Article

CIMAX: collective information maximization in robotic swarms using local communication

**Hannes Hornischer[1,2] , Joshua Cherian Varughese[1,3],
Ronald Thenius[1], Franz Wotawa[3], Manfred Füllsack[2]
and Thomas Schmickl[1]**

Abstract
Robotic swarms and mobile sensor networks are used for environmental monitoring in various domains and areas of operation. Especially in otherwise inaccessible environments, decentralized robotic swarms can be advantageous due to their high spatial resolution of measurements and resilience to failure of individuals in the swarm. However, such robotic swarms might need to be able to compensate misplacement during deployment or adapt to dynamical changes in the environment. Reaching a collective decision in a swarm with limited communication abilities without a central entity serving as decision-maker can be a challenging task. Here, we present the CIMAX algorithm for collective decision-making for maximizing the information gathered by the swarm as a whole. Agents negotiate based on their individual sensor readings and ultimately make a decision for collectively moving in a particular direction so that the swarm as a whole increases the amount of relevant measurements and thus accessible information. We use both simulation and real robotic experiments for presenting, testing, and validating our algorithm. CIMAX is designed to be used in underwater swarm robots for troubleshooting an oxygen depletion phenomenon known as "anoxia."

1. Introduction

Swarms of various lifeforms have been observed to utilize emergent group dynamics (Eberhart et al., 2001) for various tasks such as foraging (Seeley, 1992), reproduction (Bonner, 1949; Durston, 1973), or escaping predators (Brock & Riffenburgh, 1960; Cavagna et al., 2010; Magurran & Pitcher, 1987). Seeley (1992) discovered how bees use waggle dances for foraging by pointing their hive to high-quality food sources. Bonner (1949) and Durston (1973) examined the communication behavior of slime mold cells which despite its simplicity enables self-organization with respect to foraging, reproduction, and so on. Cavagna et al. (2010) analyzed how starling flocks respond to external stimuli as a collective to evade predators. Due to the availability of many eyes in a swarm, each individual spends less time and energy on scanning the environment for predators. Magurran and Pitcher (1987)

experimentally demonstrated various formations used by shoals of minnows when detecting predators. Decentralized intelligence of such kind is popularly known as swarm intelligence (Beni & Wang, 1989). Natural systems exhibiting this decentralized intelligence have inspired researchers due to their adaptability to the environment, resilience to perturbations, and underlying simplicity. Despite simple rules governing

[1]Artificial Life Laboratory, Department of Zoology, Institute of Biology, University of Graz, Graz, Austria
[2]Institute of Systems Sciences, Innovation and Sustainability Research, University of Graz, Graz, Austria
[3]Institute for Software Technology, Graz University of Technology, Graz, Austria

Corresponding author:
Hannes Hornischer, Artificial Life Laboratory, Department of Zoology, Institute of Biology, University of Graz, Merangasse 12, 8010 Graz, Austria.
Email: hannes.hornischer@edu.uni-graz.at

the behavior of individuals in a swarm, the resulting collective behavior often shows a stunning degree of complexity—as it can also be observed in the synchronized flashing of the fireflies *lampyridae* (Buck, 1988), predator evasion in fish (Magurran and Pitcher, 1987), and so on. Researchers in many emerging fields such as ubiquitous computing (Kim & Follmer, 2017), multi-robot systems (Kernbach et al., 2009; Zahadat & Schmickl, 2016), and traffic management (Renfrew & Yu, 2009) have recognized parallels between such multi-agent artificial systems and natural systems containing several actors (Fernandez-Marquez et al., 2013; Garnier et al., 2007). As a result, extensive research has been dedicated to self-organization and decentralization in complex systems (Dorigo et al., 2004). When designing swarms or sensor networks, one challenge that often needs to be addressed is the collective decision-making task (Kernbach et al., 2013).

In this article, we present an algorithm enabling a swarm of individuals with limited communication abilities to collectively maximize the amount of information available to the swarm as a whole. We use the variance in measurements of the swarm as a measure of information and use it in combination with a simple bioinspired communication mechanism to let the agents determine which direction yields the largest increase in information. In contrast to centralized swarms, individual agents use only local information and do not have access to a central entity. Hereafter, we refer to the algorithm as CIMAX.

2. Motivation

We initially designed CIMAX to address the task of documenting, examining, and ultimately forecasting the frequently but irregularly appearing anoxic waters phenomena (Runca et al., 1996) in the lagoon of Venice. During this phenomenon which is commonly referred to as "anoxia," the oxygen content of a small part of the lagoon decreases dramatically resulting in the death of animal life in that specific area. Anoxia adversely affects the flora and fauna in the lagoon and consequently may cause difficulties for the living conditions of the inhabitants and tourists in Venice, for example, due to massive amounts of decomposing dead fish in canals. A strategy to examine and document this phenomenon is to utilize a swarm of underwater robots for monitoring a set of environmental parameters such as oxygen concentration levels. For determining dynamics and spatio-temporal evolution of anoxic areas in the water body, a swarm of robots allows monitoring at various underwater locations and thus high spatial resolution. One implementation of such an artificial swarm using the "aMussel" robots (Donati et al., 2017), used for autonomous long-term underwater monitoring, was developed and extensively tested in

real-world marine environments within the project "subCULTron" (2015).

Due to problems such as expensive hardware (Akyildiz et al., 2005), high power consumption (Stojanovic & Preisig, 2009), and general complexity of communication underwater (Lanbo et al., 2008), the main approach for communication within members of a swarm of aMussels is based on using modulated light for local information transfer. Generally, when deploying a swarm at a target location, it is not guaranteed that the location is effectively covered. It is possible that only few robots are in contact with the anoxic area and the majority of the swarm is not. Moreover, even in case the swarm is optimally placed, the target area is dynamic and hence mobile. As we present in this work, the use of variance as a measure for information enables a swarm to navigate to the border of, for example, an anoxic area and thus monitor the transition where anoxic waters interact with the surrounding environment. The collected data potentially provide insight into the processes connected with the spreading of anoxic domains. Since variance is a relatively simple measure, independent of specific scenarios or environments, it does not restrain the application of this algorithm to various contexts.

3. Related work

The problem that CIMAX addresses in this article is a classical problem of collective decision-making in multi-agent systems where individual entities might make conflicting decisions based on local information. According to Trianni and Campo (2015), algorithms for collective decision-making in natural and artificial swarms can be categorized into three main mechanisms. In the first mechanism, the swarm waits for one entity to have enough information to make a decision and then propagate that decision within the swarm. Organizational structures following this mechanism can be found in form of hierarchies within animal and human societies (Ahl & Allen, 1996; Rabb et al., 1967). The second mechanism is called opinion averaging in which all individuals constantly adjust their own opinion based on their neighbors' opinions until the entire swarm eventually converges to one opinion. This mechanism for collective decision-making in robotic swarms can also be found in groups of animals which use it for effectively navigating as a collective (Codling et al., 2007; Conradt & Roper, 2005; Simons, 2004). The third mechanism is based on amplification of a particular opinion to produce a collective decision. In this mechanism, each individual randomly starts with an opinion and then changes their opinion to the other opinions value (in contrast to averaging the opinions, as done in the second mechanism) depending on how often they hear the latter opinion. The amplification

mechanism is also found within animals such as the pheromone trails selection in ants (Beckers et al., 1990; Dorigo et al., 2006), the temperature-based site selection of young bees (Garnier et al., 2009; Szopek et al., 2013), or consensus decisions in group of animals about timing of activities and travel destinations (Conradt & Roper, 2005). A version of this mechanism based on blockchain technology was used for enabling robots to reach collective decisions (Strobel et al., 2018).

Apart from those three main categories, there are more approaches to collective decision-making which are not necessarily observed in natural swarms but implemented on artificial swarms. Some decision-making processes in multi-agent systems, which are often applied to problems of distributed task allocation, rely on auction-based algorithms where agents make bids to an auctioneer (Dias et al., 2006; Koenig et al., 2010). Also decentralized approaches, where all agents constitute both bidder and auctioneer, have been proposed (Vail & Veloso, 2003). An entirely different direction to approach the problem of collective decision-making is the use of neural networks, or more precisely, distributed neural networks where each agent constitutes a slice of a "collective mind" (Otte, 2018).

The underlying mechanism of collective decision-making of the CIMAX algorithm relies on the averaging over opinions within the swarm which is associated with the second category of mechanisms presented in Trianni and Campo (2015).

Apart from collective decision-making in swarm robotics, our approach is broadly related to relocation of sensor nodes in mobile wireless sensor networks ("MWSN") (Cui et al., 2004; Li et al., 2007; Wang et al., 2005). For example, when deploying a swarm in an otherwise inaccessible environment, the swarm is often not arranged properly for effective measurement and observation due to inaccurate knowledge of target area, of dynamic changes in local conditions, or of unforeseeable events. For optimizing parameters such as coverage, connectivity, or network longevity, it is necessary to determine how the network or individual members need to be relocated. A variety of approaches has been suggested to accomplish effective relocation.

While some approaches in sensor relocation rely on having access to global information of the position of sensors, often this problem is approached in a decentralized manner. In Wang et al. (2005), the exact position of the sensors is known by the base station or similar central entity. The area covered by the sensors is increased while minimizing the travel time and the distance traveled using genetic algorithms. Such a system is used to compensate for coverage loss when sensors fail in the field. In Cui et al. (2004) and Li et al. (2007), a more decentralized approach for relocation of sensors is followed to maintain coverage of a sensor network. The sensors periodically broadcast their locations and identifiers to their neighbors and construct a Voronoi diagram. Voronoi polygons are computed using the received information. Once a node finds a hole in the Voronoi diagram, that is, a relatively large polygon, the relocation of a sensor is initiated. In Cui et al. (2004), a simulation of an odor localization scenario with a group of mobile robots is presented. The authors focus on using fuzzy logic to decide which direction to move to to eventually localize the source. They assume that measurements from each agent are easily available to other agents in the swarm wherefore the agent-to-agent communication aspect is not adequately addressed.

