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

Alexander Scheidler, Arne Brutschy, Eliseo Ferrante, and Marco Dorigo, *Fellow, IEEE*

***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. By means of positive feedback the method achieves consensus on the fastest action. The novelty of our method is that it allows robots to collectively find consensus on the fastest action without measuring explicitly the execution times of all available actions. We study two analytical models of the decision-making method in order 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 method.**

***Index Terms*—Intelligent robots, intelligent systems, multi-robot systems.**

I. Introduction

SWARM robotics deals with large groups of relatively simple robots that perform tasks that go beyond their individual capabilities [1]–[3]. 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 and of a central controller imposes challenging problems for the design of such a system. When engineering decision-making methods for swarm robotics systems, the distributed nature of these swarms must be taken into account: methods must be efficient, robust with respect to robot failures, and scale well with respect to the swarm size.

One important research question is how robot swarms can make collective decisions—a need that arises in many applications of swarm robotics systems. For example, robot swarms might need to select the best location containing objects to be retrieved [4], to decide whether a certain subtask is finished [5], or to select the shortest between a set of paths from a source to a destination [6]. Different from individual decision-making problems such as the multiarmed bandit problem [7], [8], collective decision-making in swarms typically requires some form of consensus among the individuals of the swarm. Existing methods typically rely on measuring the quality of the available options, followed by some form of explicit negotiation or consensus-finding (e.g., [5], [9], [10]).

In this paper, we introduce a collective decision-making method for swarms of robots that is based on positive feedback. 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 own opinion. Positive feedback eventually leads to consensus on one single opinion that is shared within the whole swarm. Due to a bias induced by the difference in the execution times, with high probability the consensus is reached on the opinion representing the fastest action. The proposed method works in a self-organized and decentralized way. Moreover, the method does not require robots to explicitly measure action execution times—a fact that differentiates our method from existing ones. 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. In Section II, we discuss related works and differentiate the method we propose from existing methods. In Section III, we present the decision-making method that we propose for swarms of robots. We study two analytical models of the proposed method to understand the dynamics of the consensus formation process in Section IV. We verify the

Manuscript received October 6, 2014; revised February 7, 2015; accepted April 19, 2015. Date of publication June 1, 2015; date of current version April 13, 2016. This work was supported in part by the European Research Council via the ERC Advance Grant "E-SWARM: Engineering Swarm Intelligence Systems" under Grant 246939. The work of A. Scheidler was supported by the Post-Doctoral Programme of the German Academic Exchange Service. The work of A. Brutschy and M. Dorigo was supported by the F.R.S.-FNRS of Belgium's French Community. The work of E. Ferrante and A. Scheidler was supported by the Meta-X Project, funded by the Scientific Research Directorate of the French Community of Belgium. The work of E. Ferrante was supported in part by the ESF H2SWARM Program, from the Fund for Scientific Research—Flanders, and in part by the Katholieke Universiteit Leuven Projects IDO-BioCo3 and Excellence Center Financing under Grant PF/2010/007. This paper was recommended by Associate Editor T.-H. S. Li.

A. Scheidler is with the Fraunhofer Institute for Wind Energy and Energy System Technology, Kassel 34119, Germany (e-mail: alexander.scheidler@iwes.fraunhofer.de).

A. Brutschy and M. Dorigo are with the Institut de Recherches Interdisciplinaires et de Développements en Intelligence Artificielle (IRIDIA), Université Libre de Bruxelles, Brussels 1050, Belgium (e-mail: arne.brutschy@ulb.ac.be; mdorigo@ulb.ac.be).

E. Ferrante is with the Laboratory of Socioecology and Social Evolution at University of Leuven, Leuven 3000, Belgium (e-mail: eliseo.ferrante@bio.kuleuven.be).

This paper has supplementary downloadable multimedia material available at http://ieeexplore.ieee.org provided by the authors. This includes a series of .mov files. This material is 71.6 MB in size.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2015.2429118

2168-2267 © 2015 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

applicability of the method in a real-world application scenario where a robotic swarm has to collectively select the shorter of two paths. 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 V and the obtained results are given in Section VI. We conclude this paper in Section VII.

II. Related Work

Collective decision-making has been studied intensively in the field of swarm robotics. In particular, several strategies have been investigated for collective path finding and shortest path finding. Most strategies use artificial pheromones, which have been implemented in various ways, for example by means of heat [11], oxid gas [12], alcohol [13], [14], or phosphorescent glowing paint [15]. Other authors use digital video projectors to project the pheromone trails on the ground [16]–[18]. Several studies rely on artifacts that are distributed in the environment. Such artifacts might be, for example, sensor nodes [19], [20], RFID chips [21]–[23], or other robots [6], [19], [24]–[26].

Beside shortest path finding, several other related problems that require decentralized decision-making have been studied in swarm robotics research. Wessnitzer and Melhuish [9] 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.* [17] present a site-selection mechanism inspired by the aggregation behavior of cockroaches [27], [28].

Parker and Zhang [5] presented a framework for collective decision-making that shares similarities with [10]. They proposed a method to decide about the best out of a number of possible options. The authors take inspiration from the nest selection behavior of bees. However, in contrast to this paper, the method requires that the robots are able to estimate the quality of each individual option. The same authors propose a method that allows a swarm to decide whether a current task has been completed [5]. To this end, similar to his 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 the swarm that has 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 different from this paper, where the opinion of the whole swarm emerges out of the local opinions of the swarm members.

Montes de Oca *et al.* [29] 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. It is in our plan to compare the k-unanimity rule with the majority and other rules in a formal way, in a similar fashion as in [30].

The decision-making method based on the k-unanimity rule that we present in this paper is an improvement over the majority rule-based method of [29]. As such, it has several advantages over the original method. 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 at the cost of longer convergence times. Third, in this paper only one other opinion has to be observed between two consecutive executions of actions. 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.

A theoretical investigation of the majority rule-based method is presented in [31]. Similar to the analytical model developed here, this paper takes into account the random fluctuations that occur in finite swarms. Generally, stochasticity (e.g., due to sensor noise) 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 stochastic differential equations, macroscopic models for the spatial distribution of swarms of robots [32], [33].

In [34], we conduct a preliminary analysis of a simple model of opinion dynamics which is similar to the k-unanimity rule: instead of keeping a memory of k observations, the agents weigh exponentially their past observations through a unique memory variable. In comparison to the this paper, we only conducted a Monte Carlo simulation that merely showed that the method can be used to perform selection of the shortest action in presence of differential latency.

Decision-making has also been formally studied within the machine learning community. The multiarmed bandit problem [7], [8], [35] is a related problem faced by an individual who needs to select one option among many alternative options. Each option is associated with a stochastic reward; the individual's objective is to maximize the sum of rewards earned through a sequence of decisions. Although related to the problem we tackle here, there are important differences between the multiarmed bandit problem and the collective decision-making problem considered in this paper: 1) our problem is of a collective nature and 2) we assume that each individual does not have access to a reward, that is, individuals cannot measure execution times and thus cannot perform direct comparison between the performance of the alternative options.

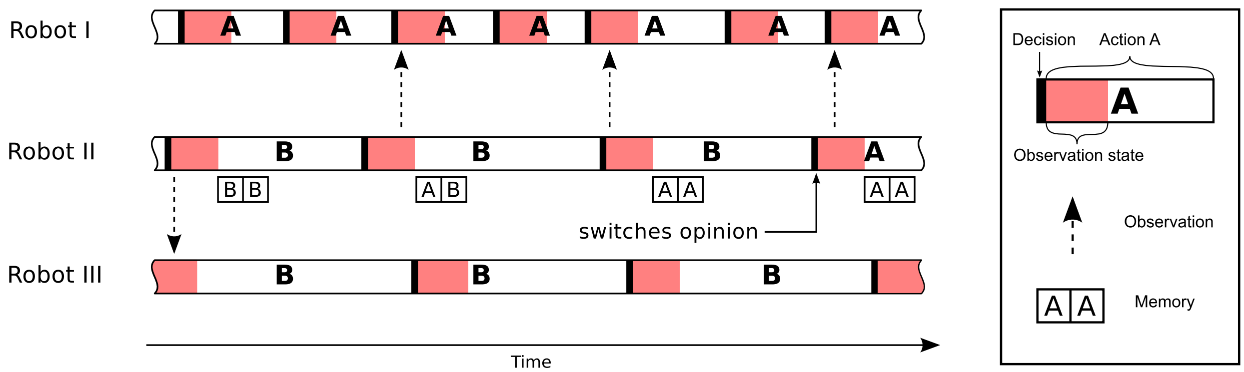


Fig. 1. Example observation and opinion switching from the point of view of Robot II. Robot I has opinion **A** and Robot II and Robot III have opinion **B**. Action **B** takes longer to be executed than action **A** on average. Due to the fixed time length of the observation state, Robot I is in observation state more often than Robot III. Robot II switches to opinion **A** after observing **A** $k = 2$ times in a row.

III. Decision-Making Method Based on the *k*-Unanimity Rule

In this paper, we consider swarms of robots that are capable of executing different actions, designated by bold roman letters such as **A**, **B**, **C**, **D**, ... Each action takes a stochastic amount of time to be executed. Initially, the robots execute the different actions in equal shares. However, the overall goal of the swarm is to break this symmetry toward the fastest action. This means that, at the end of the decision-making process, all robots should execute the same action and that this action should be the action that has the fastest average execution time. Without loss of generality, in the reminder of this paper action **A** is the fastest action.

