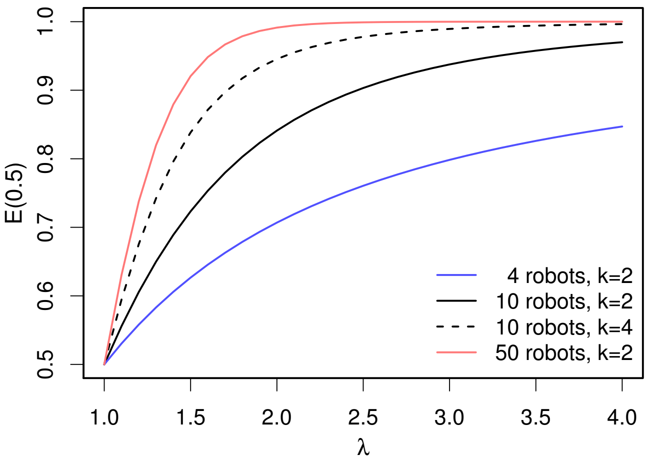


Fig. 6 Approximation of the probability $E(x)$ to converge to opinion A depending on λ for different memory sizes k and different swarm sizes N

the minority opinion becomes smaller ($E(0.6)$ drops to 0.09 for $N = 50$). For $N \to \infty$ the function $E(x)$ converges to a step function and even very small deviations towards one opinion are amplified and result in consensus on this opinion with high probability. Larger values for the memory size parameter k result in stronger positive feedback. For example, if k is increased from 2 to 4, the probability for consensus on B for 10 robots and $x = 0.6$ decreases from 0.28 to 0.17.

More interesting than the symmetric case are asymmetric execution times. Recall, if action B takes longer than action A ($\lambda = 1.5$, Fig. 5b), the critical fraction at which the final decisions is random is shifted towards smaller values ($E(x_c) = 0.5$ for $x_c = 0.4$). The shift of x_c has the consequence that swarms that start unbiased ($x = 0.5$) find consensus on action A with higher probability. The steepness of $E(x)$ near x_c is determined by the swarm size N as well as by the memory size k. More precisely, larger swarms and larger values for k lead to higher probability of finding consensus on A. Fig. 5c depicts the expected time that swarms need to find consensus $T(x)$. Near the critical fraction, swarms need longest to converge. Here, the probability that an application of the k-Unanimity rule increases the number of robots that prefer action A is similar to the probability that the number is decreased ($w^+ \approx w^-$). Therefore, the drift toward a consensus state is small. The time it takes for the swarm to converge to a decision is influenced by the swarm size N and the memory size k.

For what concerns our application of finding the fastest action, we are typically not interested in swarms that start biased. Instead, we start with initially unbiased swarms that then use the k-Unanimity rule to amplify the opinion bias that is induced by the unequal action execution times. In our model the unbiased swarm corresponds to $x = 0.5$. In the following, we will therefore concentrate on $E(0.5)$ and $T(0.5)$.

Fig. 6 presents the probability for consensus on action A dependent on the mean execution time of action B. Clearly, if both action execution times are equal the probability to find consensus on A is 0.5. Moreover, as expected, the larger the difference between the action execution times, the higher is the probability that the swarm converges to opinion A.

Fig. 7 depicts the time to convergence $T(0.5)$ versus the probability to converge to action A $E(0.5)$ for different swarm sizes $N \in \{4, 10, 50\}$ and different memory sizes $k \in \{2, \ldots, 8\}$. It can be seen that the model predicts a trade-off between the probability to converge to the action with the fastest execution time and the time the swarms need to reach the decision. Note that increasing the memory size k always increases the probability to find consensus on the fastest action. Moreover, for $\lambda > 1$ and $k \to \infty$, the probability $E(0.5)$ also converges to 1. This is because the probability, that the number of robots with opinion A decreases, vanishes faster than the probability that the number of robots with opinion A increases (i.e., $w^+/(w^+ + w^-) \to 1$).

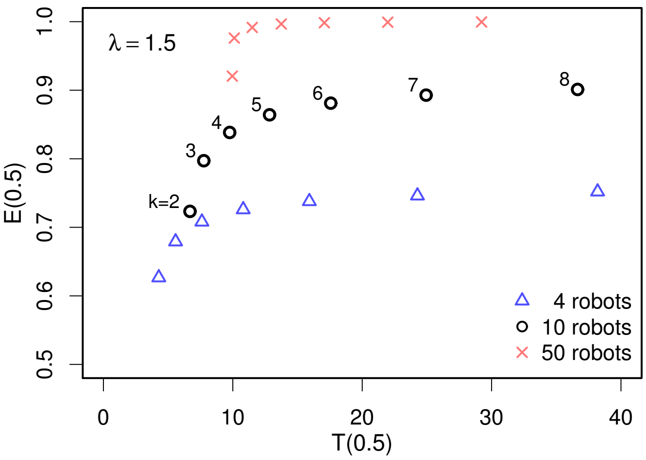


Fig. 7 Influence of memory size k. Shown is the time to convergence versus the probability to converge to action A for unbiased swarms.

However, as can be seen in Fig. 7, increasing the accuracy by increasing k quickly becomes very costly in term of convergence time. This is because the probability that any robot changes its opinion $(w^+ + w^-)$ also vanishes exponentially fast with increasing k.

4 The experimental setup

In the following, we present the setup of our real robot experiments. Moreover, we give details on the simulation model we use for further investigations of our decision making method.

The setup of our real robot experiment resembles the well-known double bridge experiment as employed by Goss et al. (1989) to show that ants are able to find the shortest path between their nest and a food source. Following the taxonomy of Winfield (2010), our experiment can be classified as a single nest, single source, homogeneous foraging task. More precisely, the robots' task is to repetitively collect objects from a *source zone* and transport them to a given *nest zone* (see Fig. 8). Since we concentrate in this paper on the decision making method, the robots only transport virtual objects. Moreover, we assume that the source zone contains an unlimited number of objects. Hence, the robots' task is to constantly travel between nest zone and source zone. The overall goal of the robot swarm is to collect as many objects as possible. The best performance can be reached when the swarm solely uses the shortest of the two paths. Note that using exclusively the shortest path is only advantageous when no strong physical interference between the robots occurs (e.g., when robots do not have to avoid each other because of a crowded condition). Indeed, large swarms might gain better performance by using both paths simultaneously, as this reduces the interference on the single paths. However, in our experiments we use a small swarm in which the effect of interference can be neglected.

The experimental area has a size of 4.5 m x 3.5 m (see Fig. 8). Three different zones are marked with colored patches on the ground. These patches let the robots determine in which zone they are. The nest zone is located in the left of the arena and the source zone is located in the right of the arena. The two zones are connected by two paths of different length. The shortest path is called "A" and the longest path is called "B". Next to the nest zone is located the so-called *observation zone*. Lights near the nest zone help the robots to navigate within the arena. Moreover, two landmarks, implemented as blue LEDs, are placed at the two bifurcations of the double-bridge.