In addition, there are numerous approaches in the field of exploration and information gathering through the utilization of multiple robots. In the following paragraph, we give a brief overview of the different approaches. Schwager et al. (2017) present an approach based on the fundamentals of information theory for information gathering under balancing expected information gain and risk of damage or failure due to local hazards. Erignac (2007) presents an approach for exploration in context of search strategies under constraints in communication. Their algorithm is designed for swarms of unmanned aircraft vehicles and is inspired by the pheromone map-based foraging strategies of some insects. Zarzhitsky et al. (2005) use small decentralized robot swarms for tracing chemical plumes. In their approach, all agents individually make a decision based on local information and information exchanged with neighbors. The swarm-wide decision-making process is implicitly done, since the individual decision results in an emergent collective movement. Smith et al. (2016) present an approach for adaptive tracking of an ocean front using autonomous underwater vehicles. This is closely related to the task the CIMAX algorithm is designed for; however, their approach involves long range communication and involves a heterogeneous team of robots. Marthaler and Bertozzi (2003) present an algorithm enabling autonomous swarms to determine static environmental boundaries using both local and/or global communication. Their algorithm is originally used in image processing and is based on energy minimizing curves.

The CIMAX algorithm differs from existing approaches in the following ways: (1) the swarm does not require access to global information—there is no need for a central entity knowing the positions of all sensors (2) nor are agents required to receive instructions or be organized by a central entity. The loss of individual agents generally does not affect the swarm's functionality. (3) CIMAX maximizes the diversity or variance of measurements collected by the swarm as a whole and by that relies on a simple and robust measure. (4) Our approach utilizes not only the content of received signals but also their properties.

We present both numerical simulation and robotic experiments to validate the presented method. Furthermore, this algorithm can be embedded into the

swarm design paradigm WOSP (Thenius et al., 2018), a framework for controlling swarms utilizing the communication mechanism we briefly introduce in section 4.1. In section 4.2., we present the algorithm and its implementation. The computational results and theoretical analysis of the algorithm are shown in section 5, including numerical simulations in the aforementioned target environment and scenario. In section 6, we present the experimental setup and results which are then discussed in section 7.

4. The CIMAX algorithm

The CIMAX algorithm enables a swarm of individuals with limited communication abilities to collectively determine which direction yields the maximum increase in information accessible to the collective. The fundamental communication mechanism presented here is inspired by slime mold (*dictyostelium discoideum*) and fireflies (*lampyridae*) and has previously been used to design various algorithms (Thenius et al., 2018; Varughese et al., 2016).

4.1. Communication paradigm

In this communication paradigm, all agents can enter three different states similar to the behavior of slime mold (*dictyostelium discoideum*): an "inactive" state in which agents are receptive to incoming communication, an "active" state where they send or relay a signal, which is followed by a "refractory" state where agents are temporarily insensitive to incoming signals. From the inactive state, an incoming message or the decision to initiate a message lets an agent transition into the active state. In the active state, an agent either relays an incoming message or initiates a new message. Subsequently agents enter the refractory state, being insensitive to incoming messages for a finite time until transitioning to the inactive state again. This communication mechanism is schematically shown in Figure 1(a), as well as the conceptual operating structure in Figure 1(b). Agents initiate a signal randomly by initially setting a timer within $t_p \in (0, t_p^{max}]$. In this manner, each agent initiates the sending of a signal at least once within a time period t_p^{max} (the maximum a timer can randomly be set to), which we refer to as "cycle."

The three states of agents ultimately allow wave-like propagation of signals through the swarm as shown in Figure 2. In Figure 2(a), almost all agents are in the inactive state, shown in black, except one agent which broadcasts a message which is shown in red. Afterward, it transitions into the refractory state, shown in blue. Neighboring agents receive the signal and switch to the active state as shown in Figure 2(b) and (c). The signal spreads in a wave-like manner. In Figure 2(d), the initiating agent switches from the refractory state into the inactive state again. Due to a fixed duration of the
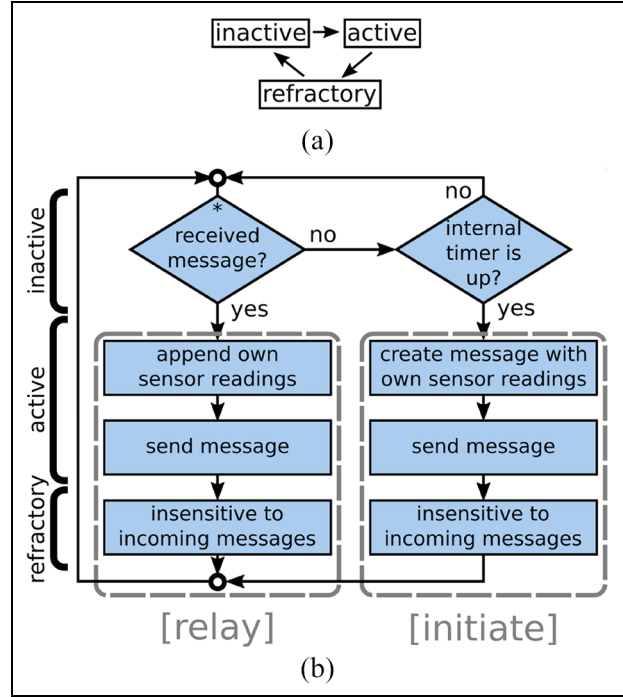

Figure 1. The WOSP communication mechanism. (a) The three-state agents can have. (b) The conceptual operating structure of an agent.

refractory state, the transition to the inactive state as well spreads in a wave-like manner, shown in Figure 2(d) and (e). In Figure 2(f), several trajectories along which the signals were broadcast are shown as red lines. It shows how messages do not necessarily propagate in a straight line, that is, perfectly radially outward from the sender. However, given a well-connected and aggregated swarm, the propagation of signals exhibits a relatively directed trajectory since agents enter the refractory state after broadcasting a signal and thus prevent the occurrence of "sharp turns" of a propagating signal.

Signals are solely received by agents in close neighborhood, that is, within perception range R of the sender (shown in the bottom right corner in Figure 2(a)) and subsequently relayed, thus propagating through the system. After receiving a signal, agents relay it with a delay of one timestep $t_{delay} = 1$ s which we henceforth use as basic unit for time.

The refractory state assures that a signal will neither "flood" a swarm, that is, signals will not (re)activate the initial sender, nor periodically propagate through the swarm, for example, as a spiraling wave. A more detailed discussion on the selection of parameters can be found in Varughese et al. (2020).

4.2. The algorithm

In this section, we present the algorithm for enabling swarms to collectively determine the direction which
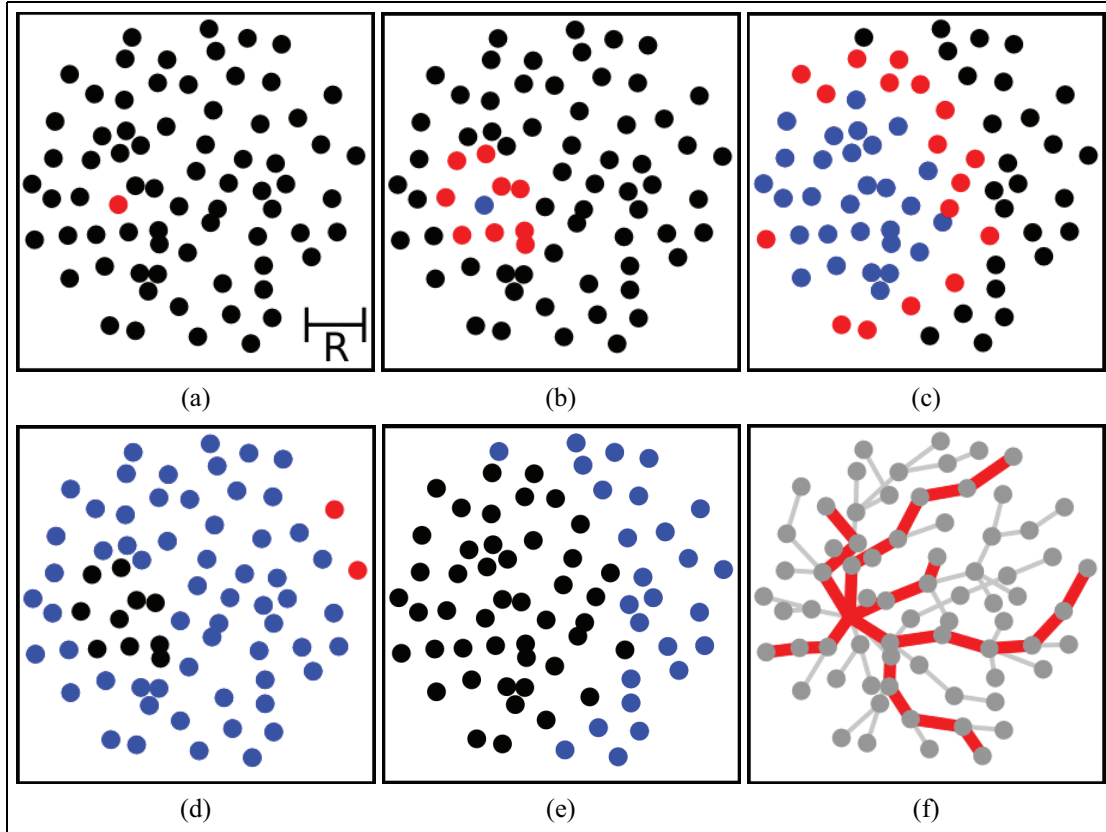
Figure 2. Illustration of wave-based communication. (a–e) A temporal sequence of a signal propagating through a swarm. (f) Several trajectories of signals during signal propagation in panels (a–e), indicated by red lines. Times (s): (a) 0, (b) 2, (c) 4, (d) 11, and (e) 16. Parameters: number of agents $N = 80$, physical size of the swarm in units perception range $R_s = 5R$, refractory time in units timesteps $t_{ref} = 10$ s.

provides the maximum amount of accessible information. We define information as diversity in measurements throughout the swarm, quantified using the variance in measurements. The swarm ultimately detects and favors diverse domains, while it avoids uniform domains which are considered as providing less information. Agents solely need to be able to communicate with nearest neighbors and have a common sense of direction. Besides the direction of an incoming signal, agents do not require any other information about position, orientation, and so on of other agents.