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**, opinion **B** corresponds to action **B**, and so on.

Robots can observe the opinion of other robots when they meet. Every robot stores its most recent k observed opinions in its memory. Once the memory is full, if a robot observes a new opinion, it removes the oldest opinion from it's memory and replaces it with the new one, in a first-in first-out fashion. Note that, we use the term memory as an abstraction; it does not imply any details regarding its technical implementation.

In between two action executions, robots can change their opinion according to the so-called k-unanimity rule, defined as follows:

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

The k-unanimity rule induces positive feedback on the opinion that is held by the majority of the robots. Applied repeatedly over time, the k-unanimity rule drives the swarm to consensus, that is, to a state in which all robots hold the same opinion. The k-unanimity rule is a generalization of the voter model, which can be obtained by setting $k = 1$. The voter model is, in itself, a very general model of collective decision-making [36], [37]. Both in our model and in the voter model, consensus must be reached eventually, because the consensus states are the only absorbing states of the system (see Section IV). That is, once consensus is reached, robots can no longer change their opinion and we say that the swarm completed the decision-making process.

Throughout this paper, we assume that opinions are initially equally distributed among robots, that is, all opinions are held by an equal number of robots. Under this assumption, if all opinions could be observed with with the same probability (i.e., because there is no symmetry-breaking difference in mean execution times), there is no majority opinion that would be amplified by the k-unanimity rule. Instead, changes in the opinion of the robots would be random. Eventually, due to these random opinion switches, the symmetry between opinions will be broken in favor of a random opinion and the k-unanimity rule will likely amplify this opinion until consensus is reached.

However, our goal is not to achieve 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 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 allowed to observe and to be observed only 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 the same for all robots regardless of their opinion. Due to the fixed length of the observation state, robots that execute the fastest action are in observation state more often: Robots that execute the fastest action spend a larger fraction of their time in the observation state compared to the other robots (see Fig. 1). Therefore, the probability of the other robots observing the opinion held by the robots that execute the fastest action is higher than the probability to observe another opinion. In (initially) unbiased swarms this breaks the symmetry between the opinions in favor of the fastest action. Consequently, with higher probability, the system will evolve consensus on this action.

Note that the proposed decision-making method does not guarantee that the fastest action is always found. Swarms might still reach consensus on a slower action. This is due to the inherent randomness of encounters and the resulting opinion switches. In particular, at the start of the decision-making process, it can happen that, by chance, the number of robots with a certain opinion becomes larger than the number of robots with opinion associated with the fastest action. This can result in a higher probability to observe a different opinion than opinion **A**. Consequently, chances are that in this case the swarm finds consensus on an opinion different from opinion **A**.

IV. 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 are useful to understand the dynamics of the decision-making method and to predict its behavior 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. Every robot holds one of $O = \{\mathbf{A}, \mathbf{B}, \mathbf{C}, \dots\}$ possible opinions. We denote the number of robots with opinion $m \in O$ by n_m. Let $x_m = n_m/N$ be the fraction of these robots. The average execution time of action m is λ_m. Without loss of generality we assume that the average execution time of action **A** is $\lambda_A = 1$. The average execution time of any other action m is equal or longer, that is, $\lambda_m \geq 1$ for $m \neq \mathbf{A}$.

Robots repeatedly execute the action that corresponds to their current opinion. Therefore, within a time unit, on average n_m/λ_m robots with opinion m finish their actions. Therefore, the probability that a robot that finishes an action has opinion m is given by

$$p_m = \frac{n_m/\lambda_m}{\sum_{i \in O} \frac{n_i}{\lambda_i}} = \frac{x_m/\lambda_m}{\sum_{i \in O} \frac{x_i}{\lambda_i}}. \tag{1}$$

Three assumptions are made here. First, in our analytical model we do not model the observation state explicitly. Instead, robots apply the k-unanimity rule directly after the execution of an action and immediately start a new action. Second, we assume that the observation memory is filled with k opinions sampled accordingly to the current rates at which robots finish their actions. This means that each of the k memorized opinions is opinion m with probability p_m. Clearly, this corresponds to a well-mixed state where the probability to observe any robot in observation state is equal and observing the opinion of a certain robot does not depend on its opinion. Third, we neglect the fact that the stored opinions in a robot's memory might have been observed at different times (between several action executions). This assumption is a reasonable simplification, as in the real system opinions stored in the observation memory must have been observed in the near past and represent therefore a snapshot of the current rates at which the robots finish their actions.

At the level of the whole swarm, a robot's application of the k-unanimity rule can have three different outcomes: the number of robots with opinion m increases by one, decreases by one, or stays as it is. The number of robots with opinion m increases if a robot with opinion $r \neq m$ changes its opinion to m. This happens if the robot observes opinion m k times in a row. The probability for this event is ${p_m}^k$. It follows that the probability that any robot with opinion r changes to m is given by $p_r({p_m}^k)$. Hence

$$w_m^+ = \sum_{r \in O,\ r \neq m} p_r {p_m}^k = (1 - p_m){p_m}^k \tag{2}$$

is the probability that an application of the k-Unanimity rule increases the number of robots with opinion m. Similarly, the probability that the number of robots with opinion m is decreased is

$$w_m^- = \sum_{r \in O,\ r \neq m} p_m {p_r}^k. \tag{3}$$

The probability that the number of robots with opinion m does not change upon an application of the k-unanimity rule is $w_m^* = 1 - w_m^+ - w_m^-$.

In the next two sections, we study the dynamics of consensus formation by means of two different theoretical models. First, we propose a continuum model that assumes an infinite number of robots. This model only allows to investigate how the average fractions of robots that prefer the different opinions evolve over time, but can be easily extended to systems with more than two opinions. Second, we propose a model that assumes a fixed swarm size and that takes the effects caused by fluctuations due to the random decisions of robots in finite swarms into account. This model is superior to the first one as it also allows to predict the probability for consensus on the shortest action and the time needed. However, it is much harder to solve especially in systems with more than two opinions.

A. Continuum Model

Recall that, within a time unit $\sum_{i \in O} n_i/\lambda_i$ robots finish their actions and apply the k-unanimity rule. This corresponds to a fraction $\sum_{i \in O} x_i/\lambda_i$ of the swarm. We can thus model the evolution of the expected fraction of robots with opinion m as

$$\dot{x}_m = \left(w_m^+ - w_m^-\right) \sum_{i \in O} \frac{x_i}{\lambda_i}. \tag{4}$$

First consider swarms of robots that have to decide between only two actions $O = \{\mathbf{A}, \mathbf{B}\}$. Fig. 2 visualizes (4) for this case. The graph shows $\dot{x}_A$, the expected change of the fraction x_A of robots with opinion **A** within a time unit, depending on the current value of x_A. The zeros of $\dot{x}_A$, that is, the stationary solutions of (4), are the (stable) consensus states $[x_A = 0]$ and $[x_A = 1]$ and the (unstable) equilibrium point $[x_A = \lambda_A/(\lambda_A + \lambda_B)]$. The latter of these three points is of particular interest as it separates the flow to the two consensus states. We denote this point as critical fraction x_c. The model predicts that if x_A is greater than x_c, then x_A will steadily increase until consensus on **A** is reached ($\dot{x}_A > 0$ for $x_A > x_c$). Analogously, for $x_A < x_c$ consensus on **B** will be found. Hence, the k-unanimity rule induces positive feedback and amplifies an existing bias. Note that the critical fraction x_c marks the state in which the probability to observe both opinions **A** and **B** is $p_A = p_B = 1/2$. In other words, at the critical fraction both actions are executed at the same rates and therefore there is no bias toward one opinion.

Beside the critical fraction x_c, the point $x_A = x_B = 1/2$ is of particular interest. This point corresponds to the initial state

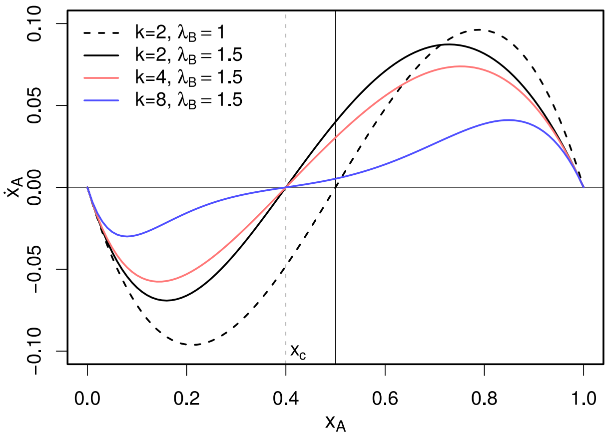


Fig. 2. Rate of change $\dot{x}_A$ of the fraction of robots with opinion **A** depending on the current system state in a swarm with two opinions. The vertical solid line marks unbiased swarms ($x_A = x_B = 0.5$). The dashed line marks the critical fraction $x_c = 0.4$ for $\lambda_B = 1.5$.