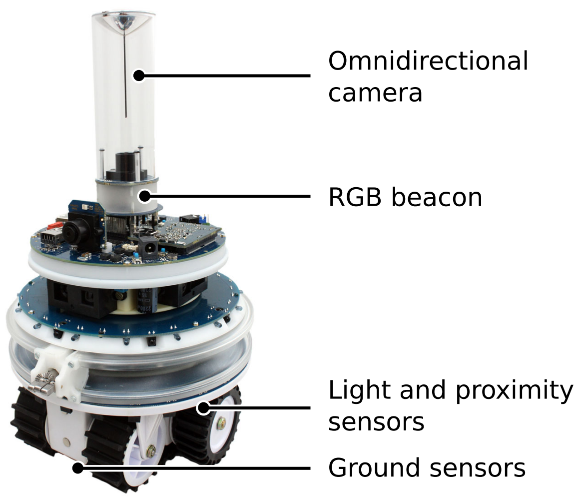


Fig. 9 Foot-bots use the RGB beacon to show their opinion. The omnidirectional camera is used to observe the opinions of other robots and to recognize the landmarks. Light and proximity sensors are used to navigate in the arena. Ground sensors are used to determine the zone in which a robot is located.

For our study we use 10 foot-bots. The foot-bot is a modular robot that was developed within the FET project *Swarmanoid* (Dorigo et al., 2011). It has a circular chassis with a diameter of 17 cm, a height of 29 cm, and a weight of 1.8 kg (see Fig. 9). The foot-bot is fully autonomous and equipped with a hot swappable battery and an on-board ARM 6 processor (i.MX31 clocked at 533 MHz and with 128 MB RAM) running a Linux-based operating system. A combination of tracks and wheels provides the foot-bot with differential drive motion capabilities. b In our experiments, we use the foot-bots' 24 IR proximity sensors to implement obstacle

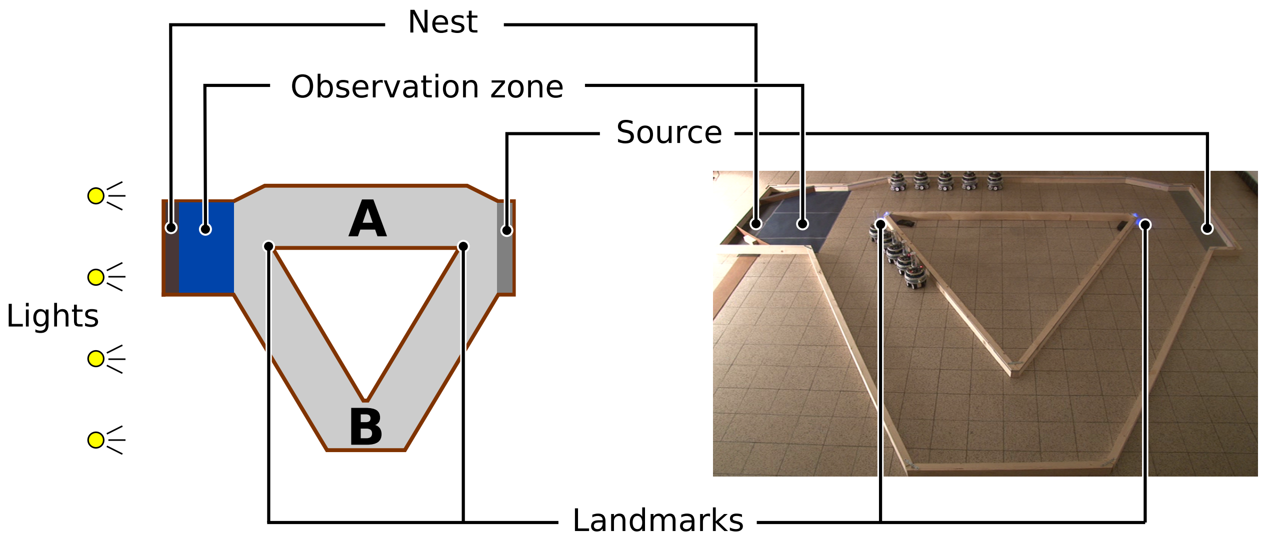


Fig. 8 Experimental setup: schematic (left) and real installation with a size of 4.5 m x 3.5 m (right). The robots constantly travel between nest zone and source zone by navigating with respect to the lights (anti-phototaxis for going to the source zone and phototaxis for going to the nest zone). Depending on their opinion they decide on which side to pass the landmarks. Robots with opinion A take path A while robots with opinion B take path B. In the observation zone robots observe each other's opinions.

avoiding behaviour, their 24 light sensors to implement light following behaviour, their 4 IR ground sensors to distinguish between the different zones in the experimental area, their RGB beacons to show the robots' current opinion and their onmidirectional 3 mega pixel camera to enable the robots to observe opinions of other robots.

4.1 Robot behaviour

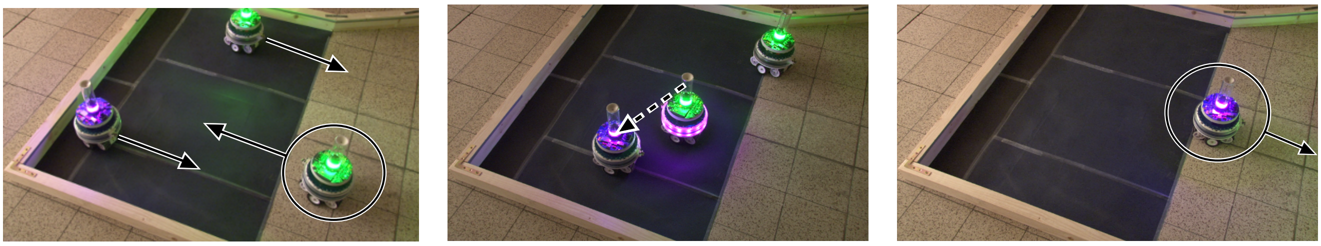
The robots have no global map of the environment and do not communicate explicitly. They navigate only with respect to the light located next to the nest zone. To get to the source zone, the robots move away from the light, that is, they perform anti-phototaxis, until they detect the source zone ground patch. When robots reach the source zone, they return to the nest zone by moving into the direction of the light, that is, by performing phototaxis.

The relation between our experimental setup and the proposed decision making method is the following. The two opinions A and B represent the two actions "travel top path" and "travel bottom path", respectively. A robot's opinion determines therefore how it navigates in the proximity of the landmarks. Robots that have opinion A and move towards the source zone try to pass the landmarks at the left hand side whereas when they go back to the nest zone they try to pass the landmarks at the right hand side. For robots with opinion B, this behaviour is mirrored accordingly. Robots are in observation state only if they are in the observation zone.

The robots use their RGB beacon to show their opinions. Robots that have opinion A light up their RGB beacon in green and robots that have opinion B light up their RGB beacon in purple. Fig. 10 illustrates different stages of the observation and decision process from the point of view of a single robot. In particular, consider the encircled robot in Fig. 10a. The robot has opinion A (green beacon—right robot). It is moving towards the nest zone and is going to enter the observation zone. As can be seen, two other robots are currently in the observation zone. One has opinion A (green beacon—top robot) and one has opinion B (purple beacon—left robot). These two robots have already visited the nest zone and are goning to leave the observation zone towards the source zone.