Each agent in the swarm measures the same single quantity X which we use as a generic placeholder for any environmental parameter or quantity measured by swarms. When one agent initiates a message, it sends its own measurement value as a message. Neighboring agents receive the message, append their own measurements, and relay it. This way a message propagating through the system incrementally grows in size with every relay. With this in mind, for easier illustration of the algorithm, we divide the entire procedure into three parts: "information gathering," "evaluation," and "collective decision."

The three parts of the algorithm are exemplary illustrated in Figure 3 for a swarm of $N = 4$ agents. The agents, represented by black circles, are arranged in a line. They are able to measure a quantity X of the environment which is represented by the background colors red and yellow.

Information gathering. Agents randomly (in time) initiate sending a message containing their own sensor readings. Each agent which has received this message stores the received information as well as the direction from which it received the message. Finally, each agent appends its own sensor readings before then broadcasting it to its neighbors. This process is schematically shown in Figure 4, resulting in a dispersion of information about the sensor readings of agents throughout the whole swarm.

Evaluation. Agents evaluate the stored messages with respect to the direction from which they were received, respectively. Agents determine the diversity of the content of all messages associated with a certain direction. Depending on the systems characteristics this is practically done, for example, by calculating the variance of
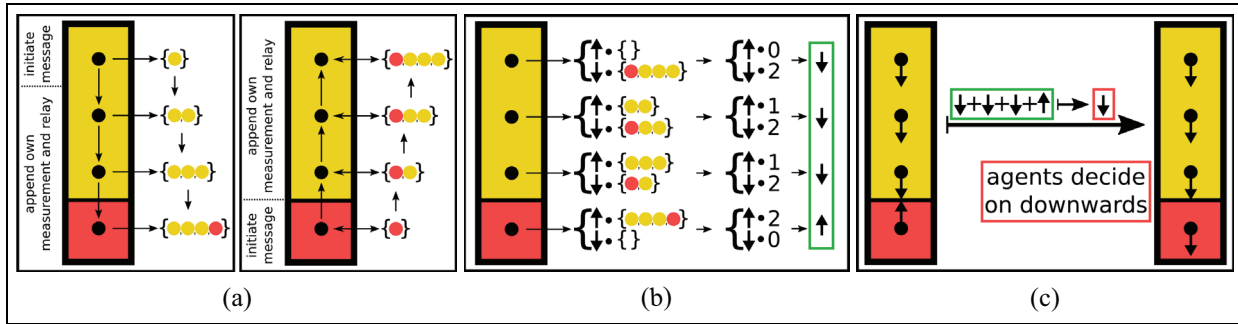
Figure 3. The three sub-parts of the algorithm can be divided. Four agents illustrated as black dots constitute a swarm in a system with two domains, yellow and red. The colors represent two different measurements of quantity X. (a) The dispersion of two independent messages. On the left hand side, the top agent (in the yellow domain) initiates a message with its own measurement, illustrated as a yellow dot in curly brackets next to the agent. The message propagates from agent to agent, each of which appends its own measurement (here depicted as color). On the right hand side, the same scenario is shown only this time the bottom agent, in the red domain, initiates the message which then propagates upward. (b) The evaluation of the two messages. The diversity of a message is illustrated as the number of different measured colors. The top agent received no messages from upward direction and thus considers a weight of $w = 0$ colors for upward, however a weight of $w = 2$ colors for downward. Its preferred direction therefore is down which is indicated by the arrow in the green box on the right hand side. All agents calculate their preferred directions in this way. (c) All agents communicate their preferred direction (not explicitly shown) and ultimately agree on a direction to move. Since three agents prefer to move downward and one agent prefers upward, the resulting common direction is downward. (a) Information gathering, (b) evaluation, and (c) collective decision.
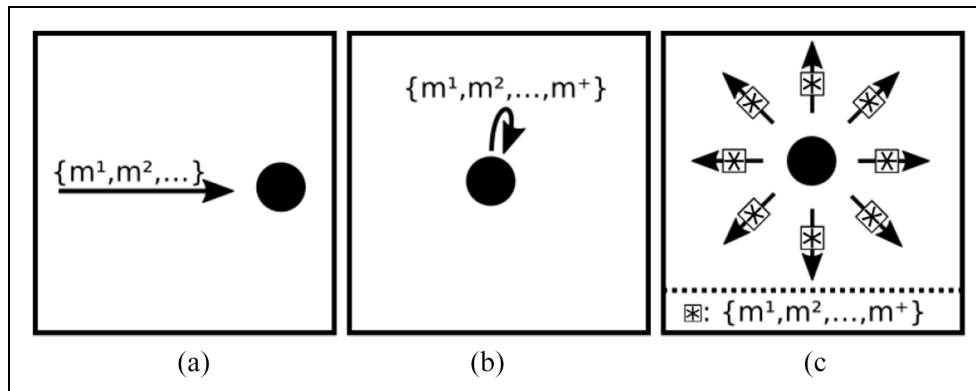

Figure 4. (a) An agent (black circle) receives an incoming message with measurements m^i. (b) It then appends its own measurement m^+ and stores the message before (c) it broadcasts this extended message.

all elements contained by those messages. The calculated diversities serve as "weights" for each respective direction. Agents finally consider the direction associated with the largest weight (e.g. variance) as their preferred, that is, the direction toward where they expect the largest diversity in information. Figure 3(b) shows the evaluation of the two messages initiated in Figure 3(a).

Collective decision. Agents agree on a common direction which yields the largest diversity in information on swarm level, based on the individual preferences of directions. One option is to let agents communicate their opinions on a preferred direction to their neighbors. Instead of relaying a message, the neighbors then simply change their own preferred direction by a small factor ρ toward the received direction. This way

opinions "diffuse" through the swarm letting it converge to a common opinion. Details on the dynamics of opinions in artificial swarms can be found in Trianni and Campo (2015) or on a more theoretical level in Hegselmann and Krause (2002) and Pluchino et al. (2006). Figure 3(c) shows the result of a collective decision on the example shown in Figures 3(a) and (b).

Algorithms 1, 2, and 3 show the pseudo-code for the three parts "information gathering," "evaluation," and "collective decision," respectively. Please note that the presented pseudo-code is just an exemplary implementation of the algorithm and that there are various alternative ways of implementing it. Namely, all three parts can be condensed into a single procedure where messages contain both measurements and preferred

Algorithm 1: Pseudo-code of "information gathering"

Mode ←"information gathering";
state ←*inactive*;
timer(t_p) ← random number $\in (0, t_p^{max}]$;
while *Mode = "information gathering"* **do**
  update timer (t_p);
  if *agent in refractory state* **then**
    wait for refractory_time;
    if *refractory_time is over* **then**
      state ← inactive;
      timer (t_p)← random number $\in (0, t_p^{max}]$;
    end
  end
  if *agent in active state* **then**
    broadcast message;
    state ← refractory;
  end
  if *agent in inactive state* **then**
    listen for incoming pings;
    if *message received* **then**
      state ← active;
      append own measurements to received message *i*;
      calculate variance V_i in measurements
      contained by message *i*;
      store variance V_i with respect to the direction of
      reception of the message,
      V_i^{dir};
    end
  end
  if *agent in inactive state and* timer(t_p) $\leqslant 0$ **then**
    state ← active;
    create empty message and append own measurements
    to message;
  end
end

Algorithm 2: Pseudo-code for "evaluation"

Mode ←"evaluation";
state ←*inactive*;
while *Mode = "evaluation"* **do**
  calculate average of variances $\overline{V^{dir}}$ of for each direction of
  reception *dir*;
  preferred direction $dir_{preferred}$← direction associated with
  largest average variance $\overline{V^{dir\_preferred}} = max\{\overline{V^{dir}}\}$;
  empty storage;
end

Algorithm 3: Pseudo-code of "collective decision"

Mode ←"collective decision";
state ←*inactive*;
timer(t_p) ← random number $\in (0, t_p^{max}]$;
while *Mode = "collective decision"* **do**
  update timer (t_p);
  if *agent in refractory state* **then**
    wait for refractory_time;
    if *refractory_time is over* **then**
      state ← inactive;
      timer (t_p)← random integer $\in (0, t_p^{max}]$;
    end
  end
  if *agent in active state* **then**
    broadcast preferred direction $dir_{preferred}$;
    state ← refractory;
  end
  if *agent in inactive state* **then**
    listen for incoming pings;
    if *message received* **then**
      state ← active;
      adjust own preferred direction by factor ρ toward
      preferred direction contained by the incoming message;
    end
  end
  if *timer(t_p) $\leqslant$ 0* **then**
    state ← active;
  end
end

directions of agents; however, for easier understanding we chose to present the algorithm separated into parts.

5. Simulation

In this section, we first present the behavior of a swarm in systems consisting of a discrete and a smooth linear transition, respectively, to give an intuitive understanding of its behavioral dynamics. We then examine a computational scenario closer to a real application case. All simulations were conducted and visualized using Python. We consider a swarm of agents within a two-dimensional space. Each agent has a perception range R, within which it detects signals sent by neighboring agents. Agents in the swarm are distributed in a circular area of diameter $D = 6R$. Based on an analysis of communication in the swarm, we present in section 5.4. and Figure 12, we chose the number of agents in the swarm N relative to D in a manner such that agents on average have five neighbors within perception range in order to ensure sufficient connectivity within the swarm. Taking $D = 6R$ results in a swarm of $N = 61$ agents.

We illustrate the functionality of the algorithm for different positions of the swarm within an environment. Every time the swarm comes to a collective decision, we manually relocate the swarm along the direction the swarm decided on and effectively show a potential trajectory of the swarm. We relocate the swarm as a whole by a distance of $s = 0.33R$ along the direction the swarm decided on, without changing the agents' relative positions. Accordingly, we exclude any interaction between agents other than communication and treat agents as point particles. We refer to the the process of the swarm coming to a collective decision as negotiation period $\mathcal{P}$. One negotiation period consists of $\mathcal{P} = 2t_p^{max}$, where $t_p^{max} = 500$ s.