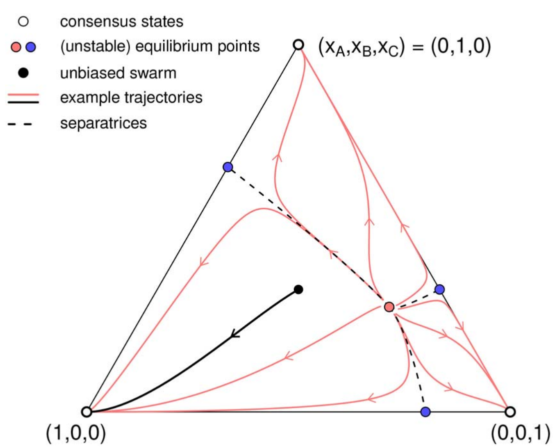


Fig. 3. Example trajectories of the continuum model (4) with three opinions.

of the swarm where both opinions are present in equal proportions. We call a swarm in this state unbiased. The vertical solid line in Fig. 2 marks this point.

If action **B** takes longer than action **A** ($\lambda_B > 1$) the critical fraction x_c is shifted toward values smaller that 0.5. For example, for $\lambda_B = 1.5$ the critical fraction is $x_c = 0.4$ (vertical dashed line in Fig. 2). However, it still holds that for any $x > x_c$ consensus is found on **A**. Consequently, the model predicts that unbiased swarms (i.e., $x_A = x_B = 1/2$) always find consensus on **A**, the fastest action.

When the memory size k increases, the rate of change $\dot{x}$ decreases and swarms need more time to find consensus (see Fig. 2). This is because for higher k it is harder to observe the same opinion k times in a row. This is particularly the case near the critical fraction. Note that the model presented in the next section shows that increasing the memory size k not only slows down the decision process, but also increases the probability to find the fastest action. This property cannot be shown with the continuum model.

Fig. 3 visualizes the state space of a system with three opinions $O = \{\mathbf{A}, \mathbf{B}, \mathbf{C}\}$, memory size $k = 2$ and execution times $\lambda_A = 1$, $\lambda_B = 2$, and $\lambda_C = 4$. The state of the swarm is now defined by the triple (x_A, x_B, x_C) denoting the fractions of robots with the three opinions. The black trajectory shown in Fig. 3 illustrates the evolution of an unbiased swarm, that is, a swarm that starts with equal fractions of robots for the three opinions $(1/3, 1/3, 1/3)$. As it can be seen, the model predicts that an unbiased swarm will converge to the state $(1, 0, 0)$ (consensus on the fastest opinion A). In analogy to the critical fraction that determines the fate of a swarm with two opinions, in systems with three opinions so-called separatrices divide the phase space into three areas that determine the outcome of the decision process.

B. Master Equation Approach

The continuum model we developed in the previous section shows how the fractions of robots with different opinions evolve on average. For a tuple of initial fractions $(x_A, x_B, x_C, \ldots)$, the continuum model predicts exactly one of the consensus states as final outcome. In real swarms, however, as long as an opinion is held by at least one robot, consensus on this opinion is still reachable.

In the following, we propose a second analytical model that also takes into account the effects caused by fluctuations due to the random decisions of robots in finite swarms. With this model, we can approximate the probability to reach consensus on a certain opinion. We chose to model only swarms with two opinions $O = \{\mathbf{A}, \mathbf{B}\}$ as, although in principle possible, it is an extremely difficult endeavor to solve master equations when more than two options are present.

Let E_n be the probability to eventually reach consensus on **A**, if currently $0 \leq n \leq N$ robots have opinion **A**. The $N + 1$ probabilities E_n can be estimated as follows. If n robots have opinion **A** ($n < N$), 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_A^+ and after this event, the probability for consensus on **A** is E_{n+1}. Therefore, $w_A^+ E_{n+1}$ is the probability that the next application of the k-unanimity rule increases the number of robots with opinion **A** from n to $n+1$ and that consensus on **A** will be found eventually. Considering also the two remaining outcomes of the application of the k-unanimity rule we obtain a so-called master equation

$$E_n = w_A^+ E_{n+1} + w_A^- E_{n-1} + w_A^* E_n. \tag{5}$$

Solving this master equation would mean to derive a nonrecursive, closed form for E_n. However, it is much easier to approximate its solution using a continuous function $E(x)$. This function is defined for $x \in [0, 1]$ and at the points $x = n/N$ ($n \in [0, \ldots, N]$) its value is $E(x) = E_n$. Rewriting the master equation in terms of $E(x)$ and applying a second order Taylor expansion leads to

$$\begin{aligned} E(x) &= w_A^+ E\left(\frac{n+1}{N}\right) + w_A^- E\left(\frac{n-1}{N}\right) + w_A^* E\left(\frac{n}{N}\right) \\ &= w_A^+ E(x + 1/N) + w_A^- E(x - 1/N) + w_A^* E(x) \\ &= w_A^+ \left[E(x) + \frac{1}{N}\frac{\partial E(x)}{\partial x} + \frac{1}{2}\frac{1}{N^2}\frac{\partial^2 E(x)}{\partial x^2}\right] \\ &\quad + w_A^- \left[E(x) - \frac{1}{N}\frac{\partial E(x)}{\partial x} + \frac{1}{2}\frac{1}{N^2}\frac{\partial^2 E(x)}{\partial x^2}\right] + w_A^* E(x). \end{aligned} \tag{6}$$

Because the sum $w_A^+ + w_A^- + w^*$ equals 1, the term $E(x)$ can be eliminated in (6) and we derive the second order differential equation

$$0 = \left[w_A^+ - w_A^-\right]\frac{1}{N}\frac{\partial E(x)}{\partial x} + \left[w_A^+ + w_A^-\right]\frac{1}{2}\frac{1}{N^2}\frac{\partial^2 E(x)}{\partial x^2}$$
$$= 2N\left[\frac{p_A^{k-1} - (1-p_A)^{k-1}}{p_A^{k-1} + (1-p_A)^{k-1}}\right]\frac{\partial E(x)}{\partial x} + \frac{\partial^2 E(x)}{\partial x^2}. \quad (7)$$

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 find consensus on **A** is one ($E_N = 1$). These are 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. Within a time unit $n + (N - n)/\lambda_B$ robots finish their actions. We can determine the expected time between two robots finishing their action (between two applications of the k-unanimity rule) as

$$\delta t(x) = \frac{1}{n + (N-n)/\lambda_B} = \frac{p_A}{xN}. \quad (8)$$

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

$$T_n = \delta t(x) + w_A^+ T_{n+1} + w_A^- T_{n-1} + w^* T_n. \quad (9)$$

Inserting (8) into (9) and applying the same steps as used to derive (7) leads to

$$0 = \frac{2Np_A}{x\left[p_A(1-p_A)^k + (1-p_A)p_A^k\right]}$$
$$+ 2N\left[\frac{p_A^{k-1} - (1-p_A)^{k-1}}{p_A^{k-1} + (1-p_A)^{k-1}}\right]\frac{\partial T(x)}{\partial x} + \frac{\partial^2 T(x)}{\partial x^2}. \quad (10)$$

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

Fig. 4 shows solutions for $E(x)$ and $T(x)$ for $\lambda_B = 1.0$ and $\lambda_B = 1.5$ and different memory sizes k and swarm sizes N. Again the critical fraction x_c and the fraction $x = 1/2$ that relates to unbiased swarms are of particular interest.

Recall that for equal action execution times ($\lambda_B = 1$) the critical fraction is $x_c = 0.5$. The continuum model predicts that for $x > x_c$ consensus is found on **A** (see the previous section). However, this only holds on average. As long as consensus is not reached there is always a nonzero probability for both consensus states. As can be seen in Fig. 4(a), 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 ten robots six robots prefer **A** (that is, $x = 0.6$), the probability that consensus is reached on **B** is still $1 - E(0.6) \approx 0.28$. However, for larger swarms the probability to reach consensus on the minority opinion becomes smaller ($1 - E(0.6)$ drops to 0.09 for $N = 50$). For $N \to \infty$ the function $E(x)$ converges to a step function and small deviations toward one opinion are amplified and result in consensus on this opinion with high probability.

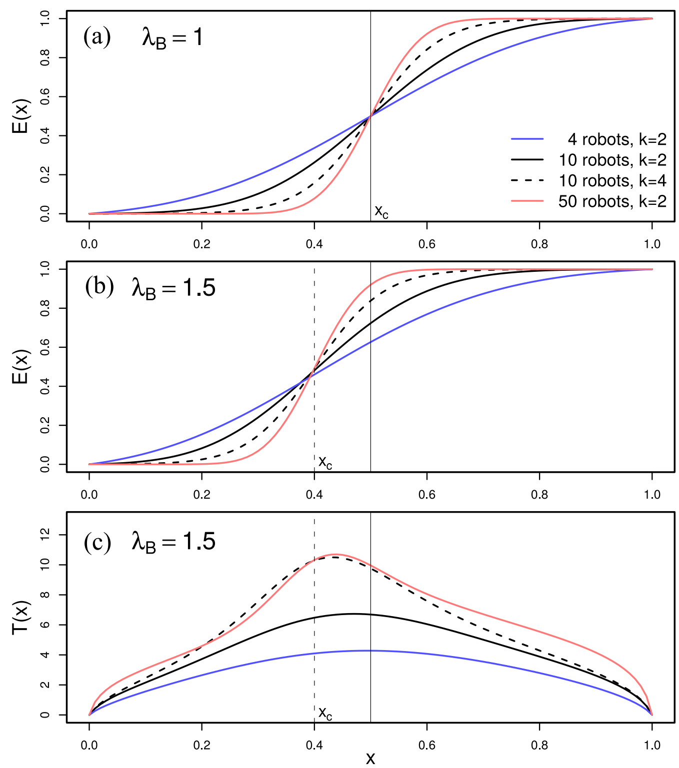


Fig. 4. Influence of the initial fraction x of robots that start with opinion **A**. (a) Approximation of $E(x)$, the probability to find consensus on opinion **A**, for equal action execution times ($\lambda_B = 1$). (b) $E(x)$ if action **B** takes longer than action **A** ($\lambda_B = 1.5$). (c) Expected time to consensus for $\lambda_B = 1.5$.