In the observation zone, robots try to observe another robot's opinion, that is, they use their omnidirectional camera to detect another robot's RGB beacon. If a robot recognizes multiple RGB beacons it chooses one randomly. The considered robot in our example observes opinion B from the left robot (indicated by an arrow in Fig. 10b).

The decision process for the robots works according to the method given in Section 2, that is, if a robot with opinion A (resp. B) observes opinion B (resp. A) k times in a row, it switches its own opinion. This switch is delayed until the robot leaves the observation zone. Thus, as long as a robots is located in the observation



(a) (b) (c)

Fig. 10 Illustration of the observation and decision process shown on the example of a single robot. (a) A robot with opinion A (encircled) enters the observation zone. (b) The robot observes another robot with opinion B (the robot shows this to the experimentor by flashing its LED-ring) and stores the observation in its memory. (c) The robot leaves the observation zone and the application of the k-Unanimity rule changes its opinion to B.

zone, it keeps and propagates the opinion that is associated with its last executed action. The considered example robot observed B k times in a row and switches to B when leaving the observation zone (Fig. 10c).

Note that if a robot leaves the observation zone without observing any opinion, it memorizes its own opinion, that is, it observes itself. Through simulation studies we found this rule to be superior compared to observing nothing (i.e., not modifying the memory). Moreover, note that a robot observes exactly one opinion after each action execution. In principle, a robot in the observation zone could memorize the opinions of all observable robots. This might result in a faster convergence of the system. However, to observe more than one opinion a robot must be recognize if a certain robots was already observed. This makes it necessary to introduce IDs for robots or to enable the robots to track already observed robots. Neither of these options can be realized without the use of more sophisticated hardware and/or software implementations.

Experiment	**I**	**II**	**III**
Runs	15	15	15
Memory Size k	2	2	4
Execution times ratio λ **(approx.)**	1.3	1.9	1.3

Table 1 Parameter for the real robot experiments

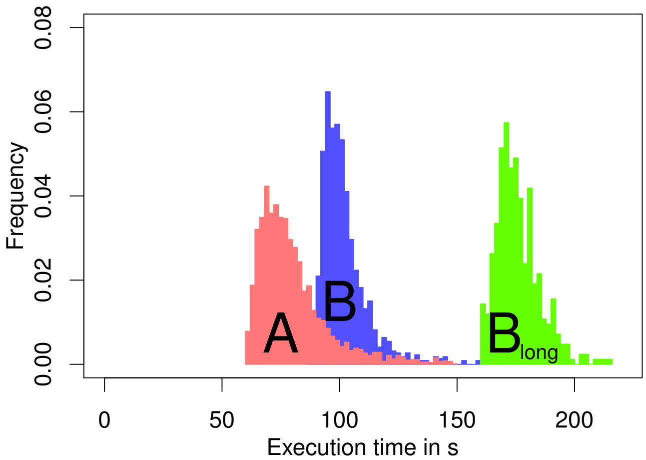


Fig. 11 Distributions of the travel times for path A, path B and path B in experiment II, recorded in the real robot experiments and used for the simulation model.

4.2 Parameters and initial conditions for the real robot experiments

The 10 robots of the swarm are divided into two groups of 5 robots each. The members of one group start with opinion A and the members of the other group start with opinion B. The robots start moving to the source zone in pairs of two robots, one for each of the two opinions. The time interval between the consecutive starts of two pairs is approximately 15 seconds. This ensures a homogeneous distribution of robots in the arena and avoids the formation of clusters of robots at the start of the experiments.

We conduct three different experiments, each consisting of 15 independent runs (see Table 1). In Experiment I the robots use a memory of size $k = 2$. In Experiment II the robots also use a memory of size $k = 2$, but we increase the difference between the execution times of the two actions by letting robots that move on path B drive with half the base velocity, thereby simulating a longer path. In Experiment III all robots move with the same velocity as in Experiment I, but we increased the size of the memory to $k = 4$.

Recall that in the analytical model the parameter λ determines the ratio between the average execution times of the two actions. Similar, the value for λ given in Table 1 corresponds to the ratio between the average execution times of the two actions in the real robot experiments (i.e., the two travel times on the two paths). The given values for λ are not explicitly defined parameters, but are determined by the arena setup. They are derived from real travel times that were collected during the real robot experiments. Fig. 11 visualizes the distributions of the collected travel times (the collected data consists of 3027 travel times for path A, 1096 travel times for path B and 418 travel times for path B in Experiment II—where robots move with half base velocity).

4.3 Simulation model

Additionally to the real robot experiments we use a Monte Carlo simulation model to investigate how the decision making mechanism performs in a wider variety of parameter setups. The simulation model is a simple event based multi agent model. This means that no representation of the physical environment nor physical interactions between robots are modeled in the simulation. Instead, the robots are represented by simple agents that either execute one of the two possible actions or are in observation state for a fixed time interval.

The execution times of the two actions in the simulation model are sampled from real travel times collected in the real robot experiments (as explained in Section 4.2).

In the simulation model, the time robots stay in the decision state is set to 20 seconds for all robots. Similar to the real robot experiments, in simulation the robots start consecutively in pairs of robots of different opinions. The time between the start of two pairs is set to 150 seconds divided by the number of simulated robots. For 10 robots this corresponds to the 15 second between the consecutive starts of the robots as used in the real robot experiments. The default value for the memory size is $k = 2$.

4.4 Performance metric

In this paper action A is always the fastest action. The outcome of a single run of a real robot experiment, as well as a simulation run, is either consensus on action A or action B. Hence, the fraction of runs that end in consensus on action A represents a measure of how well the swarm performs in finding the fastest action for a given parameter combination. We call this the *accuracy* of the decision making method.

In our simulation experiments we also test swarms that start biased, that is, swarms that start with unequal fractions of robots favoring the different opinions. For a given parameter setup we are interested in the probability that the simulated swarm converges to A. For swarms that start unbiased this probability directly relates to the accuracy of our method. In order to estimate the probability to converge to opinion A, we conduct 10 000 independent simulation runs per parameter set and calculate the fraction of runs that converge to action A. The confidence interval for 10 000 trials and 95% confidence level is $< \pm\ 0.01$. In other words, the results we present for the probability to converge to action A should be considered with an error of 1% in mind.