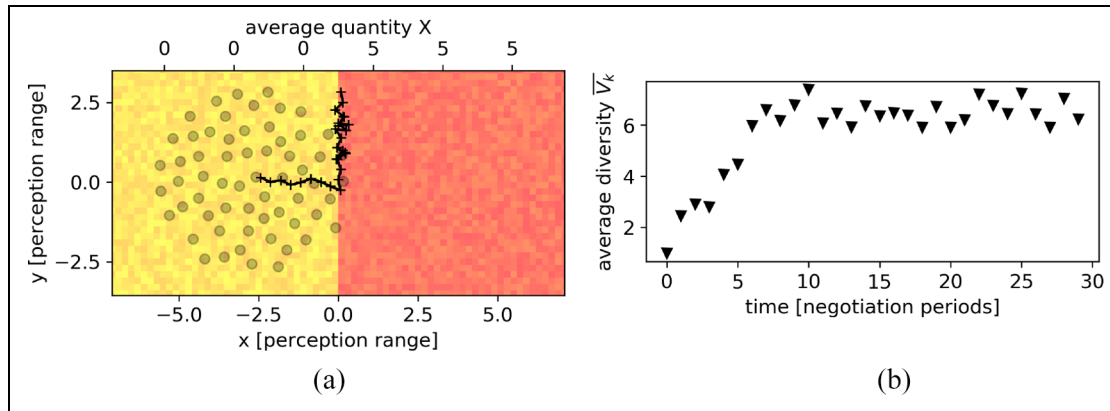In the following we quantify diversity using the messages stored by agents. We define diversity V_k^{dir}

Figure 5. (a) The trajectory of the average position of the swarm (+) within a system with two domains, yellow with and red. In light gray circles, all $N = 61$ agents in the swarm are shown at their initial positions, each circle representing one agent. (b) The diversity averaged over all agents in the swarm $\overline{V_k}$ against time t from an observer's perspective.

associated with an agent k and a direction of reception dir as the variance of the measurements m_j^k ($m_j^k \in \mathbb{R}$) contained by all stored messages of this agent associated with the respective direction of reception

$$V_k^{dir} = \frac{1}{n} \sum_{j=1}^{n} (m_j^k - \overline{m^k})^2 \qquad (1)$$

where n represents the total number of measurements m_j^k and $\overline{m^k}$ represents the average of those measurements $\overline{m^k} = (1/n)\sum_{j=1}^{n} m_j^k$.

5.1. Discrete distribution of environmental factors

In Figure 5, the scenario of a swarm close to a sharp transition of a measured quantity X is shown, left hand side in yellow for low values ($X \in [-0.5, 0.5]$), right hand side in red for high values of X ($X \in [4.5, 5.5]$). Everywhere the quantity X is subject to a small time-dependent random noise $-0.5 < \xi(t) < 0.5$. The position of the swarm, represented by a black " + ," is initially at position $(x, y) = (-2.5, 0)$, in the yellow domain. In the beginning of the simulation the swarm's decision to move is straight toward the right, toward the border of the two domains. From there at $(x, y) \approx (0, 0)$ it would move upward along the border of the two domains in a less directed manner, effectively performing a one-dimensional random walk. All agents are illustrated as gray dots at their initial position (with the center of mass of the swarm at $(x, y) = (-2.5, 0)$). In the following sections, we use "the position of the swarm" as a synonym of "the position of the center of mass of the swarm."

Figure 5(b) shows the average diversity $\overline{V_k}$ within the swarm (as viewed by an external and all-knowing observer) against time. Initially, the average diversity is close to $\overline{V_k} = 0$ and increases until $t = 8\mathcal{P}$ where it reaches a plateau around $\overline{V_k} = 6$. This corresponds to
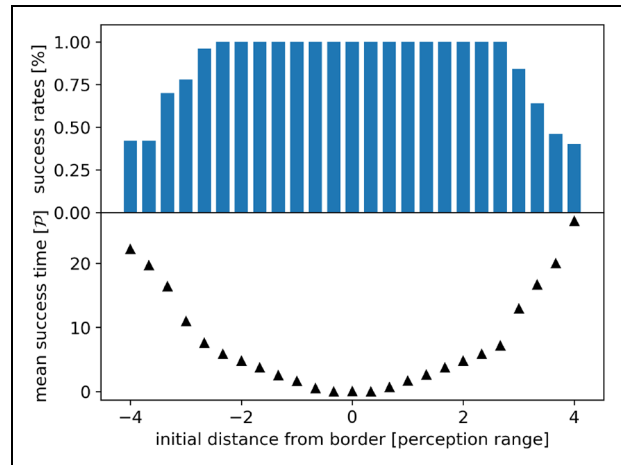

Figure 6. The top graph shows the success rates of the swarm reaching the border between the two domains versus its initial position relative to the border in our simulation experiment depicted in Figure 5. Each data point is the result of 50 independent simulations. After 50 negotiation periods, a simulation was stopped and counted as failed. The bottom graph shows the mean time for a swarm until it succeeds, only taking into account successful simulations.

the point when the swarm reached the border. Please note that the swarm is not attracted by domains of higher values of X, but instead by largest average diversity of measurements and therefore navigates toward the transition.

In Figure 6, we show the rate by which a swarm successfully reaches the border between the two domains. We count a simulation as successful if the position of the swarm reaches a distance to the border smaller than $|x| < 0.33 R$ within a finite simulation time of $t_{fin} = 50\mathcal{P}$, that is, the time in which the swarm is relocated 50 times along the direction of the collective decision. In Figure 6, the success rates (histogram in top figure) and
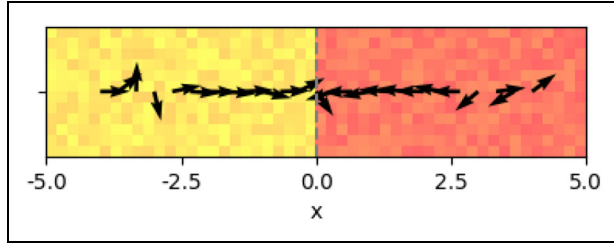
Figure 7. The preferred direction of a swarm in a system with two domains, yellow and red. Every arrow indicates the preferred direction of a swarm with its center at the arrow's position. The arrows were calculated each with a single simulation and averaged over two negotiation periods.

corresponding mean time until success (bottom figure) of a simulation are shown for different initial distances of the swarm from the border. For initial distances smaller than $|x_{init}| < 2.5R$, the success rates are 1 and the corresponding success time decreases linearly. This shows that if the swarm initially perceives the other domain (yellow and red domains as shown in Figure 5, respectively), it consistently decides on moving straight toward the other domain. For distances further away, the swarm's decision is in a random direction until by chance it perceives the respective other domain.

This consistent behavior allows us to illustrate the expected collective decision of a swarm close to the border as shown in Figure 7. It shows the preferred direction of a swarm as arrows. Each arrow represents the preferred direction of a swarm with its center at the arrow's location. For distances from the border $|d| > 2.5$, the preferred direction is random since no agent in the swarm is located in the respective other domain and thus the swarm has no information about its existence. In this case, the swarm is located in an almost uniform area and thus does not develop a

preferred direction. For distances from the border $|d| < 2.5$, the swarm's preferred direction is toward the border.

5.2. Gradient distribution of environmental factors

Figure 8 shows a swarm close to a gradient in X. For $x < 0$, the system exhibits fluctuating values in $X \in [-0.5, 0.5]$, for $x \geqslant 0$ the temporal average in X linearly increases. The swarm initially starts at position $(x, y) = (-2.5, 0)$ and decides on the direction toward the right. For $x \gtrsim 2.33$, the swarm's decision on a preferred direction is relatively random. In Figure 8(b), the diversity averaged over all agents in the swarm $\overline{V_k}$ is shown against time. It increases from $\overline{V_k} \approx 0.3$ until at $t = 19\mathcal{P}$ (when the swarm would move randomly) it reaches a plateau where it fluctuates around $\overline{V_k} = 20$.

In Figure 9, we show the rate at which a swarm successfully maximizes its average diversity $\overline{V_k}$. We count a simulation as successful if the swarm reached a position of $x \geqslant 2.33$ within a finite simulation time of $t_{fin} = 50\mathcal{P}$, that is, the time in which the swarm is relocated 50 times. In Figure 9, the success rates (top histogram) and corresponding mean time until success (bottom graph) of a simulation are shown for different initial distances of the swarm from the onset point of the linear-increase domain. For initial positions $x_{init} \geqslant 2.33$, the swarm succeeds instantly as its initial position already fulfills the condition for success. For $-2 \leqslant x_{init} < 2.33$, the swarm succeeds in the majority of conducted simulations, the success times increase with decreasing distance from the border between the two domains. At $x_{init} \leqslant -2$, for decreasing distance from the border the success rates decrease significantly at $x_{init} = -4$, and they reach 25%. For $x_{init} \leqslant -2$, the swarm is too far away from the domain of increasing values in X and therefore does not perceive it anymore. In this case, the swarm effectively
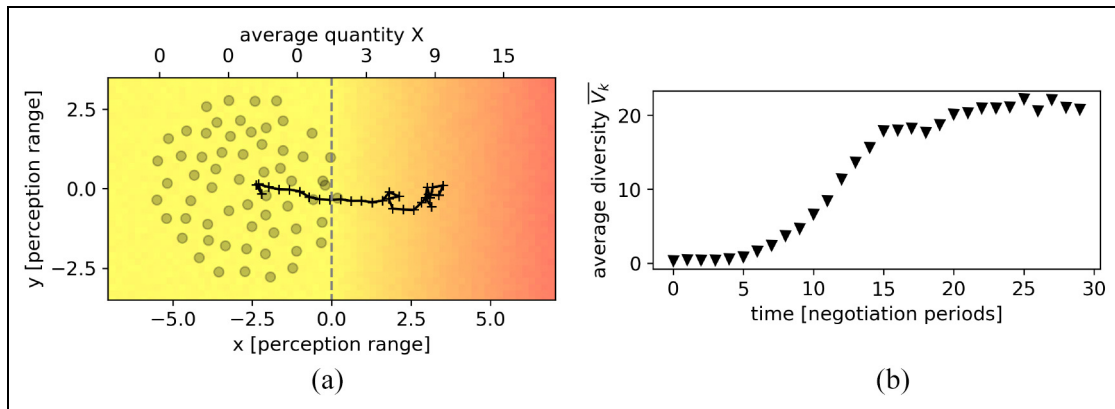