Recall that the continuum model correctly predicts that the rate of change $\dot{x}$ decreases with increasing k. That is, the probability to change opinion after an action execution decreases. The master equation model shows that if a robot changes its opinion, then for larger values of k the probability to switch to the majority opinion is higher. That is, increasing k results in stronger positive feedback. For example, if k is increased from 2 to 4, the probability for consensus on **A** for ten robots and $x = 0.6$ increases from 0.72 to 0.83.

In the case of asymmetric execution times, that is, if action **B** takes longer than action **A** [$\lambda_B = 1.5$, Fig. 4(b)], the critical fraction is smaller than 0.5 ($E(x_c) = 0.5$ for $x_c = 0.4$). Consequently, as already predicted by the continuum model, 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 as well as a larger memory size k lead to a higher probability of consensus on **A**.

As stated before, we assume that opinions are equally distributed among robots at the beginning of an experiment. However, in case opinions are randomly distributed among robots, the probability to find consensus on the fastest action might decrease, that is, robots might favor the longer action. Our model can be used to estimate this effect: for swarms with two opinions, the starting distribution of randomly generated opinions would be binomial. By convoluting this distribution with the estimation for the exit probability $E(x)$ one can estimate the exit probability for a swarm starting with

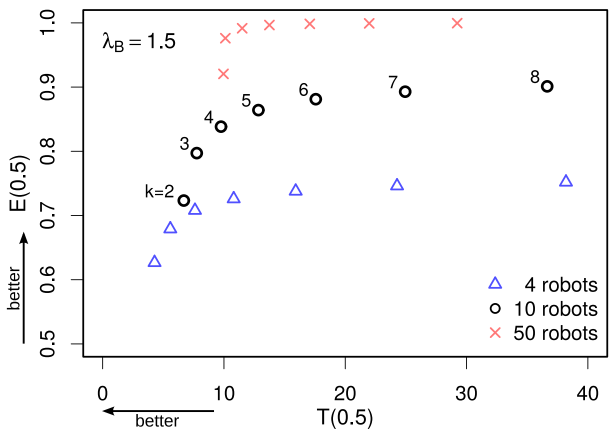


Fig. 5. Influence of memory size k. Shown is the time to convergence versus the probability to converge to action **A** for unbiased swarms.

randomly chosen opinions. For example, as Fig. 4(b) shows, the probability for an unbiased swarm of ten robots to find consensus on **A** is 0.723. If those ten robots start with a random opinion, this value would decrease to 0.695.

Fig. 4(c) depicts the expected time $T(x)$ to find consensus for the same situation as shown in Fig. 4(b). Near the critical fraction the probability to observe the two actions is nearly the same ($w_A^+ \approx w_A^-$). Therefore, the drift toward the consensus states is small and the expected time to convergence is high. However, the maximum time to consensus is not found exactly at the critical fraction, but slightly shifted toward higher values of x. A similar shift was observed in [31], but it remains unexplained.

For the normal case of unbiased starting swarms ($x = 0.5$), Fig. 5 depicts the time $T(0.5)$ to convergence versus the probability $E(0.5)$ to converge to action **A** for different swarm sizes $N \in \{4, 10, 50\}$ and different memory sizes $k \in \{2, \ldots, 8\}$. As it can be seen, the model predicts a trade-off between the probability to converge on the action with the fastest execution time and the time the swarm needs to reach the decision: Increasing the memory size k increases the probability to find consensus on **A** at the cost of a longer decision time. Clearly, since $w_A^+/(w_A^+ + w_A^-) \rightarrow 1$ for $k \rightarrow \infty$, the probability $E(0.5)$ converges to 1 for any $\lambda_B > 1$. However, as it can be seen in Fig. 5, increasing the accuracy by increasing k quickly becomes very costly in terms of convergence time because the probability that a robot changes its opinion ($w_A^+ + w_A^-$) also decreases exponentially with increasing k. The results shown in Fig. 5 demonstrate that choosing the right memory size k for a given application is not trivial and depends on the required precision and the available time.

The probability for consensus on action **A** depends not only on the number of robots and on the size of the memory, but also on the difference in action execution times. Obviously, if the action execution times are equal ($\lambda_A = \lambda_B$), the probability to find consensus on **A** is 0.5. On the other hand, the larger the difference between the action execution times, the higher is the probability that the swarm converges to opinion **A**. A visualization of these results can be found in the supplementary material [38].

V. Experimental Setup

The setup of our real robot experiment resembles the well-known double bridge experiment used to show that ants are able to find the shortest path between their nest and a food source [39]. This setup was chosen mainly for demonstration purposes and does not restrict by any means the applicability of our method. Following the taxonomy of [40], our experiment can be classified as a single nest, single source, homogeneous foraging task. The robots' task is to repeatedly collect objects from the source zone and transport them to the nest zone (see Fig. 6). The robots transport virtual objects and the source zone contains an unlimited number of objects. Hence, the robots' task is to constantly travel between the nest and the source zones. The overall goal of the robot swarm is to collect as many objects as possible. The best performance can be reached when the swarm uses solely the shortest of the two paths. Note that using exclusively the shortest path is advantageous only if no strong physical interference between the robots occurs (e.g., in situations where robots do not have to constantly avoid each other because of a high robot density in the arena). Large swarms might gain better performance by using both paths simultaneously, as this reduces the interference on the single paths. However, as we used a small swarm in our experiments, the effect of interference can be neglected.

The experimental arena has a size of 4.5 m x 3.5 m (see Fig. 6). 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 short path is called "**A**" and the long path is called "**B**." The observation zone is located next to the nest zone. Lights near the nest zone help the robots navigate within the arena. Moreover, two landmarks are placed at the two bifurcations of the double bridge. The landmarks help robots to navigate at the bifurcations; they are implemented using blue LEDs.

For this paper, we use ten marXbots [41]. The marXbot is a modular robot that was developed within the FET project Swarmanoid [42]. It has a circular chassis with a diameter of 17 cm, a height of 29 cm, and a weight of 1.8 kg. We use the marXbot's 24 IR proximity sensors to implement the obstacle avoidance behavior, their 24 light sensors to implement the light following behavior, their four IR ground sensors to distinguish between the different zones in the experimental arena, their RGB beacons to show the robots' current opinion and their onmi-directional camera to enable the robots to observe opinions of other robots.

A. Robot Behavior

The robots have no global map of the environment. They navigate only using a light source located next to the nest zone as a reference. 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

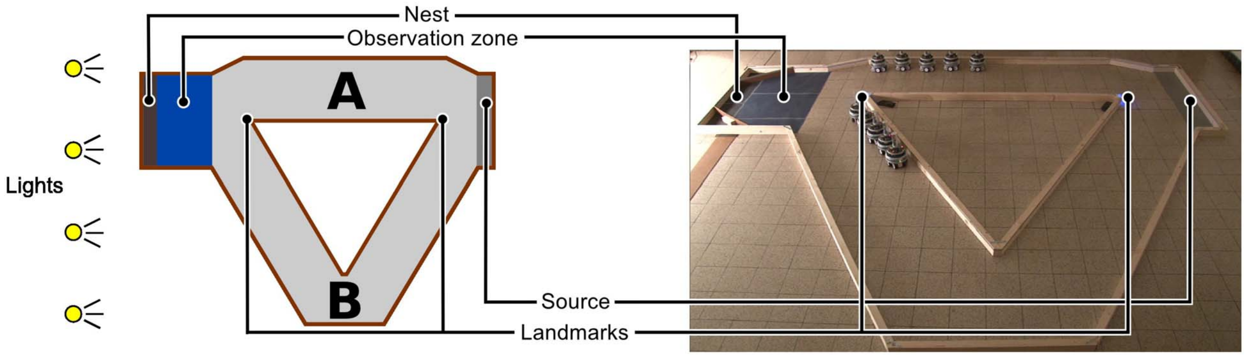


Fig. 6. Experimental setup: schematic (left) and real installation with a size of 4.5 m x 3.5 m (right). The robots constantly travel between nest zone and source zone by navigating with respect to the lights (anti-phototaxis for going to the source zone and phototaxis for going to the nest zone). Depending 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.