5 Results and discussion

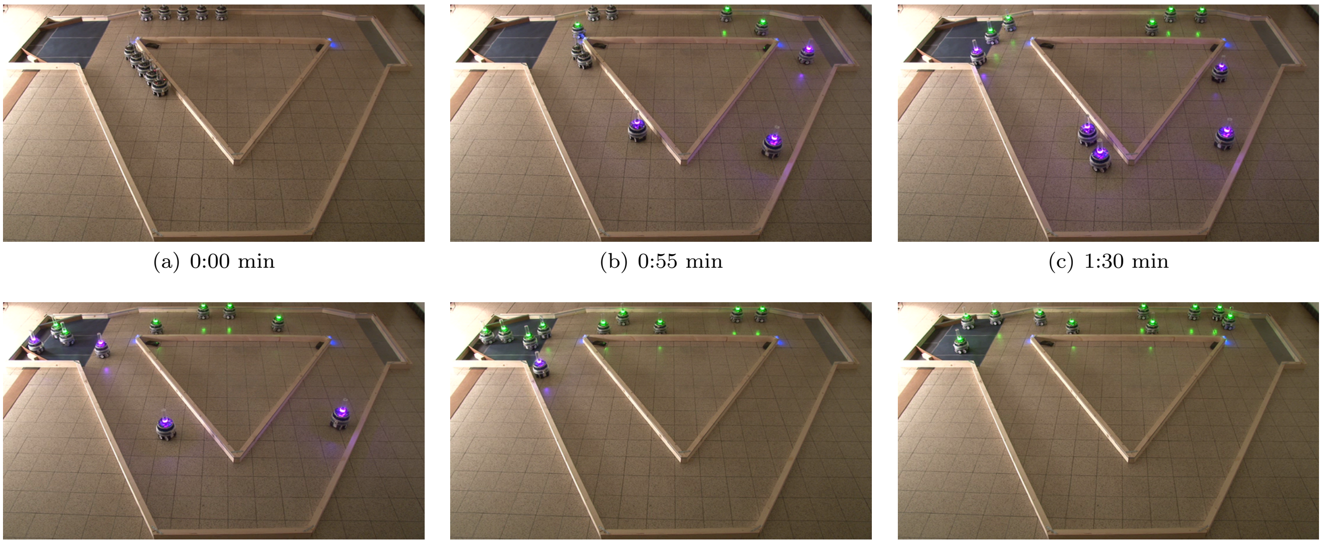
In the following, we present the results of our real robot experiments and compare them to the corresponding simulation results. We show that the simulation model resembles the real robot experiments closely. We also point out differences between the results of the real robot experiments and the simulation model, as well as discuss their causes. Thereafter, we present simulation experiments that were conducted in order to investigate system parameters that affect the performance of the decision making method but exceed the possibilities of real robot experiments.

5.1 Real robot experiments I, II, and III and comparison with the simulation model

Fig. 12 shows stages of a typical run of Experiment I[2]. At the start of a run the robots form two queues of 5 robots each, placed on the two paths (Fig. 12a). The robots start to move to the source zone in pairs—one robots out of each queue—to avoid the occurrence of robot clusters at the begin of the run (Fig. 12b). During the experimental run, robots with opinion B switch to opinion A (Fig. 12c-e). Eventually, the swarm converges to the shortest path (Fig. 12f). The depicted experimental run took 16 minutes.

A summary of the results of all experiments can be found in Fig. 13. In Experiment I (memory size $k = 2$) 10 out of 15 runs successfully converged to the short path A (accuracy 0.67). This result is in accordance with the simulation results where an accuracy of 0.68 is obtained. The time the system needs to converge to

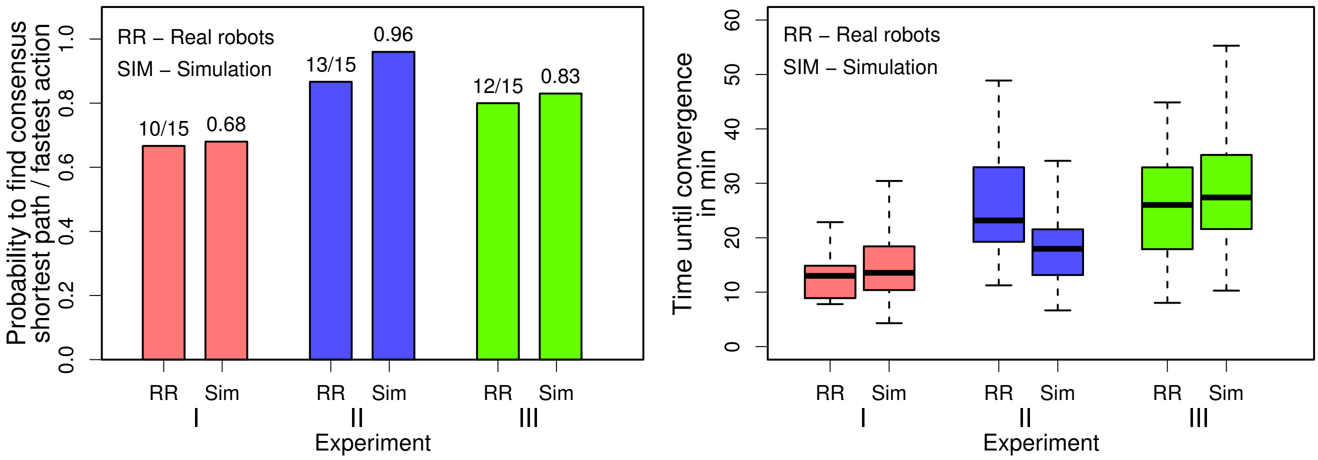
[2] See also the supplementary material at `http://iridia.ulb.ac.be/supp/IridiaSupp2011-020/` for an example video of an experimental run.



(a) 0:00 min (b) 0:55 min (c) 1:30 min

(d) 8:20 min (e) 15:45 min (f) 16:10 min

Fig. 12 An example of a typical run of an experiment. (a) The robots are equally distributed in the two paths. (b) The robots start in pairs to avoid the occurence of robot clusters at the begin of the run. (c-e) The robots successively switch to the shortest path A. (f) The swarm has converged to the shortest path A.



(a) (b)

Fig. 13 Summary of the experimental results. Experiment I: execution time ratio $\lambda \approx 1.3$ and memory $k = 2$ resulted in 10 out of 15 successful runs and runs took 15 min on average. Experiment II: increasing the execution time for B to $\lambda \approx 1.9$ led to 13 successful runs but also doubles the time needed to converge; Experiment III: increasing the the number of opinions the robots store to 4 resulted in 12 runs that converged to A and in a strongly increased convergence time

the shortest path is also predicted well by the simulation model (see Fig. 13b). Single experiments last from 9 minutes minimum up to 24 minutes maximum. On average it takes approximately 15 minutes to find the shortest path.

The goal of Experiment II is to investigate the influence that the ratio between the action execution times has on the accuracy of the method. Recall that in Experiment II we increase the difference between the execution times for the two actions by letting robots on the longer path move with only half of the base velocity. This increases the ratio between the times for the two different paths from approximately 1.3 to approximately 1.9. The analytical model predicts that a larger difference between the action execution times increases the bias in the observed opinions and helps the robots to find the fastest action more easily. The results of real robot Experiment II are in accordance with this prediction. Here 13 out of 15 runs successfully converged to the shortest path (accuracy 0.86). The simulation of Experiment II leads to a high accuracy of 0.96.