Figure 8. (a) The trajectory of the position of the swarm (+) in a system with gradient distribution of environmental factors. In light gray circles, all $N = 61$ agents in the swarm are shown at their initial positions, each circle representing one agent. (b) The diversity averaged over all agents in the swarm $\overline{V_k}$ against time t from an observer's perspective.
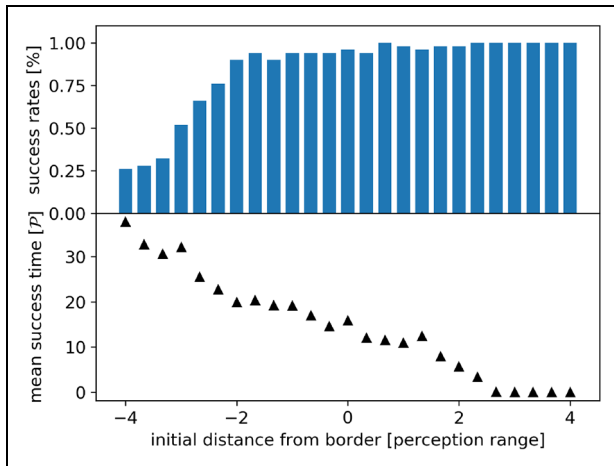
Figure 9. The top graph shows the success rates of the swarm reaching the border (as depicted in Figure 5) between the two domains versus the initial position of the swarm. For each distance, we conducted 50 independent simulations. After 50 negotiation periods, a simulation was stopped and counted as failed. The bottom graph shows the mean time for a swarm until it succeeds, only taking into account successful simulations.
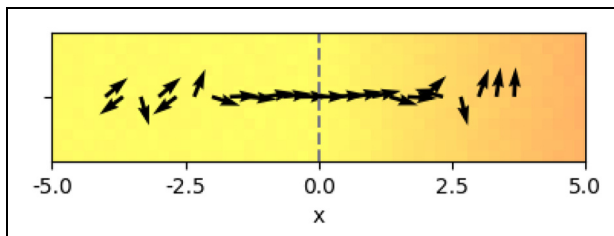
Figure 11. The preferred direction of a swarm within a system with a circular domain of levels of X deviating from its surrounding. Each arrow represents the preferred direction of a swarm with its center at an agents position. Each arrow is the result of a single negotiation period. The circular domain extends radially with a radius of $r = 5$. The levels of X linearly decrease from a maximum of 5 down to 0. The levels of X are subject to a random noise of $-0.5 \leqslant \xi(t) \leqslant 0.5$.

Figure 10. The preferred direction of a swarm in a system with (toward the right hand side) linearly increasing average levels of X. Every arrow indicates the preferred direction of a swarm with its center at the arrow's position. The arrows were calculated each with a single simulation and averaged over two negotiation periods.

5.3. Two-dimensional cloud

As a simple two-dimensional distribution of locally deviating environmental parameters, we consider an area of radially varying values of X. In Figure 11, the quantity X fluctuates around a constant value $X \in [4.5, 5.5]$ in the yellow domain. For radial distances from the center of the cloud between $d \in [3, 4.5]$, the quantity X linearly decreases to $X \in [-0.5, 0.5]$. For $d < 3$, the red domain exhibits $X \in [-0.5, 0.5]$. Figure 11 shows the preferred direction of a swarm as arrows, the position of each arrow indicating the position of the swarm. For radial distances from the center of cloud of $d \geqslant 7$, the swarm would move randomly as it does not detect the circular domain. For $7 \gtrsim d \gtrsim 4$, the swarm would move toward the circular domain, whereas for $d \leqslant 4$ it would move radially away from it toward its border. As soon as the swarm perceives the circular domain, it collectively decides on the direction toward its border where the swarm measures the largest average diversity.

5.4. Density of the swarm

A crucial parameter is the density of the swarm since connectivity among agents has to be ensured. If the number of agents N within a swarm of diameter D is too small, agents have few or even no neighbors and

performs a random walk and by chance it would get closer to the domain of increasing values in X and ultimately succeed, that is, reach $x \geqslant 2.33$ within simulation time. Only successful simulations were taken into account when calculating the mean success times.

As soon as the swarm is entirely on a linear gradient (for $x > 2.3$), that is, the swarm does not perceive the domain with constant values in X, both directions (to lower and to higher values, respectively) provide the same average diversity. This is implicitly shown in Figure 10 where each arrow denotes the preferred direction of a swarm with center at its position. For $x < -2.2$, the swarm's collective decision on where to move is random since it does not perceive the linear gradient (starting at the dashed line). For $x > -2.2$, the swarm would move toward the right up to $x = 2.2$, where it would move randomly.
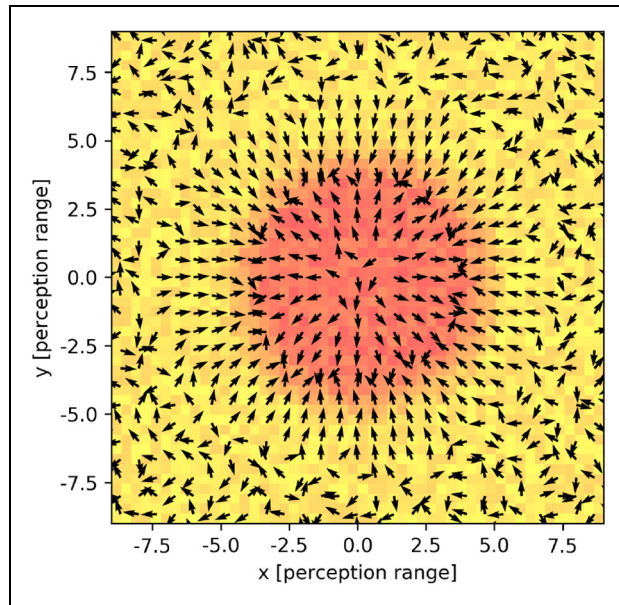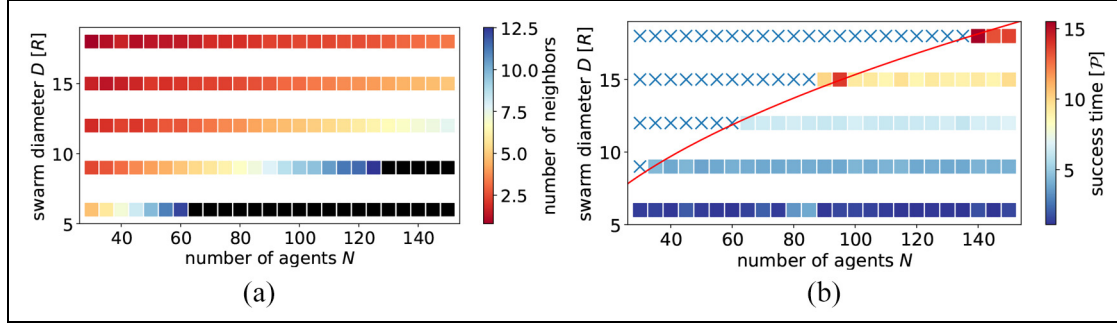
Figure 12. (a) The average number of neighbors of an agent with respect to the total number of agents *N* in the swarm and the diameters *D* of the swarm is shown. The black squares in the bottom right of the figure represent average numbers of neighbors larger than 12.5. (b) The time (in unit of negotiation periods) until a swarm in a set of simulations is counted as successful versus the number of agents *N* in the swarm and the diameter *D* of the swarm. For every set of parameters, we ran 10 independent simulations. In case the swarm did not reach the boundary between the two domains within 60 negotiation periods, we stopped the simulation. The blue crosses mark the sets of parameters for which one or more simulation runs did not succeed. The red line is a fit along the border between successful and unsuccessful simulations.

communication throughout the swarm is not necessarily provided. In Figure 12(a), we show the average number of neighbors versus number of agents in the swarm and the swarm's diameter. The black squares in the bottom right of Figure 12(a) represent average numbers of neighbors larger than 12.5. We conducted simulations testing functionality of the swarm for each configuration shown in Figure 12(b). We placed a swarm within a system consisting of two uniform domains, equivalent to the system presented in Figure 5(a). The swarm was placed in a distance from the border between the two domains of $d = 0.4D$, such that the swarm could perceive the other domain. We conducted 10 independent simulations per set of parameters. We counted the simulation as successful when the center of the swarm reached a distance $d \leqslant 1R$ to the border between the two domains. If a swarm did not reach the border within 60 negotiation periods, we stopped the simulation.

In Figure 12(b), we show the time (or number of negotiation periods) it took a swarm on average to reach the border between the two domains. If any of the 10 simulations per set of parameters were not counted as success, instead of the success time we plotted a cross. This is the case for most simulation in the top left of Figure 12(b), whereas all other sets of simulations were successful. The red line is a fit of shape $D(N) \sim \sqrt{N}$ along the border between the successful and unsuccessful simulations. The fit corresponds to the line along which each agent has approximately 2.9 neighbors. Hence for the presented setup of the swarm, a minimum of approximately three neighbors per agent ensures consistent functionality of the algorithm. For the simulations presented in the previous sections, the diameter is set to $D = 6R$ and the total number of agents to $N = 61$ which corresponds to an average of five neighbors.

Given a sufficient connectivity within a swarm, the density does not influence functionality of the algorithm. The success times are relatively constant for each diameter of the swarm, respectively. With larger swarm diameter *D*, the success times increase, since for a larger diameter, the distance of the center of the swarm to the border is larger. Hence, it took more negotiation periods until the swarm reached the border and finished a simulation successfully.