Fig. 7. Illustration of the observation and decision process shown on the example of a single robot. (a) Robot with opinion **A** (encircled) enters the observation zone. (b) Robot observes another robot with opinion **B** (the robot shows this to the experimenter by flashing its LED ring) and stores the observation in its memory. (c) Robot leaves the observation zone and the application of the k-unanimity rule changes its opinion to **B**.

opinions **A** and **B** represent the two actions "travel path A" and "travel path B," respectively. At the bifurcations between the two paths, designated by the aforementioned landmarks, robots navigate toward the path that corresponds to their opinion. More specifically, robots that have opinion **A** and are moving toward the source zone try to pass the landmark at the left hand side whereas robots going back to the nest zone try to pass the landmark at the right hand side. For robots with opinion **B**, this behavior 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. 7 illustrates different stages of the observation and decision process from the point of view of a single robot. The encircled robot in Fig. 7(a) has opinion **A** (green beacon—right robot). It is moving toward the nest zone and is entering the observation zone. 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 going to leave the observation zone moving toward 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 robot in our example observes opinion **B** from the left robot [indicated by an arrow in Fig. 7(b)]. In principle, more elaborate mechanisms can be implemented on the specific robotic platform we use. For instance, robots might observe each other's opinions by locally communicating them using an on-board range-and-bearing sensing and actuation system, which would also allow to communicate robot IDs. However, we choose not to use this solution as this would constrain the method to a specific robotic platform, which is not our intention.

The decision process is based on the k-unanimity rule introduced in Section III: if a robot with opinion **A** (resp. **B**) observes opinion **B** (resp. **A**) k times in a row, it changes its own opinion. This change is delayed until the robot leaves the observation zone. Thus, as long as a robots is located in the observation zone, it keeps and propagates the opinion that is associated with its last executed action. The robot in the example observed opinion **B** k times in a row and changes to **B** when it leaves the observation zone [Fig. 7(c)].

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). However, if the swarm is small or the average time in the observation state is very short, the probability that a robot observes another one can become very small. In this case, the self-observation rule can lead to long convergence times, because robots mostly observe themselves. If the convergence time is important, it might be more practical to drop the self-observation rule in favor of faster convergence times.

TABLE I
PARAMETERS FOR THE REAL ROBOT EXPERIMENTS

Experiment	**I**	**II**	**III**
Runs	15	15	15
Memory Size k	2	2	4
Average execution time λ_B	1.3	2.3	1.3

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 able to recognize if a certain robot has already been observed, or to distinguish between two or more robots present in an image. Neither of these options can be realized without the use of more sophisticated hardware and/or software implementations, and we chose not to do so because our main point is not to solve a real engineering problem, but to show that the proposed decision-making mechanisms works in a very simple and hardware and software-limited proof of concept scenario.

B. Experiments With Real Robots

The swarm of ten robots is initially divided into two groups of five robots each. Group one starts with opinion **A** and group two starts with opinion **B**. The robots start moving to the source zone in pairs of two robots, one from each group. The time interval between the consecutive starts of two pairs is approximately 15 s. This ensures a homogeneous distribution of robots in the arena and avoids the formation of clusters of robots at the beginning of the experiments.

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

To determine the values for λ_B as given in Table I, in the real robot experiments the travel times of the robots were measured. Fig. 8 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 speed). λ_B was then derived by normalizing the average execution time for using path **B** with respect to the average execution time of using path **A**.

C. Experiments With the Simulation Model

In addition 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 than those that can be studied using the real robots. The simulation model is a simple event-based multiagent model. No representation of the physical environment nor physical interactions between robots are implemented in the simulation model. Instead, robots are represented as simple agents that either execute one of the two possible actions or are in observation state for a fixed time interval.

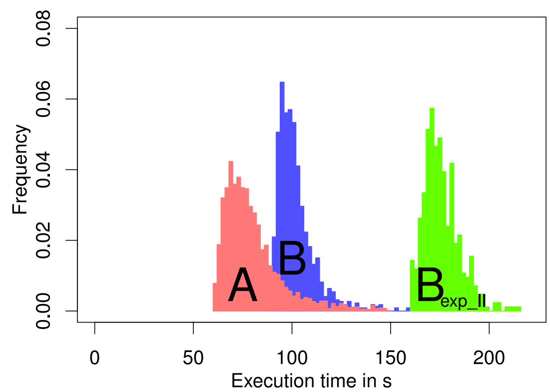


Fig. 8. Distributions of the travel times for path **A**, path **B**, and path **B** in experiment II, recorded in the real robot experiments and used for the simulation model.

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 V-B).

In the simulation model, the time robots stay in the observation state is set to 20 s 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 15 s as in the real robot experiments. The default value for the memory size is $k = 2$.

D. Performance Metric

In order to quantify how well the swarm performs in finding the fastest action for a given parameter combination, we measure the fraction of experimental runs that end in consensus on action **A** (the fastest action). We call this the accuracy of the decision-making method.

In the 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 95% confidence interval for 10 000 trials 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.

VI. RESULTS AND DISCUSSION

In the following, we present the results of our real robot experiments and compare them to the results of the simulation model. We show that the simulation model resembles the real robot experiments closely. However, we also point out differences and discuss their causes. Moreover, we present

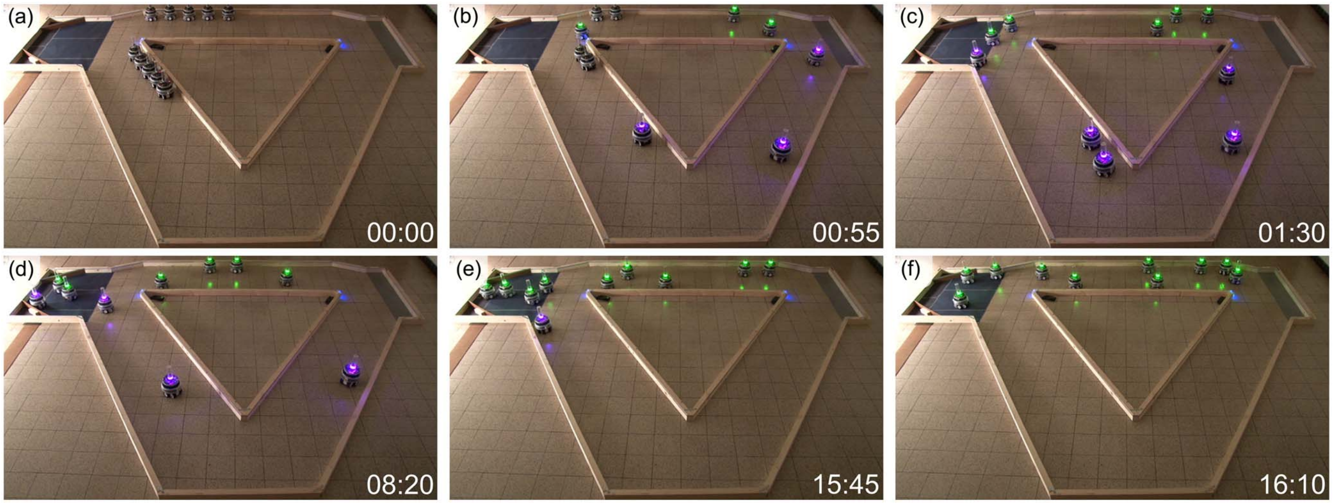
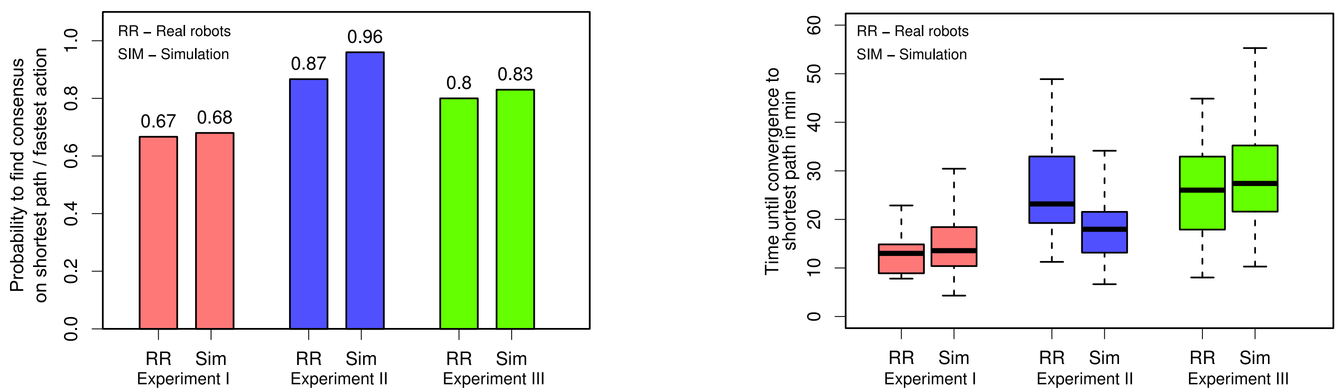


Fig. 9. Example of a typical run of a real robot experiment. (a) At the beginning of the experiment, robots are equally distributed between the two paths. (b) Robots start in pairs to avoid the formation of robot clusters. (c)–(e) Robots successively switch to the shortest path **A**. (f) Swarm has converged to the shortest path **A**. Time is indicated in minutes:seconds.