The time the swarm needs to converge in Experiment II differs between the simulation results and the real robot results (see Fig. 13b). The difference in accuracy and convergence time between simulation and real robot experiments is due to a small bias towards opinion B in the setup of the real robot experiment. Robots that take path B are slowed down immediately after they leave the observation zone and remain therefore observable by robots in the observation zone for a longer time. However, as physical interactions are not taken into account, this bias is not present in the simulation model. Hence, in the real robot implementation of Experiment II it is slightly more likely to observe a robot with opinion B compared to the simulation model. Consequently, the accuracy is lower and the time the method needs to converge is longer.

The time the swarm needs to converge in simulations of Experiment I does not differ strongly to the time the swarm needs to converge in Experiment II. The reason is that in Experiment II, due to the large difference in the action execution times, robots with opinion A mostly observe other robots with opinion A. Therefore they have a smaller probability to switch to opinion B than their counterparts in Experiment I. Indeed, considering only those experimental runs that converge to opinion A, on average only 0.88 robots switch to opinion B in Experiment II whereas in Experiment I 3.72 robots switch to opinion B on average.

Next, we investigate the influence of the memory size k. To this end, in Experiment III we increase the memory size k from 2 to 4. The results show that this increases the accuracy of the decision making method. In the real robot experiments 12 out of 15 runs converged to the shortest path (accuracy 0.8). The simulation of Experiment III shows that the accuracy is 0.83. However, as also predicted by the analytical model, the overall convergence time increases. Clearly, it takes more time to reach the required number of observations needed to switch the opinions.

Our real robot experiments show that the decision making method can also cope with the presence of errors. Such errors occur due to different reasons. For example, robots sometimes happen to take the wrong path with respect to their opinion because other robots cover the sight to the landmarks at the bifurcations. Moreover, small traffic jams and noise in the ground sensors can lead to longer or shorter task execution times. However, the method always converges and with high probability to the shortest path.

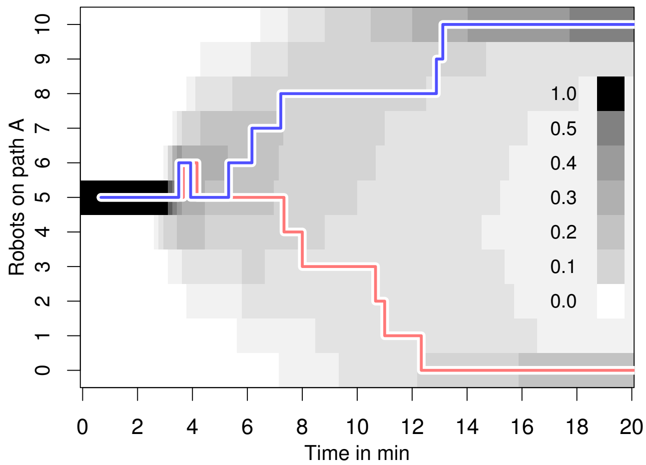


Fig. 14 Distribution of robots over time collected over 50000 simulations of Experiment I. The shade of gray indicates the probability to find a certain number of robots with opinion A at a given time in the system. The two lines correspond to data collected in two real robot experiments.

5.1.1 Distribution of robots over time

Fig. 14 depicts the evolution of the number of robots on path A over time. The shade of gray indicates the probability to find a certain number of robots with opinion A at a given time in the system. The corresponding data was obtained from the simulation model. The two depicted lines are examples of data collected in two real robot experiments. The evolution of the real robot experiments show that during a single run opinion switches in both directions can occur. Moreover, the top (darker) trajectory visualizes an effect that can occur in real robot experiments but not in simulation. In particular, in the depicted run from minute 7 to minute 13 the number of robots with opinion A remains constant although only two robots with opinion B are left. The reason is that these two remaining robots moved together in the arena as a group. When such a group enters the observation zone there is a high probability that the group members observe themselves and thus do not change opinion. However, eventually such groups dissolve and the system converges.

5.2 Extended analysis in the simulation model

5.2.1 Swarm size and initial bias

We study the influence of the swarm size and of the initial distribution of opinions. The analytical model predicts that the accuracy of the method increases with

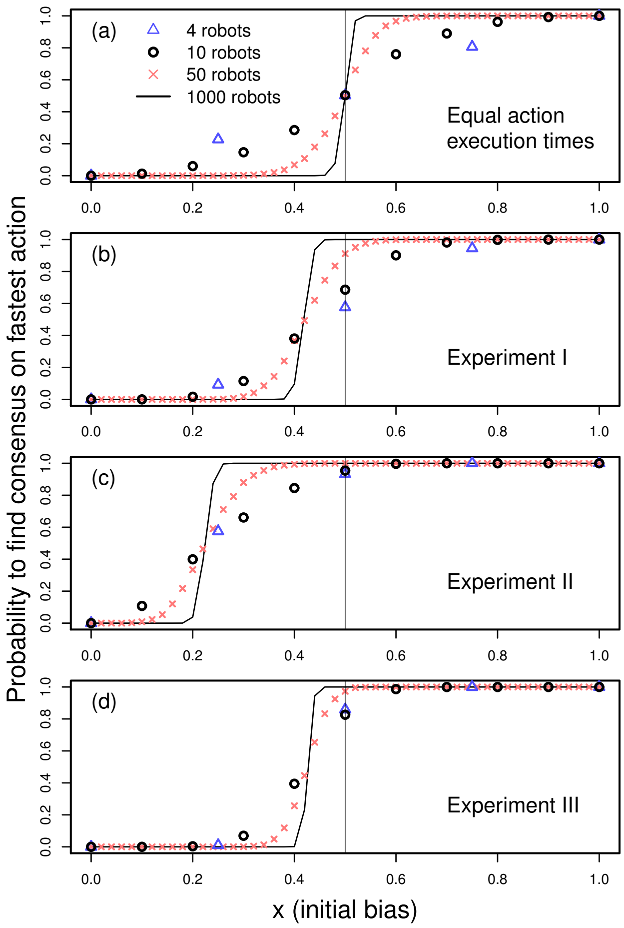


Fig. 15 Probability that the swarm finds the shortest path as a function of the initial bias for different swarm sizes

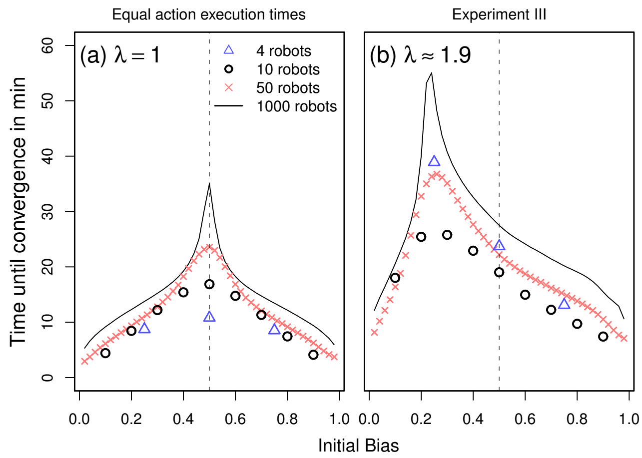


Fig. 16 Time to convergence as a function of the initial bias for different swarm sizes

the swarm size. To verify this important result, we simulate experiments I, II and III with different swarm sizes $(4, 10, 50$ and 1000 robots). Moreover, we investigate how the initial bias affects our decision making method. In accordance to Section 3, the initial bias gives the fraction of robots that start with opinion A. We denote the initial bias with x and we denote the initial bias for which the outcome of the experiments is random (the critical fraction) with x_c.