6. Robotic experiments

In the following, we demonstrate the minimum feasibility of the presented algorithm on real robots. It is to be noted that while those minimal working experiments do not provide deeper inside into the performance or functionality of the algorithm, they serve as a proof of concept for the communication and decision-making process of the algorithm. The experiments therefore do not represent a fully functional swarm under real conditions.

For the robotic experiments, we used the aMussel robots, developed in the project subCULTron (Donati et al., 2017). They communicate via modulated light and are used, amongst others, for examining the anoxic waters phenomenon in the lagoon of Venice (Runca et al., 1996) by diving down to the floor of the lagoon. They are equipped with a variety of sensors and communication devices (Donati et al., 2017), including sensors for oxygen levels as well as ambient light. aMussels can only dive up to the water surface and down to the floor of a water body and have no other means of transportation of their own. In the field they are transported by a different type of robot, so-called "aPads," which constitute a part of the heterogeneous robotic swarm within project subCULTron (Thenius et al., 2016). Accordingly, in the following experimental setup, aMussels can rather be viewed as stationary sensors.
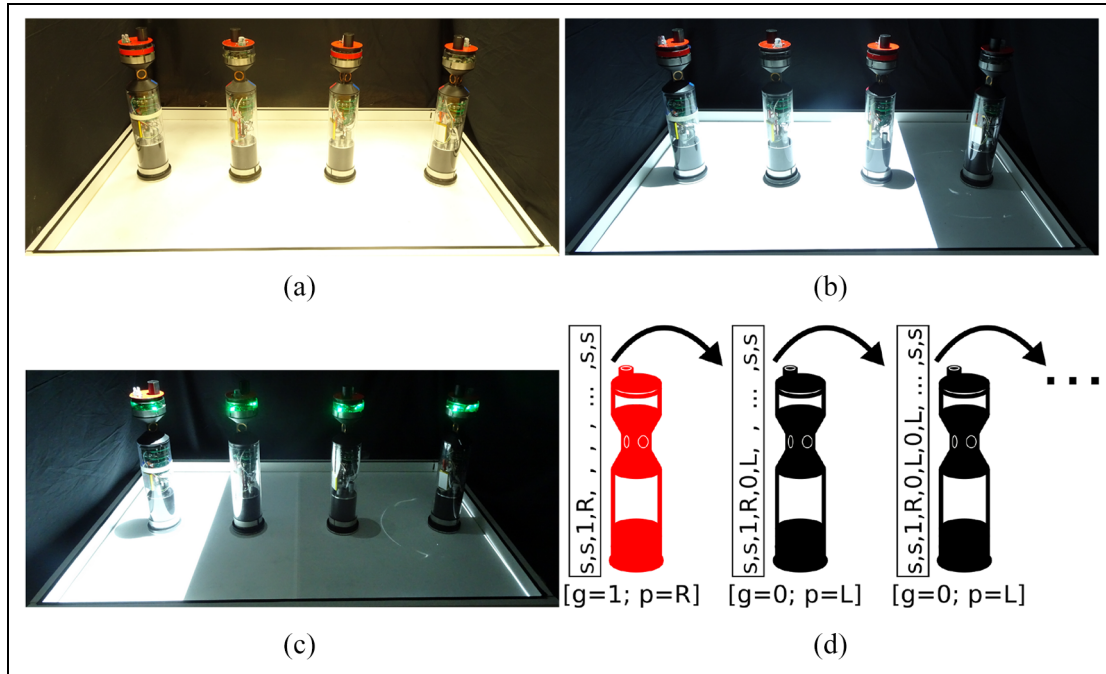
Figure 13. (a) Four aMussels in an arena used for the experiments. (b) The experimental setup with one of the six considered light configurations. The aMussels did not decide on a preferred direction yet as their top caps are not illuminated. (c) An experiment counted as successful with all aMussels agreeing on the common preferred direction toward the illuminated domain, indicated by the green LEDs in their top caps. (d) Schematic illustration of broadcasting and relaying messages.

6.1. Experimental setup

For testing the algorithm, we used aMussels under lab conditions outside water in an one-dimensional setup. Four aMussels were arranged in a linear manner in an arena as shown in Figure 13(a). As an emulation of oxygen gradients, we used an ambient light gradient which allowed us to perform the experiments in the lab outside of a water environment. We hence were able to establish precisely controlled environmental conditions and predictably changing environments. Two projectors were located above the arena and used for varying the light intensity on the arena floor as shown in Figure 13(b) and (c) where different parts of the floor are brightly illuminated and others dark. In this experiments, we considered two states of illumination: lights on or off. The setup of the system corresponds to the simulation of a swarm close to a sharp transition, presented in section 5.1.

The sensor for measuring ambient light values is located at the top cap of the aMussels. In this experiment, they communicated via modulated green light. We counted an experiment as successful as soon as the robots agreed on the direction toward the border between the two different domains. While lights for communication are located in the center of their body, the LED's in their top caps (as visible in Figure 13(c) in green) indicated their preferred direction. From the perspective of the camera, green represents the preferred direction "left" and blue represents "right."

The algorithm introduced in section 4.2. was implemented on the aMussels; however, the information gathering and negotiation phases were fused into a single phase. All messages sent by aMussels in this experiment contained both the sensor readings and their preferred direction. This is illustrated in Figure 13(d), where the red aMussel initiates the sending of a message where it broadcast its own sensor reading (local ambient light measurement value $g \in \{0, 1\}$) as well as its current preferred direction ($p \in \{R, L\}$). Other aMussels which successively receive the message (shown on the right side in black color) add their own sensor readings and preferred directions to the message before relaying it, respectively. The messages broadcast by the aMussels are shown to the left of each aMussel. Messages started and ended with the characters "ss" for ease of parsing. Based on the sensor readings in the stored messages, the aMussels continuously evaluated from which direction they received messages with largest variance in measurements, that is, which direction they individually considered most preferable to move toward and likewise which direction they broadcast as their own current preferred direction. Based on the preferred directions in the stored messages, at the end of this phase (consisting of both the information gathering phase and the negotiation phase), the aMussels evaluated which direction was favored by the majority of the swarm and thus which direction they ultimately decided on moving toward. This phase consisted of a time
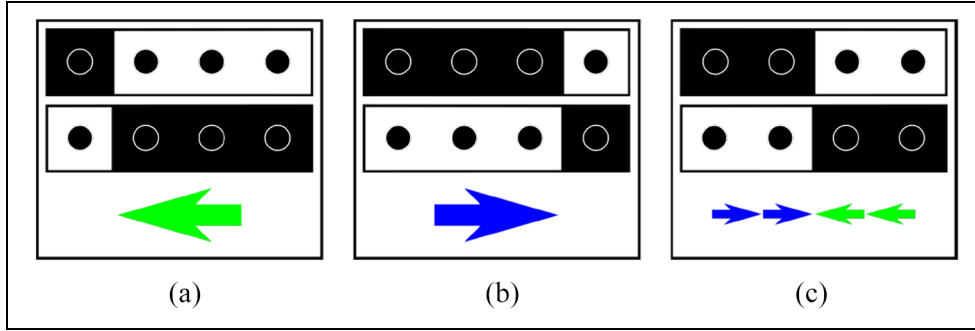
Figure 14. Six different light configurations which were tested in the experiments. (a, b) The configurations for which we expect aMussels to decide to move to the left and to the right, respectively. (c) The configurations for which we expect the aMussels to not agree on a common direction but to choose directions toward the border of the domains of different luminosity. The two aMussels left from the border decide to move to the right and vice versa.

Table 1. The parameter values used in the experiments.

Parameter	Value
Cycle length	55 ± 15 s
Refractory time	4 s
Message length	32 bytes
Negotiation period	10 cycle lengths
Length of messages	32 bytes

period of 10 cycles, meaning every aMussel initiated at least 10 messages. The parameter values used in the experiment are given in Table 1. The robots randomized the time when they initiated a message during a cycle at the beginning of every cycle. As a result, in this experiment the effective cycle length of individual robots varied between 40 and 70 s as indicated in Table 1.

6.2. Experimental results

We conducted experiments for six different light configurations as schematically shown in Figure 14. The arrows shown below each configuration indicate the results of each set of experiments. The direction of the arrow indicates the collective decision of the swarm in which direction to move and its color denotes the color used by the aMussels to indicate the respective direction they decided on (e.g. see Figure 13(c)). For Figure 14(c), we counted an experiment as successful if the aMussels ultimately decided to move toward the border between the two domains of luminosity, that is, the two aMussels on the left choose to move toward the right and vice versa.

For each light configuration, the experiment was independently repeated five times with a success rate of 100%. In order to test how well the aMussels adapt to changing light configurations, we conducted another set of experiments with alternating light configuration in which the robots need to change their previously

reached consensus. After reaching consensus, the light configurations were changed such that the expected direction for the robots to decide on was inverted. The experiments were considered as successful if the robots correctly found consensus for the initial light configuration and then switched their opinion accordingly. We conducted this experiment five independent times with all experiments successful.

7. Discussion

In this article, we demonstrated how a simple bioinspired communication behavior can be used to reach a swarm-level decision of which direction to collectively relocate in order to maximize swarm-level information access. We also demonstrated how this algorithm works in robots of an underwater swarm with limited communication range and local information.

7.1. Simulations

We presented simulation results in section 5 to give an intuitive understanding of the algorithm's functionality. For both a spatially discrete and a gradual change in measured quantity X in the system, the swarm could successfully maximize its diversity in measurements. For systems with a discrete change in X (section 5.1.), the swarm, within proximity of nearby variations, succeeded in 100% of all simulations, whereas for systems with a gradual change in X (section 5.2.), the success rates vary between 85% and 100%. Also, the mean success times in the latter system (Figure 9) are significantly larger compared to the prior (Figure 6). It shows that the performance of artificial swarms using this algorithm increases with increasingly steep gradients in measured quantity in the system. Furthermore, as could be observed in Figure 8 for $x > 2.5$, for a constant gradient a swarm does not have a preferred direction to move. Since for both directions along the gradient, the measurements result in the same variance, a swarm's

decision on a preferred direction would effectively be random.