(a) (b)

Fig. 10. Summary of the experimental results with ten robots. Experiment I: $\lambda_B \approx 1.3$ 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 $\lambda_B \approx 2.3$ led to 13 successful runs but also doubles the time needed to converge. Experiment III: increasing memory size to $k = 4$ resulted in 12 runs that converged to **A** and in a strongly increased convergence time. (a) Probability to find consensus on shortest path. (b) Time until convergence to shortest path.

simulation experiments that were conducted to investigate system parameters that exceed the possibilities of real robot experiments.

A. *Comparison of Real-World and Simulation Experiments*

Fig. 9 shows different stages of a typical run of Experiment I.[1] Initially the robots start in two queues of five robots each, placed on the two paths [Fig. 9(a)]. The robots start to move to the source zone in pairs [Fig. 9(b)]. Robots with opinion B switch to opinion A [Fig. 9(c)–(e)]. Eventually, the swarm converges to the shortest path [Fig. 9(f)]. The depicted experimental run took 16 min.

Fig. 10 shows the results of all real robot experiments and compares them to the results of the simulation experiments. In Experiment I (memory size $k = 2$) 10 out of 15 runs successfully converged to the short path **A** (accuracy 0.67). This is in accordance with the simulation model where an accuracy of 0.68 is reached, and with the analytical model that would predict an accuracy of 0.66 (see Fig. S2 in supplementary material [38]).

The time the system needs to converge to the shortest path is also predicted well by the simulation model [see Fig. 10(b)]. Single experiments last from 9 min minimum up to 24 min maximum. On average it takes approximately 15 min to find the shortest path.

The goal of Experiment II is to investigate the influence of the ratio between the action execution times on the accuracy of the method. The analytical model predicts that increasing the difference between the action execution times increases the bias in the observed opinions and therefore increases the accuracy of the method. The results of real robot Experiment II are in accordance with this prediction [see Fig. 10(a)]: 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 difference in accuracy and convergence time between simulation and real robot experiments for Experiment II

[1]See also the supplementary material for a video recording of an experimental run [38].

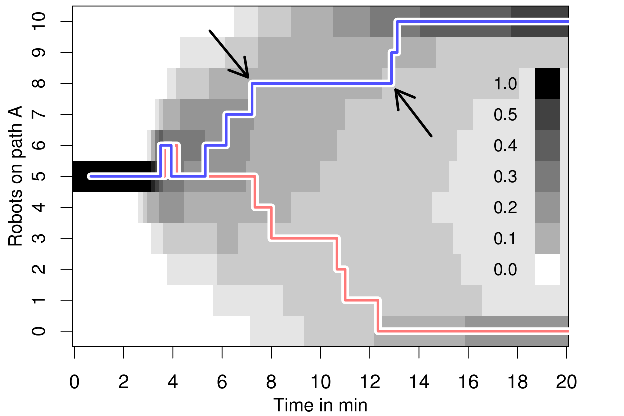


Fig. 11. Distribution of robots over time collected over 50 000 simulations of Experiment I. The shade of gray indicates the probability to find a certain number of robots with opinion **A** at a given time in the system. The two lines correspond to data collected in two real robot experiments.

[see Fig. 10(b)] is due to a small bias toward opinion **B** in the experimental setup. Robots that take path **B** slow down immediately after they leave the observation zone. This causes them to remain within the range of the omnidirectional camera of the robots in the observation zone for a longer time than robots that take path **A**. This bias introduced by this behavior is not present in the simulation model as physical interactions are not taken into account. 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. Clearly, altering the setup of Experiment II by increasing the speed of robots on path **A** instead of slowing down robots on path **B** would lead to the same bias toward **A** because robots with opinion **A** will leave the observation distance of the robots in the decision zone faster.

In Experiment III, we increase the memory size k from 2 to 4. As predicted by the theoretical model, this leads to an increased accuracy at the cost of an increased convergence time [see Fig. 10(a)]: 12 out of 15 runs converged to the shortest path (accuracy 0.8).

Fig. 11 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 predicted by the simulation model. The two lines show data collected in two different runs of the real robot experiments.

The arrows in Fig. 11 mark an observation in the real robot experiments that can rarely happen in simulation. From 7 to 13 min the number of robots with opinion **A** remains constant although only two robots with opinion **B** are left. The reason is that these two robots moved closely together in the arena as a pair. When the pair entered the observation zone, with a high probability the two robots observed each other and therefore did not change opinion. However, eventually the pair dissolved and the system converged.

The real robot experiments showed that the decision-making method can also cope with the presence of errors. We found two main sources of such errors in our real robot experiments.

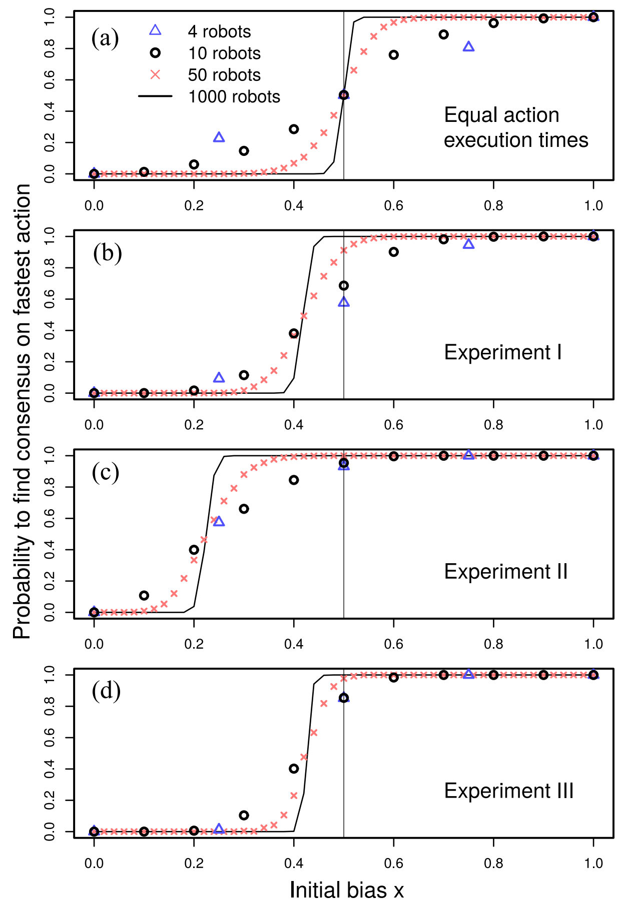


Fig. 12. Probability that the swarm finds the shortest path as a function of the initial bias for different swarm sizes. (a) Equal action execution times. (b) Experiment I. (c) Experiment II. (d) Experiment III.

First, 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. Second, small traffic jams and noise in the ground sensors can lead to a high variance in the task execution times. Moreover, the well-mixed assumption might be violated due to correlations between robots, for example, robots travelling in small clusters. However, the method always converged and with high probability consensus was found on the shortest path.

B. Extended Analysis Using the Simulation Model

1) Swarm Size and Initial Bias: To further study the influence of the swarm size we simulate experiments I, II, and III with different swarm sizes (4, 10, 50, and 1000 robots). Moreover, we investigate the influence of the initial bias, that is, of the fraction of robots that start with opinion **A**. In accordance with Section IV, we denote the initial bias with x and the fraction corresponding to a random experimental outcome (the critical fraction) with x_c.

As can be seen in Fig. 12(a), for equal task execution times ($\lambda_B = 1$), as predicted by the analytical model, swarms tend to reach consensus on the opinion that was initially favored by the majority of robots. Also in accordance with the analytical model, the simulation shows that the critical bias x_c in Experiment I is smaller than 0.5 [Fig. 12(b)].

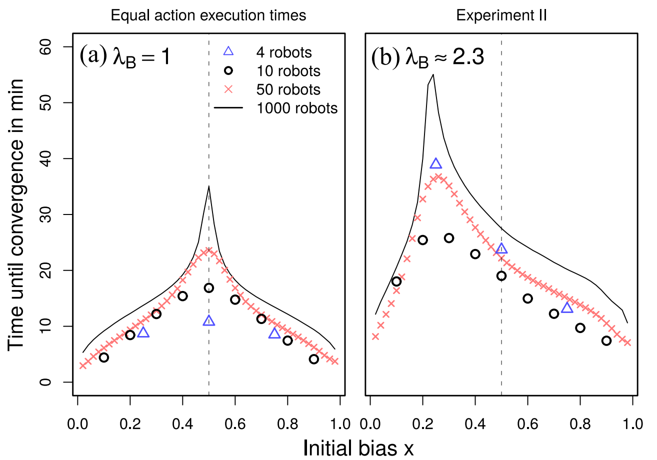


Fig. 13. Time to convergence as a function of the initial bias for different swarm sizes. (a) $\lambda_B = 1$. (b) $\lambda_B \approx 2.3$.

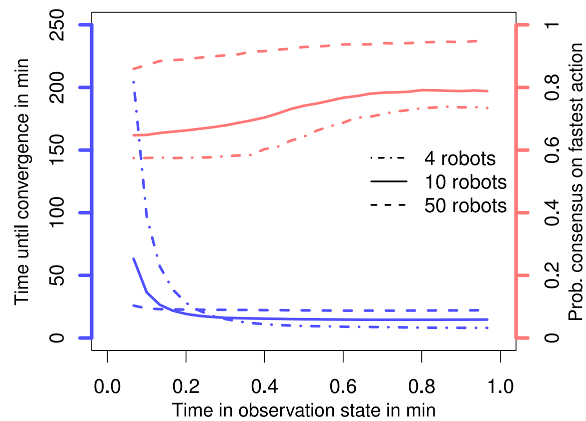


Fig. 14. Dependence of the time until convergence (darker color) and the probability to find consensus on the fastest action (lighter color) on the time the robots stay in the observation state.