Concerning the probability to reach consensus on the fastest action, the simulations are in accordance with the predictions of the analytical model (Fig. 15a). As predicted by the analytical model, for equal task execution times the initial bias strongly influences the swarm's decision. In particular, swarms tend to reach consensus on the opinion that was initially favored by the majority of robots. This effect is stronger for larger swarms.

In accordance to the analytical model, the simulation shows that the critical bias x_c in Experiment I is shifted towards smaller values (Fig. 15b). More precisely, since in Experiment I the mean execution time ratio is $\lambda \approx 1.3$ the critical bias is now given by $x_c \approx 1/(1+1.3) = 0.43$. Consequently, swarms that start unbiased $(x = 0.5)$ have higher probability to find consensus on action A, the fastest action. The larger the swarm the more likely this outcome. Large swarms of 1000 robots found consensus on the fastest action in all conducted experiments.

Due to the longer execution time of action B, in Experiment II strong shift of the critical bias can be observed (Fig. 15c). Unbiased swarms find consensus on the fastest action with high probability. Even when a swarm starts slightly biased towards opinion B, the robots are able to find consensus on A. For example, if 6 out of 10 robots start with opinion B the swarm still finds consensus on opinion A with high probability (≈ 0.8).

The critical bias x_c in Experiment III (memory size $k = 4$) is the same as the one in Experiment I (Fig. 15d). However, as also predicted by the analytical model, the positive feedback is stronger for larger k. Consequently, the probability for unbiased swarms to find consensus on the fastest action A is higher than in Experiment I.

Fig. 16 depicts the time swarms need to find consensus depending on the initial bias x. In Fig. 16a the results for equal action execution times are shown. The values are symmetrical to $x = 0.5$ (no initial bias). It takes the longest time to find consensus near $x = 0.5$. Moreover, it can be seen that the time to convergence grows sublinearly with the number of robots. Thus, increasing the swarm size will have a strong influence on the decision accuracy but only a marginal influence on the time the system needs to converge.

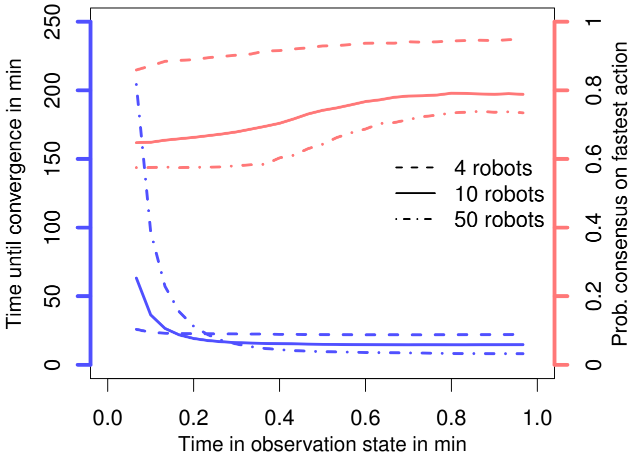


Fig. 17 Dependence of the performance of the system on the time the robots stay in the observation state.

In Fig. 16b the time swarms need to converge in Experiment III are depicted. The shift of the critical bias x_c influences the curves for the time to consensus. The curves are not symmetrical anymore since the longest convergence times are experienced near x_c.

5.2.2 Time in observation state

The observation zone in our double-bridge experiment has a fixed size. Thus, the time robots stay in the observation state is also fixed. However, in a different application of the decision method the time robots stay in the observation state might be adjustable. Therefore, it is interesting to investigate the influence of the duration of the observation state in our simulation model.

Fig. 17 presents the time the system needs to converge and the probability to converge to the fastest action depending on the time the robots stay in the observation state. If the robots stay only a short time in the observation state it is unlikely that it will observe another robot. Since robots observe their self if they do not observe another robot, the the probability to observe a different opinion that the one held k times becomes very small. Consequently, the time needed to find consensus can be very long. Moreover, the influence of random fluctuations increases and this lowers accuracy of the decision making method.

However, for larger swarms, as more robots will be in the observation state and therefore the probability of not observing other robots becomes smaller, the mentioned effects disappear.

5.2.3 Memory Size k

Experiment III shows that increasing the memory size from $k = 2$ to $k = 4$ leads to higher accuracy and longer convergence time for swarms of 10 robots. The analytical model also predicts that increasing the memory size k increases the accuracy of the decision making method on the cost of longer convergence times. To investigate the influence of the memory size further, we simulate swarms of size $N \in \{4, 10, 50\}$ that use memory sizes $k \in \{2, \ldots, 8\}$. A visualization of the results can be found in the supplementary material.

The simulation results show a trade-off between accuracy and time needed to converge. The shape of the curves are similar to the predictions of the analytical model. However, for small swarm sizes, in contrast to the analytical model, the simulation model predicts very large convergence times. Clearly, for small swarms the observation zone is often empty. Therefore, for large k the probability that a robot encounters other robots k times in a row becomes very small. For example, for swarms of 4 robots that use a memory of size $k = 8$ it takes more than 2 simulated days on average to converge to a decision.

6 Related work

Path finding and shortest path finding have been studied intensively in swarm robotics. Most of the proposed methods are based on the simulation of pheromones. Several approaches have been tested, for example, based on heat (Russell, 1997), oxid gas (Russell, 1999), alcohol (Fujisawa et al., 2008a,b), or phosphorescent glowing paint (Mayet et al., 2010). Other authors use digital video projectors to project the pheromone trails on the ground (Sugawara et al., 2004; Garnier et al., 2007; Hamann et al., 2007). Several studies rely on artifacts that are distributed in the environment. Such artifacts might be, for example, sensor nodes (Hara and Balch, 2007; Vigorito, 2007), RFID chips (Mamei and Zambonelli, 2005; Herianto and Kurabayashi, 2009) or other robots (Werger and Matarić, 1996; Payton et al., 2001; Nouyan et al., 2008; Hara and Balch, 2007; Nouyan et al., 2009; Campo et al., 2010; Ducatelle et al., 2010).

Beside shortest path finding, several other problems that need decentralized decision making have been studied in swarm robotics research. Wessnitzer and Melhuish (2003) investigate how a swarm of robots can decide which of two targets to hunt collectively. One of the proposed methods uses a majority voting between the individuals of the swarm to find consensus on a target. However, the target that is finally selected is random since no further measures like distance or target velocity are taken into account. Garnier et al. (2009) present a site-selection mechanism inspired by the aggregation behavior of cockroaches.