7.2. Experiments

In section 6, we presented experimental results of a simplified laboratory demonstration of the algorithm implemented on robots using $N = 4$ aMussels in a one-dimensional setup. The resulting behavior is in full qualitative agreement with the results of the corresponding numerical simulations (Figure 7) in all of the experiments shown in Figure 14(a) and (b).

While in the simulations messages were sent and received instantly, in experiments it is possible that several robots send a message quasi-simultaneously, thus potentially creating a corrupted message. However, in the unlikely event of robots perceiving such a message and accepting it as valid, this message quickly loses significance due to averaging over all incoming messages.

While in the simulations we consistently let the swarm come to a swarm-wide coherent decision where to move, in the experiments we considered an additional case. The decision result shown in Figure 14(c), where two aMussels decide to move left and two aMussels decide to move right, occurred for symmetric configurations of agents within the setup due to the way we implemented the algorithm on the aMussels. In the implementation of the algorithm on the aMussels, we intentionally left space for such a decision results due to two reasons: (1) it was more energy efficient, robust, and straightforward for the demonstration of CIMAX on the aMussels and (2) during field experiments aMussels are transported by a different species of robot, the aPads (Thenius et al., 2016), which will serve as an additional level to control for the case of such decision results. However, for increasing numbers of robots in the swarm, which are distributed in two dimensions, the probability of reaching such decision results becomes vanishingly small. Nonetheless, such cases need to be accounted for when implementing the algorithm on robotic swarms with respect to their specific abilities and target environments. Despite the implementation leaving space for the discussed decision results and the simplification of restricting the experiments to one dimension, they serve as a first proof of concept of the algorithm.

7.3. Swarm configuration and sizes of messages

It is also worth pointing out that the configuration of a swarm, for example, the spatial distribution of agents has a significant influence on the preferred direction. Let us consider a system of entirely random values in X. If agents are distributed in a line, the preferred direction can only be along the linear distribution of the swarm as messages are only shared along this line. This needs to be taken into account in case a swarm

tends to group or shape up in symmetric ways in order to avoid systematic errors. However, shapes of the swarm such as circles, squares, and triangles do not significantly influence functionality of the presented algorithm given that the swarm is connected. The influence of the shape of a swarm on functionality of algorithms was examined in Varughese et al. (2020). In the case of a circularly shaped swarm, as used in this work, the maximum size of a message l grows with the diameter of the swarm R_s: $\max(l) \sim R_s/R$. If the number of members of a swarm is disproportionately large and therefore the sizes of messages grow exceedingly large, one could think of a different protocol ensuring an upper limit for the lengths of messages. If, for example, the measured quantity X is within an expected spectrum of values, one could, instead of measurements added to a message, send a histogram of measured values. Each agent would then simply increment the bin of the histogram which corresponds to the agent's measurement.

7.4. Connectivity

For both simulations and experiments shown in this work, we assumed/ensured an interconnected swarm with every agent being connected to at least one neighbor at all times—in the presented simulations on average each agent was connected to five neighboring agents. This assumption allowed the demonstration of the conceptual functionality of the algorithm; however, it is not a feasible assumption in a real-world scenario. Although it could be shown in previous work how the presented underlying communication mechanism exhibits significant resilience to signal loss (Varughese et al., 2017), in practice a number of steps need to be taken in order to ensure connectivity of robots. In a real-world scenario, one has to account for possible occlusions, alignment problems, and other prospective challenges.

7.5. Simultaneous signaling and communication protocols

If two or more neighboring agents initiate sending of their preferred direction simultaneously, they do not receive any messages since they are insensitive during and briefly after sending. This algorithm is based on the statistical evaluation of large numbers of collected messages, wherefore this event generally does not affect the functionality of the algorithm.

In case t_p^{max} is chosen very small, simultaneous signaling can occur more frequently which would lead to higher probability of ignored messages. This in turn would result in an increasing amount of time necessary for the swarm to reach a collective decision coherent with the environmental conditions—or even result in failure of the algorithm. Generally, in order to handle conflicts in signaling, t_p^{max} has to be chosen appropriately by the

user. In order to concentrate on the core of the CIMAX algorithm, we kept the randomization of time slots simple by using an initial randomization. However, more sophisticated de-sychronization techniques (Maistrenko et al., 2004) can be employed when de-sychronization is crucial.

A more detailed analysis of such events within the type of presented swarm was carried out by Varughese et al. (2020). However, there are numerous approaches of implementing communication protocols compatible to the CIMAX algorithm. The presented communication mechanism is related to the family of gossip protocols, which comprises a large variety of different procedures and implementations for distributing information within a swarm or network (Birman, 2007; Shah et al., 2009).

7.6. Topology of target environment

In this work, we assumed an open environment and did not account for obstacles or physical hazards. This is a plausible assumption for parts of the lagoon of Venice, i.e., for the target environment of the robotic swarm this algorithm was originally designed for. However, for other types of swarms and in other types of environments, for instance environments containing domains which could lead to instant loss or failure of robots, one has to extend the algorithm to prevent the swarm from gradually losing members. One has to account for scenarios such as, when the majority of a swarm finds it favorable to move in a certain direction which, if moved along it, would lead to individual agents entering a hazardous domain and subsequently fail.

7.7. Measure for information

For evaluating incoming messages between robots, we used the variance of the received measurements. Although variance is a simple measure, it is effective with respect to functionality of the algorithm for a swarm measuring a single environmental parameter. Due to the statistical nature of variance, random noise in measurements does not affect the functionality of the algorithm, given that the amount of measurements used for calculating the variance needs to be sufficiently large to compensate the effects of random noise. In contrast to our approach, Cui et al. (2004) use a fuzzy logic–based evaluation. Such complex measures could be used in place of variance in the CIMAX algorithm when dealing with complex parameter spaces while following the same information gathering, evaluation, and collective decision phases. Furthermore, in this work, we only considered a single quantity being measured; however, this algorithm constitutes a general approach for collective decision-making in this particular class of swarms or networks. Multiple quantities can be considered, resulting in a swarm maximizing the data points within a phase space spanned by the number of considered quantities. This hence would enable such a swarm to autonomously explore an environment of high complexity in environmental parameters, taking into account previously collected data and adjusting to environmental changes and variations.

As the algorithm could be proven conceptually functional with respect to collective decision-making, it will be further implemented and tested in the future on larger swarms and in-field within the framework of subCULTron.

ORCID iD

Hannes Hornischer https://orcid.org/0000-0002-0556-4716

References

Ahl, V., & Allen, T. F. (1996). *Hierarchy theory: A vision, vocabulary, and epistemology*. Columbia University Press.

Akyildiz, I. F., Pompili, D., & Melodia, T. (2005). Underwater acoustic sensor networks: Research challenges. *Ad Hoc Networks*, 3(3), 257–279.

Beckers, R., Deneubourg, J. L., Goss, S., & Pasteels, J. M. (1990). Collective decision making through food recruitment. *Insectes Sociaux*, 37(3), 258–267.

Beni, G., & Wang, J. (1989). Swarm intelligence in cellular robotic systems. In *Proceedings of the NATO Advanced Workshop on Robots and Biological Systems* (Vol. 3, pp. 268–308). Springer.

Birman, K. (2007). The promise, and limitations, of gossip protocols. *ACM SIGOPS Operating Systems Review*, 41(5), 8–13.

Bonner, J. T. (1949). The social amoebae. *Scientific American*, 180(6), 44–47.

Brock, V. E., & Riffenburgh, R. H. (1960). Fish schooling: A possible factor in reducing predation. *ICES Journal of Marine Science*, 25(3), 307–317.

Buck, J. (1988). Synchronous rhythmic flashing of fireflies. ii. *The Quarterly Review of Biology*, 63(3), 265–289.

Cavagna, A., Cimarelli, A., Giardina, I., Parisi, G., Santagati, R., Stefanini, F., & Viale, M. (2010). Scale-free correlations in starling flocks. *Proceedings of the National Academy of Sciences*, 107(26), 11865–11870.

Codling, E., Pitchford, J., & Simpson, S. (2007). Group navigation and the "many-wrongs principle" in models of animal movement. *Ecology*, 88(7), 1864–1870.

Conradt, L., & Roper, T. J. (2005). Consensus decision making in animals. *Trends in Ecology & Evolution*, 20(8), 449–456.

Cui, X., Hardin, C. T., Ragade, R. K., & Elmaghraby, A. S. (2004) A swarm-based fuzzy logic control mobile sensor network for hazardous contaminants localization. In *2004 IEEE International Conference on Mobile Ad-hoc and Sensor Systems* (IEEE Cat. No.04EX975, pp. 194–203). IEEE.

Dias, M. B., Zlot, R., Kalra, N., & Stentz, A. (2006). Market-based multirobot coordination: A survey and analysis. *Proceedings of the IEEE*, 94(7), 1257–1270.

Donati, E., van Vuuren, G. J., Tanaka, K., Romano, D., Schmickl, T., & Stefanini, C. (2017). aMussels: Diving and anchoring in a new bio-inspired under-actuated robot class for long-term environmental exploration and monitoring. In Y. Gao, S. Fallah, Y. Jin, & C. Lekakou (Eds.), *Conference Towards Autonomous Robotic Systems* (pp. 300–314). Springer.

Dorigo, M., Birattari, M., & Stutzle, T. (2006). Ant colony optimization. *IEEE Computational Intelligence Magazine*, 1(4), 28–39.

Dorigo, M., Trianni, V., Şahin, E., Groß, R., Labella, T. H., Baldassarre, G., Nolfi, S., Deneubourg, J. L., Mondada, F., Floreano, D., & Gambardella, L. M. (2004). Evolving self-organizing behaviors for a swarm-bot. *Autonomous Robots*, 17(2), 223–245.

Durston, A. (1973). Dictyostelium discoideum aggregation fields as excitable media. *Journal of Theoretical Biology*, 42(3), 483–504.

Eberhart, R. C., Shi, Y., & Kennedy, J. (2001). *Swarm intelligence*. Elsevier.