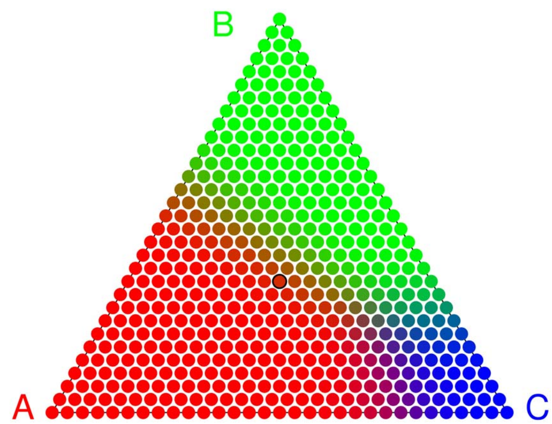


Fig. 15. Consensus opinion for a swarm of $N = 30$ robots with $m = 3$ different opinions and execution time ratios $\lambda_A = 1,\ \lambda_B = 2,\ \lambda_C = 4$.

More precisely, since in Experiment I the mean execution time ratio is $\lambda_B \approx 1.3$ 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. For instance, large swarms of 1000 robots found consensus on the fastest action in all conducted experiments.

In Experiment II, the longer execution time of action **B** results in a strong shift of the critical bias [Fig. 12(c)]. In Experiment III, the larger memory results in a stronger positive feedback [Fig. 12(d)]. Consequently, in both experiments, the probability for unbiased swarms to find consensus on the fastest action **A** is higher than in Experiment I.

Fig. 13 depicts the measured time to find consensus depending on the initial bias x. The maximal convergence times can be found near the critical values, because here the drift toward the consensus states induced by the k-unanimity rule is low. However, it can also be seen that the time to convergence grows only 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.

2) Time in Observation State: The observation zone in our double-bridge experiment has a fixed size. Thus, the time robots remain in the observation state is also fixed. However, in a different application of the decision method the time robots remain in the observation state might be adjustable. Therefore, we investigate the influence of the duration of the observation state in our simulation model (Fig. 14).

If a robot remains only a short time period in the observation state, it is unlikely that it will observe another robot. Since robots observe themselves if there are no other robots around, the probability to switch opinion is small. Consequently, the time needed to find consensus can be very long. Moreover, the influence of random fluctuations increases and this lowers the accuracy of the decision-making method. However, for larger swarms the mentioned effects disappear, as more robots will be in the observation state and therefore the probability of not observing other robots will become smaller.

The results show that time in observation state should exceed a certain value (0.2) to lead to a reasonable time until convergence. However, the figure also shows that the longer robots stay in observation state, the higher the probability for consensus on the fastest action.

3) Memory Size k: Both analytical models predict that increasing the memory size k increases the accuracy of the decision-making method at the cost of longer convergence times. The real robot Experiment III and the corresponding simulations confirm this finding, too. To investigate the influence of the memory size further, we simulate swarms of size $N \in \{4, 10, 50\}$ that use memory sizes $k \in \{2, \ldots, 8\}$. As expected, the results show a trade-off between accuracy and time needed to converge (see supplementary material [38]). The shape of the curves are similar to the predictions of the analytical model. However, for small swarm sizes, and in contrast to the analytical model, the simulation model predicts very large convergence times. Furthermore, when using small swarms, the observation zone is often empty. Accordingly, the probability that a robot encounters other robots k times in a row becomes very small for large k. For example, for swarms of four robots that use a memory of size $k = 8$, it takes more than two simulated days on average to converge to a decision.

4) Number of Actions m: Fig. 15 visualizes the outcome of the decision-making method for a swarm of $N = 30$ robots with $m = 3$ different actions and execution time ratios $\lambda_A = 1,\ \lambda_B = 2,\ \lambda_C = 4$. As in Fig. 3, obtained with the continuum model, the position of a point in the figure corresponds to certain fractions of robots for the three opinions. The color of the point gives the final outcome of the decision-making process when the swarm is initialized according to these fractions. The central point of the triangle corresponds to an unbiased swarm, that is, a swarm where ten robots start with opinion **A**, ten with **B**, and ten with C. As it can be seen, also in the case of $m = 3$ opinions, unbiased swarms tend to find consensus on the fastest action **A** with high probability.

VII. 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 corresponds to 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 the shortest 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 the opinion of other robots. As such, only the indirect and anonymous communication of opinions is necessary. The experiments with real robots showed that the method also works in the presence of errors due to sensor noise or robot failures.

This paper shows that the accuracy of the decision-making method is determined by a number of different factors. First, 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 over small swarms. Second, the accuracy is higher the more the two actions differ in execution time. Last, increasing the size of the observation memory *k* also leads to higher accuracy of the method. The value of *k* can be adjusted to regulate the accuracy of the decision-making at the cost of a longer convergence time.

Regarding future research, we believe that there are several promising directions. First, an experiment using real robots in a setting with more than two actions would complement the theoretical results brought forwards in this paper. Second, using larger swarms of smaller robots could shed some light on the question whether the proposed method is robust to changes in the type of physical interference. Third, and maybe most interesting, one could weaken the *k*-unanimity rule in favor of a quorum-based rule.

References

[1] M. Dorigo and E. Şahin, "Guest editorial. Special issue: Swarm robotics," *Auton. Robots*, vol. 17, nos. 2–3, pp. 111–113, 2004.

[2] M. Dorigo, M. Birattari, and M. Brambilla, "Swarm robotics," *Scholarpedia*, vol. 9, no. 1, p. 1463, 2014.

[3] M. Brambilla, E. Ferrante, M. Birattari, and M. Dorigo, "Swarm robotics: A review from the swarm engineering perspective," *Swarm Intell.*, vol. 7, no. 1, pp. 1–41, 2013.

[4] A. Gutiérrez, A. Campo, F. C. Santos, F. Monasterio-Huelin, and M. Dorigo, "Social odometry: Imitation based odometry in collective robotics," *Int. J. Adv. Robot. Syst.*, vol. 26, no. 2, pp. 129–136, 2009.

[5] C. A. C. Parker and H. Zhang, "Collective unary decision-making by decentralized multiple-robot systems applied to the task-sequencing problem," *Swarm Intell.*, vol. 4, no. 3, pp. 199–220, 2010.

[6] F. Ducatelle, G. Di Caro, C. Pinciroli, and L. M. Gambardella, "Self-organized cooperation between robotic swarms," *Swarm Intell.*, vol. 5, no. 2, pp. 73–96, 2011.

[7] D. Berry and B. Fristedt, *Bandit Problems: Sequential Allocation of Experiments* (Monographs on Statistics and Applied Probability Series). London, U.K.: Chapman and Hall, 1985.

[8] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Mach. Learn.*, vol. 47, nos. 2–3, pp. 235–256, 2002.

[9] J. Wessnitzer and C. Melhuish, "Collective decision-making and behaviour transitions in distributed ad hoc wireless networks of mobile robots: Target-hunting," in *Proc. Eur. Conf. Artif. Life (ECAL)*, Dortmund, Germany, 2003, pp. 893–902.

[10] C. A. C. Parker and H. Zhang, "Cooperative decision-making in decentralized multiple-robot systems: The best-of-N problem," *IEEE/ASME Trans. Mechatronics*, vol. 14, no. 2, pp. 240–251, Apr. 2009.

[11] R. A. Russell, "Heat trails as short-lived navigational markers for mobile robots," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, vol. 4. Albuquerque, NM, USA, 1997, pp. 3534–3539.

[12] R. A. Russell, "Ant trails—An example for robots to follow?" in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Detroit, MI, USA, 1999, pp. 2968–2703.

[13] R. Fujisawa, S. Dobata, D. Kubota, H. Imamura, and F. Matsuno, "Dependency by concentration of pheromone trail for multiple robots," in *Proc. 6th Int. Conf. Ant Colony Optim. Swarm Intell. (ANTS)*, Brussels, Belgium, 2008, pp. 283–290.

[14] R. Fujisawa, S. Dobata, K. Sugawara, and F. Matsuno, "Designing pheromone communication in swarm robotics: Group foraging behavior mediated by chemical substance," *Swarm Intell.*, vol. 8, no. 3, pp. 227–246, 2014.

[15] R. Mayet, J. Roberz, T. Schmickl, and K. Crailsheim, "Antbots: A feasible visual emulation of pheromone trails for swarm robots," in *Proc. 7th Int. Conf. Swarm Intell.*, Brussels, Belgium, 2010, pp. 84–94.

[16] K. Sugawara, T. Kazama, and T. Watanabe, "Foraging behavior of interacting robots with virtual pheromone," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, vol. 3. Sendai, Japan, 2004, pp. 3074–3079.

[17] 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," in *Proc. IEEE Swarm Intell. Symp. (SIS)*, Honolulu, HI, USA, 2007, pp. 37–44.