Parker and Zhang (2009) present a framework for collective decision making that shares similarities with our work. They propose a method to decide about the best out of a number of possible options. The authors take inspiration from the nest selection behavior in bees. However, in contrast to our work, the method demands that the robots are able to estimate the quality of single options. The same authors propose a method that allows a swarm to decide if a current task has been completed (Parker and Zhang, 2010). To this end, similar to our work, every robot is endowed with its own opinion about the completion status and memorizes a certain number of observed opinions of other robots. The memory is used to locally estimate the fraction of robots that have a certain opinion. If this estimation exceeds a predetermined threshold for at least one robot, this single robot initiates the commitment to the new opinion for the whole swarm. This behavior is in contrast to our work where the opinion of the whole swarm emerges out of the local opinions of the swarm members.

Montes de Oca et al. (2009) present a first study that exploits the bias induced by differing action execution times, to decide about the fastest action. The authors present simulation experiments of robots that communicate their (binary) opinions through the environment. Inspired by the binary Particle Swarm Optimization algorithm, single robots choose one of two possible actions by combining their personal opinion and the opinion they read from the environment. It was shown that with a high probability all robots eventually choose the fastest action.

Montes de Oca et al. (2011) also use positive feedback on the bias induced by differing action execution times to decide on the fastest action. The force that drives the agents to consensus on the fastest action is given by the so-called *Majority rule*. In the proposed method, in contrast to the method presented here, robots do not decide based on observed opinions stored in a memory. Instead, they form ad-hoc teams of three or more robots and apply the Majority rule on the opinions held by the members within the teams. The authors investigate the method in an agent based simulation. Moreover, they demonstrate the application of their method in a robot group transport application using physics based simulations.

The decision making method based on the k-Unanimity rule that we present in this paper has several advantages over the Majority rule based method of Montes de Oca et al. (2011). First, no teams have to be formed. The necessity to form ad-hoc teams restricts the Majority rule based method to those applications in which teams of robots are required. Second, in the k-Unanimity rule based method the accuracy of the decision can be adjusted. As shown, a higher accuracy can be achieved by using a larger memory of the cost of longer convergence times. Third, in the here presented method between two consecutive executions of actions, only one other opinion has to be observed. This is advantageous from the implementation point of view, as it is not necessary that the robots are able to distinguish each other. Fourth, in contrast to the Majority rule based method, the k-Unanimity rule based method also works well in relatively small swarms.

Scheidler (2011) presents a theoretical investigation of the Majority rule based method. Similar to the analytical model presented here, this work takes into account the random fluctuations that occur in finite swarms. Generally, stochasticity (e.g., due to sensor noise or the random encounters of the robots) is an inherent property of swarm robotics systems. It is a promising research direction to study how to include stochasticity in analytical models for swarm robotics systems. This can help to derive, based on stochastical differential equations, macroscopic models for the spatial distribution of swarms of robots (Hamann and Wörn, 2008; Berman et al., 2011).

7 Conclusion

In this paper we proposed a self-organized decision making method that allows swarm robotics systems to collectively find the fastest action out of a set of possible actions. In the presented method every robot is endowed with its own opinion about which is the fastest action. The robots apply a simple local rule (the k-Unanimity rule) to find consensus on one opinion. With high probability this opinion represents the fastest action.

We used an analytical model to show that the k-Unanimity rule amplifies an existing opinion bias. Moreover, we have shown that if the opinions are associated with different execution times, swarms tend to select the action that has the shortest mean execution time. The theoretical model predicts a trade-off between the accuracy of the method and the time it needs to converge.

We validated the decision making method in a set of real robot experiments. The goal of the robotic swarm was to find a short path between two locations. The robots used only local information and indirect communication. The experiments have shown that the proposed method allows a real swarm of robots to collectively select the shortest of two paths. The swarm can accomplish this without the need to measure traveling times. Moreover, the robots do not need sophisticated communication capabilities. Instead, the robots only need to be able to observe other robot's opinions. As such, only the indirect and anonymous communication of opinions is necessary. The experiments with real robots showed that the method works also in the presence of errors due to sensor noise or robot failures.

Our study showed that the accuracy of the decision making method is determined by different factors. The accuracy increases with the swarm size. Large swarms find the shorter action with higher accuracy while the time they need to converge does not increase drastically. The accuracy is also higher the more the two actions differ in execution time. Increasing the size of the observation memory k leads to higher accuracy of the method. The value of k can be adjusted and allows to regulate the accuracy of the decision making at the cost of longer convergence time.

Acknowledgements

This work was partially supported by the European Research Council via the ERC Advance Grant "E-SWARM: Engineering Swarm Intelligence Systems" (grant 246939). The information provided is the sole responsibility of the authors and does not reflect the European Commission's opinion. The European Commission is not responsible for any use that might be made of data appearing in this publication. Alexander Scheidler is supported by the Postdoc Programme of the German Academic Exchange Service (DAAD). Arne Brutschy and Marco Dorigo acknowledge support from the F.R.S.-FNRS of Belgium's French Community. Eliseo Ferrante and Alexander Scheidler acknowledge support from the Meta-X project, funded by the Scientific Research Directorate of the French Community of Belgium.

References

Berman, S., Kumar, V., and Nagpal, R. (2011). Design of control policies for spatially inhomogeneous robot swarms with application to commercial pollination. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2011)*, pages 378–385. IEEE Press, Piscataway, NJ.

Campo, A., Gutiérrez, A., Nouyan, S., Pinciroli, C., Longchamp, V., Garnier, S., and Dorigo, M. (2010). Artificial pheromone for path selection by a foraging swarm of robots. *Biological Cybernetics*, 103(5):339–352.

Diwold, K., Schaerf, T., Myerscough, M., Middendorf, M., and Beekman, M. (2011). Deciding on the wing: in-flight decision making and search space sampling in the red dwarf honeybee Apis florea. *Swarm Intelligence*, 5(2):121–141.

Dorigo, M., Floreano, D., Gambardella, L. M., Mondada, F., Nolfi, S., Baaboura, T., Birattari, M., Bonani, M., Brambilla, M., Brutschy, A., Burnier, D., Campo, A., Christensen, A. L., Decugnière, A., Caro, G. D., Ducatelle, F., Ferrante, E., Fröster, A., Gonzales, J. M., Guzzi, J., Longchamp, V., Magnenat, S., Mathews, N., de Oca, M. M., O'Grady, R., Pinciroli, C., Pini, G., Rétornaz, P., Roberts, J., Sperati, V., Stirling, T., Stranieri, A., Stützle, T., Trianni, V., Tuci, E., Turgut, A. E., and Vaussard, F. (2011). Swarmanoid: a novel concept for the study of heterogeneous robotic swarms. Technical Report TR/IRIDIA/2011-014, IRIDIA, Université Libre de Bruxelles, Brussels, Belgium.

Dorigo, M. and Sahin, E. (2004). Guest editorial. Special issue: Swarm robotics. *Autonomous Robots*, 17(2–3):111–113.