[18] H. Hamann, M. Szymanski, and H. Wörn, "Orientation in a trail network by exploiting its geometry for swarm robotics," in *Proc. IEEE Swarm Intell. Symp. (SIS)*, Honolulu, HI, USA, 2007, pp. 310–315.

[19] K. J. Hara and T. R. Balch, "Pervasive sensor-less networks for cooperative multi-robot tasks," in *Distributed Autonomous Robotic Systems 6*. Tokyo, Japan: Springer, 2007, pp. 305–314.

[20] C. M. Vigorito, "Distributed path planning for mobile robots using a swarm of interacting reinforcement learners," in *Proc. 6th Int. Joint Conf. Auton. Agents Multiagent Syst. (AAMAS)*, Honolulu, HI, USA, 2007, pp. 120:1–120:8.

[21] M. Mamei and F. Zambonelli, "Physical deployment of digital pheromones through RFID technology," in *Proc. Int. Joint Conf. Auton. Agents Multiagent Syst. (AAMAS)*, Pasadena, CA, USA, 2005, pp. 1353–1354.

[22] H. Herianto and D. Kurabayashi, "Realization of an artificial pheromone system in random data carriers using RFID tags for autonomous navigation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Kobe, Japan, 2009, pp. 2288–2293.

[23] A. A. Khaliq, M. Di Rocco, and A. Saffiotti, "Stigmergic algorithms for multiple minimalistic robots on an RFID floor," *Swarm Intell.*, vol. 8, no. 3, pp. 199–225, 2014.

[24] D. Payton, M. Daily, R. Estowski, M. Howard, and C. Lee, "Pheromone robotics," *Auton. Robots*, vol. 11, no. 3, pp. 319–324, 2001.

[25] S. Nouyan, R. Groß, M. Bonani, F. Mondada, and M. Dorigo, "Teamwork in self-organized robot colonies," *IEEE Trans. Evol. Comput.*, vol. 13, no. 4, pp. 695–711, Aug. 2009.

[26] A. Campo *et al.*, "Artificial pheromone for path selection by a foraging swarm of robots," *Biol. Cybern.*, vol. 103, no. 5, pp. 339–352, 2010.

[27] S. Garnier, J. Gautrais, M. Asadpour, C. Jost, and G. Theraulaz, "Self-organized aggregation triggers collective decision making in a group of cockroach-like robots," *Adapt. Behav.*, vol. 17, no. 2, pp. 109–133, 2009.

[28] A. Campo, S. Garnier, O. Dédriche, M. Zekkri, and M. Dorigo, "Self-organized discrimination of resources," *PLoS ONE*, vol. 6, no. 5, 2011, Art. ID e19888.

[29] M. A. Montes de Oca, E. Ferrante, A. Scheidler, M. Birattari, and M. Dorigo, "Opinion formation models with differential latency: A self-organized collective decision-making mechanism," *Swarm Intell.*, vol. 5, nos. 3–4, pp. 305–327, 2011.

[30] G. Valentini, H. Hamann, and M. Dorigo, "Efficient decision-making in a self-organizing robot swarm: On the speed versus accuracy trade-off," in *Proc. 14th Int. Conf. Auton. Agents Multiagent Syst. (AAMAS)*, Istanbul, Turkey, 2015, pp. 1305–1314.

[31] A. Scheidler, "Dynamics of majority rule with differential latencies," *Phys. Rev. E*, vol. 83, no. 3, 2011, Art. ID 031116.

[32] H. Hamann and H. Wörn, "A framework of space-time continuous models for algorithm design in swarm robotics," *Swarm Intell.*, vol. 2, nos. 2–4, pp. 209–239, 2008.

[33] S. Berman, V. Kumar, and R. Nagpal, "Design of control policies for spatially inhomogeneous robot swarms with application to commercial pollination," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Shanghai, China, 2011, pp. 378–385.

[34] M. A. Montes de Oca, E. Ferrante, A. Scheidler, and L. F. Rossi, "Binary consensus via exponential smoothing," in *Proc. 2nd Int. Conf. Complex Sci. Theory Appl. (COMPLEX)*, Santa Fe, NM, USA, 2012, pp. 244–255.

[35] G. Pini, M. Gagliolo, A. Brutschy, M. Dorigo, and M. Birattari, "Task partitioning in a robot swarm: A study on the effect of communication," *Swarm Intell.*, vol. 7, nos. 2–3, pp. 173–199, 2013.

[36] C. Castellano, S. Fortunato, and V. Loreto, "Statistical physics of social dynamics," *Rev. Modern Phys.*, vol. 81, no. 2, pp. 591–646, 2009.

[37] P. Clifford and A. Sudbury, "A model for spatial conflict," *Biometrika*, vol. 60, no. 3, pp. 581–588, 1973.

[38] A. Scheidler, A. Brutschy, E. Ferrante, and M. Dorigo. (2013). *Supporting material for the article: The k-Unanimity Rule for Self-organized Decision Making in Swarms of Robots*. [Online]. Available: http://iridia.ulb.ac.be/supp/IridiaSupp2011-020/

[39] S. Goss, S. Aron, J.-L. Deneubourg, and J. M. Pasteels, "Self-organized shortcuts in the Argentine ant," *Naturwissenschaften*, vol. 76, no. 12, pp. 579–581, 1989.

[40] A. Winfield, "Towards an engineering science of robot foraging," in *Distributed Autonomous Robotic Systems 8*, J. Ota, H. Asama, H. Kurokawa, and K. Sekiyama, Eds. Berlin, Germany: Springer, 2010, pp. 185–192.

[41] M. Bonani *et al.*, "The marXbot, a miniature mobile robot opening new perspectives for the collective-robotic research," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*. Taipei, Taiwan, 2010, pp. 4187–4193.

[42] M. Dorigo *et al.*, "Swarmanoid: A novel concept for the study of heterogeneous robotic swarms," *IEEE Robot. Autom. Mag.*, vol. 20, no. 4, pp. 60–71, Dec. 2013.

Alexander Scheidler received the Diploma and Ph.D. degrees in computer science from the University of Leipzig, Leipzig, Germany, in 2005 and 2010, respectively.

He was a Post-Doctoral Fellow with IRIDIA, Université Libre de Bruxelles, Brussels, Belgium, from 2010 to 2012 and has been with the Fraunhofer Institute for Wind Energy and Energy System Technology, Kassel, Germany, since 2012. His current research interests include swarm intelligence, optimization methods for distribution grid planning, and HPC in power system research.

Arne Brutschy received the Diploma degree in computer science from the University of Leipzig, Leipzig, Germany, and the Ph.D. degree in applied sciences from IRIDIA, Université Libre de Bruxelles, Brussels, Belgium, in 2014.

His current research interests include artificial intelligence and swarm robotics, with a focus on selforganized task allocation and task partitioning.

Mr. Brutschy was a recipient of an "Aspirant" scholarship from the Funds for Scientific Research F.R.S.-FNRS of the Belgian French Community.

Eliseo Ferrante received the B.Sc. degree in computer science engineering from the Politecnico di Milano, Milano, Italy, in 2004, the M.Sc. degree in computer science engineering from the Politecnico di Milano, in 2007, the M.Sc. degree in computer science from the University of Illinois at Chicago, Chicago, IL, USA, in 2007, and the Ph.D. degree in applied sciences from Université Libre de Bruxelles, Brussels, Belgium, in 2013.

From 2012 to 2014, he was a Research Associate with Katholieke Universiteit Leuven, Leuven, Belgium. Since 2014, he has been a FWO Post-Doctoral Researcher with the Department of Biology, Katholieke Universiteit Leuven. He was an Interdisciplinary Researcher interested in self-organization and its interplay with evolution. His expertise ranges from swarm robotics, to statistical physics, evolutionary biology, and optimization.

Marco Dorigo (S'92–M'93–SM'96–F'06) received the Laurea, Master of Technology, degree in industrial technologies engineering in 1986, and the Ph.D. degree in electronic engineering in 1992 from the Politecnico di Milano, Milan, Italy, and the title of Agrégé de l'Enseignement Supérieur, from Université Libre de Bruxelles (ULB), Brussels, Belgium, in 1995.

From 1992 to 1993, he was a Research Fellow with the International Computer Science Institute, Berkeley, CA, USA. In 1993, he was a NATO-CNR Fellow, and from 1994 to 1996, a Marie Curie Fellow. Since 1996, he has been a tenured Researcher of the FNRS, the Belgian National Funds for Scientific Research, and the Co-Director of IRIDIA, ULB. He is the inventor of the Ant Colony Optimization metaheuristic. His current research interests include swarm intelligence, swarm robotics, and metaheuristics for discrete optimization. He is Editor-in-Chief of Swarm Intelligence, and an Associate Editor or a member of the Editorial Boards of several journals on computational intelligence and adaptive systems.

Prof. Dorigo was a recipient of the Italian Prize for Artificial Intelligence in 1996, the Marie Curie Excellence Award in 2003, the Dr. A. De Leeuw-Damry-Bourlart Award in Applied Sciences in 2005, the Cajastur International Prize for Soft Computing in 2007, an ERC Advanced Grant in 2010, and the IEEE Frank Rosenblatt Award in 2015. He is a fellow of AAAI and ECCAI.