Ducatelle, F., Di Caro, G., and Gambardella, L. M. (2010). Cooperative self-organization in a heterogeneous swarm robotic system. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2010)*, pages 87–94. ACM Press, New York, NY.

Ducatelle, F., DiCaro, G., Pinciroli, C., and Gambardella, L. (2011). Self-organized cooperation between robotic swarms. *Swarm Intelligence*, 5(2):73–96.

Fujisawa, R., Dobata, S., Kubota, D., Imamura, H., and Matsuno, F. (2008a). Dependency by concentration of pheromone trail for multiple robots. In *LNCS 5217. Proceedings of the International Conference on Ant Colony Optimization and Swarm Intelligence (ANTS 2008)*, pages 283–290. Springer, Berlin, Germany.

Fujisawa, R., Imamura, H., Hashimoto, T., and Matsuno, F. (2008b). Communication using pheromone field for multiple robots. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2004)*, pages 1391–1396. IEEE Press, Piscataway, NJ.

Garnier, S., Gautrais, J., Asadpour, M., Jost, C., and Theraulaz, G. (2009). Self-organized aggregation triggers collective decision making in a group of cockroach-like robots. *Adaptive Behavior*, 17(2):109–133.

Garnier, S., Tache, F., Combe, M., Grimal, A., and Theraulaz, G. (2007). Alice in pheromone land: An experimental setup for the study of ant-like robots. In *Proceedings of the IEEE Swarm Intelligence Symposium (SIS 2007)*, pages 37–44. IEEE Press, Piscataway, NJ.

Goss, S., Aron, S., Deneubourg, J.-L., and Pasteels, J. M. (1989). Self-organized shortcuts in the argentine ant. *Naturwissenschaften*, 76(12):579–581.

Gutiérrez, A., Campo, A., Santos, F. C., Monasterio-Huelin, F., and Dorigo, M. (2009). Social odometry: Imitation based odometry in collective robotics. *International Journal of Advanced Robotic Systems*, 26(2):129–136.

Hamann, H., Szymanski, M., and Wörn, H. (2007). Orientation in a trail network by exploiting its geometry for swarm robotics. In *Proceedings of the IEEE Swarm Intelligence Symposium (SIS 2007)*, pages 310–315. IEEE Press, Piscataway, NJ.

Hamann, H. and Wörn, H. (2008). A framework of space-time continuous models for algorithm design in swarm robotics. *Swarm Intelligence*, 2(2-4):209–239.

Hara, K. and Balch, T. (2007). Pervasive sensor-less networks for cooperative multi-robot tasks. *Distributed Autonomous Robotic Systems 6*, pages 305–314.

Herianto, H. and Kurabayashi, D. (2009). Realization of an artificial pheromone system in random data carriers using RFID tags for autonomous navigation. In *IEEE International Conference on Robotics and Automation (ICRA 2009)*, pages 2288–2293. IEEE Press, Piscataway, NJ.

Mamei, M. and Zambonelli, F. (2005). Physical deployment of digital pheromones through RFID technology. In *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2005)*, pages 1353–1354. ACM Press, New York, NY.

Mayet, R., Roberz, J., Schmickl, T., and Crailsheim, K. (2010). Antbots: a feasible visual emulation of pheromone trails for swarm robots. In *Proceedings of the 7th international conference on Swarm intelligence*, LNCS 6234, pages 84–94, Berlin, Heidelberg. Springer-Verlag.

Montes de Oca, M. A., Ferrante, E., Mathews, N., Birattari, M., and Dorigo, M. (2009). Optimal collective decision-making through social influence and different action execution times. In Curran, D. and O'Riordan, C., editors, *Proceedings of the Workshop on Organisation, Cooperation and Emergence in Social Learning Agents of the European Conference on Artificial Life (ECAL 2009)*. Springer, Berlin, Germany.

Montes de Oca, M. A., Ferrante, E., Scheidler, A., Birattari, M., and Dorigo, M. (2011). Opinion formation models with differential latency: A self-organized collective decision-making mechanism. *Swarm Intelligence*, 5(3-4).

Nouyan, S., Campo, A., and Dorigo, M. (2008). Path formation in a robot swarm: Self-organized strategies to find your way home. *Swarm Intelligence*, 2(1):1–23.

Nouyan, S., Groß, R., Bonani, M., Mondada, F., and Dorigo, M. (2009). Teamwork in self-organized robot colonies. *IEEE Transactions on Evolutionary Computation*, 13(4):695–711.

Parker, C. A. C. and Zhang, H. (2009). Cooperative decision-making in decentralized multiple-robot systems: The best-of-N problem. *IEEE/ASME Transactions on Mechatronics*, 14(2):240–251.

Parker, C. A. C. and Zhang, H. (2010). Collective unary decision-making by decentralized multiple-robot systems applied to the task-sequencing problem. *Swarm Intelligence*, 4(3):199–220.

Payton, D., Daily, M., Estowski, R., Howard, M., and Lee, C. (2001). Pheromone robotics. *Autonomous Robots*, 11:319–324.

Russell, R. A. (1997). Heat trails as short-lived navigational markers for mobile robots. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 1997)*, pages 3534–3539. IEEE Press, Piscataway, NJ.

Russell, R. A. (1999). Ant trails – An example for robots to follow? In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 1999)*, pages 2968–2703. IEEE Press, Piscataway, NJ.

Şahin, E. (2005). Swarm robotics: From sources of inspiration to domains of application. In Şahin, E. and Spears, W. M., editors, *LNCS 3342. Swarm Robotics. SAB 2004 International Workshop*, pages 10–20. Springer, Berlin, Germany.

Scheidler, A. (2011). Dynamics of majority rule with differential latencies. *Physical Review E*, 83(3):031116.

Seeley, T. D. (2010). *Honeybee Democracy.* Princeton University Press, Princeton, NJ.

Sugawara, K., Kazama, T., and Watanabe, T. (2004). Foraging behavior of interacting robots with virtual pheromone. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2004)*, pages 3074–3079. IEEE Press, Piscataway, NJ.

Vigorito, C. M. (2007). Distributed path planning for mobile robots using a swarm of interacting reinforcement learners. In *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*, AAMAS '07, pages 120:1–120:8. ACM Press, New York, NY.

Werger, B. and Matarić, M. (1996). Robotic "food" chains: Externalization of state and program for minimal-agent foraging. In *Proceedings of the International Conference on Simulation of Adaptive Behavior: From Animals to Animats (SAB 1996)*, pages 625–634. MIT Press, Cambridge, MA.

Wessnitzer, J. and Melhuish, C. (2003). Collective decision-making and behaviour transitions in distributed ad hoc wireless networks of mobile robots: Target-hunting. In *Proceedings of the European Conference on Artificial Life (ECAL 2003) LNCS 2801*, pages 893–902. Springer, Berlin, Germany.

Winfield, A. (2010). Towards an engineering science of robot foraging. In H. Asama, H. Kurokawa, J. O. and Sekiyama, K., editors, *Distributed Autonomous Robotic Systems 8*, pages 185–192. Springer, London, UK.