Erignac, C. (2007, May 7–10). *An exhaustive swarming search strategy based on distributed pheromone maps* [conference session]. AIAA Infotech@ Aerospace 2007 Conference and Exhibit, Rohnert Park, CA, United States.

Fernandez-Marquez, J. L., Serugendo, G. D. M., Montagna, S., Viroli, M., & Arcos, J. L. (2013). Description and composition of bio-inspired design patterns: A complete overview. *Natural Computing*, 12(1), 43–67.

Garnier, S., Gautrais, J., Asadpour, M., Jost, C., & 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., Gautrais, J., & Theraulaz, G. (2007). The biological principles of swarm intelligence. *Swarm Intelligence*, 1(1), 3–31.

Hegselmann, R., & Krause, U. (2002). Opinion dynamics and bounded confidence: Models, analysis and simulation. *Journal of Artificial Societies and Social Simulation*, 5(3). http://jasss.soc.surrey.ac.uk/5/3/2/2.pdf

Kernbach, S., Häbe, D., Kernbach, O., Thenius, R., Radspieler, G., Kimura, T., & Schmickl, T. (2013). Adaptive collective decision-making in limited robot swarms without communication. *The International Journal of Robotics Research*, 32(1), 35–55.

Kernbach, S., Thenius, R., Kernbach, O., & Schmickl, T. (2009). Re-embodiment of honeybee aggregation behavior in an artificial micro-robotic system. *Adaptive Behavior*, 17(3), 237–259.

Kim, L. H., & Follmer, S. (2017). UbiSwarm: Ubiquitous robotic interfaces and investigation of abstract motion as a display. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(3), 66.

Koenig, S., Keskinocak, P., & Tovey, C. (2010). Progress on agent coordination with cooperative auctions. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence* (pp. 1713–1717). Association for Computing Machinery.

Lanbo, L., Shengli, Z., & Jun-Hong, C. (2008). Prospects and problems of wireless communication for underwater sensor networks. *Wireless Communications and Mobile Computing*, 8(8), 977–994.

Li, X., Santoro, N., & Stojmenovic, I. (2007). Mesh-based sensor relocation for coverage maintenance in mobile sensor networks. In *International Conference on Ubiquitous Intelligence and Computing* (pp. 696–708). Springer.

Magurran, A. E., & Pitcher, T. J. (1987). Provenance, shoal size and the sociobiology of predator-evasion behaviour in minnow shoals. *Proceedings of the Royal Society of London B*, 229(1257), 439–465.

Maistrenko, Y., Popovych, O., Burylko, O., & Tass, P. A. (2004). Mechanism of desynchronization in the finite-dimensional kuramoto model. *Physical Review Letters*, 93(8), 084102.

Marthaler, D., & Bertozzi, A. L. (2003). Collective motion algorithms for determining environmental boundaries. In *SIAM Conference on Applications of Dynamical Systems*.

Otte, M. (2018). An emergent group mind across a swarm of robots: Collective cognition and distributed sensing via a shared wireless neural network. *The International Journal of Robotics Research*, 37(9), 1017–1061.

Pluchino, A., Latora, V., & Rapisarda, A. (2006). Compromise and synchronization in opinion dynamics. *The European Physical Journal B-Condensed Matter and Complex Systems*, 50(1–2), 169–176.

Rabb, G. B., Woolpy, J. H., & Ginsburg, B. E. (1967). Social relationships in a group of captive wolves. *American Zoologist*, 7(2), 305–311.

Renfrew, D., & Yu, X. H. (2009). Traffic signal control with swarm intelligence. In *Proceedings of the Fifth International Conference on Natural Computation* (pp. 79–83). IEEE.

Runca, E., Bernstein, A., Postma, L., & Silvio, G. D. (1996). Control of macroalgae blooms in the Lagoon of Venice. *Ocean & Coastal Management*, 30(2), 235–257.

Schwager, M., Dames, P., Rus, D., & Kumar, V. (2017). A multi-robot control policy for information gathering in the presence of unknown hazards. In H. Christensen, & O. Khatib (Eds.), *Robotics research* (pp. 455–472). Springer.

Seeley, T. D. (1992). The tremble dance of the honey bee: Message and meanings. *Behavioral Ecology and Sociobiology*, 31(6), 375–383.

Shah, D. (2009). Gossip algorithms. *Foundations and Trends® in Networking*, 3(1), 1–125.

Simons, A. M. (2004). Many wrongs: The advantage of group navigation. *Trends in Ecology & Evolution*, 19(9), 453–455.

Smith, R. N., Cooksey, P., Py, F., Sukhatme, G. S., & Rajan, K. (2016). Adaptive path planning for tracking ocean fronts with an autonomous underwater vehicle. In M. Hsieh, O. Khatib, & V. Kumar (Eds.), *Experimental robotics* (pp. 761–775). Springer.

Stojanovic, M., & Preisig, J. (2009). Underwater acoustic communication channels: Propagation models and statistical characterization. *IEEE Communications Magazine*, *47*(1), 84–89.

Strobel, V., Castelló Ferrer, E., & Dorigo, M. (2018). Managing byzantine robots via blockchain technology in a swarm robotics collective decision making scenario. In *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems* (pp. 541–549). Association for Computing Machinery.

subCULTron. (2015). *Submarine cultures perform long-term robotic exploration of unconventional environmental niches.* http://www.subcultron.eu/

Szopek, M., Schmickl, T., Thenius, R., Radspieler, G., & Crailsheim, K. (2013). Dynamics of collective decision making of honeybees in complex temperature fields. *PLOS ONE*, *8*(10), Article e76250.

Thenius, R., Moser, D., Varughese, J. C., Kernbach, S., Kuksin, I., Kernbach, O., Kuksina, E., Mišković, N., Bogdan, S., Petrović, T., Babić, A., Boyer, F., Lebastard, V., Bazeille, S., Ferrari, G. W., Donati, E., Pelliccia, R., Romano, D., Van Vuuren, J., & . . . Schmickl, T. (2016). subCULTron—Cultural development as a tool in underwater robotics. In *Artificial Life and Intelligent Agents Symposium* (pp. 27–41). Springer.

Thenius, R., Varughese, J. C., Moser, D., & Schmickl, T. (2018). WOSPP—A wave oriented swarm programming paradigm. *Ifac-PapersOnline*, *51*(2), 379–384.

Trianni, V., & Campo, A. (2015). *Fundamental collective behaviors in swarm robotics*. Springer.

Vail, D., & Veloso, M. (2003). Multi-robot dynamic role assignment and coordination through shared potential fields. *Multi-Robot Systems*, *2*, 87–98.

Varughese, J. C., Hornischer, H., Thenius, R., Zahadat, P., Wotawa, F., & Schmickl, T. (2020). A swarm design paradigm unifying swarm behaviors using minimalistic communication. *Bioinspiration & Biomimetics*, *15*, 1–27.

Varughese, J. C., Thenius, R., Schmickl, T., & Wotawa, F. (2017). Quantification and analysis of the resilience of two swarm intelligent algorithms. In C. Benzmüller, C. Lisetti, & M. Theobald (Eds.), *3rd Global Conference on Artificial Intelligence, EPiC Series in Computing* (*Vol. 50*, pp. 148–161). EasyChair.

Varughese, J. C., Thenius, R., Wotawa, F., & Schmickl, T. (2016). FSTaxis algorithm: Bio-inspired emergent gradient taxis. In *Proceedings of the 15th International Conference on the Synthesis and Simulation of Living Systems* (pp. 330–337). MIT Press.

Wang, G., Cao, G., La Porta, T., & Zhang, W. (2005). Sensor relocation in mobile sensor networks. In *Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies* (*Vol. 4*, pp. 2302–2312). IEEE.

Zahadat, P., & Schmickl, T. (2016). Division of labor in a swarm of autonomous underwater robots by improved partitioning social inhibition. *Adaptive Behavior*, *24*(2), 87–101.

Zarzhitsky, D., Spears, D. F., & Spears, W. M. (2005). Swarms for chemical plume tracing. In *Proceedings 2005 IEEE Swarm Intelligence Symposium* (pp. 249–256). IEEE.

About the Authors

Hannes Hornischer received a MSc in Physics at the University of Göttingen, Germany, where he worked at the Max Planck Institute for Dynamics and Self-Organization. Within framework of his PhD he worked at the Artificial Life Laboratories in Graz, Austria, in the EU-funded project subCULTron which developed and deployed an underwater robotic swarm in the lagoon of Venice, Italy. In his research he focuses on self-organization in swarms, group coordination and artificial intelligence.

Joshua Cherian Varughese received his PhD from Graz University of Technology, Austria, in November 2019. He is currently a postdoctoral researcher at the Artificial Life Lab at the Department of Biology, University of Graz. His research interests include self-organization, swarm intelligence, control systems, and swarm communication. For his PhD work associated with subCULTron, he developed a swarm design paradigm using minimalistic communication. In addition to having 5 years of experience working with industrial robotic arms, he worked on the EU-funded project subCULTron which developed and deployed an underwater robotic swarm in the lagoon of Venice, Italy.

Ronald Thenius received his PhD in Zoology at the University of Graz, Austria, and currently works as senior researcher in the Artificial Life Lab, also contributing to the project HIVEOPOLIS. Former projects he participated in are I-SWARM, SYMBRION and REPLICATOR, CoCoro and subCULTron. His field of research is swarm robotics and bioinspired algorithms. Besides this his main interest is the interaction of genes, neurons, hormones, and memes in artificial evolution, especially regarding evolution and embryogenesis of morphological and neuronal structures (e.g. evolution of the artificial brain). The "long-term goal" of his research is the development of an artificial open-ended evolutionary model, in which to observe the feedbacks between ecological processes, morphological adaptation, and trans-individual genetic interactions.

Franz Wotawa received an MSc. in Computer Science (1994) and a PhD in 1996 both from the Vienna University of Technology. He is currently professor of software engineering at the Graz University of Technology, the head of the Institute for Software Technology, and since 2017 the head of the Christian Doppler Laboratory for Quality Assurance Methodologies for Autonomous Cyber-Physical Systems. His research interests include model-based and qualitative reasoning, theorem proving, mobile robots, verification and validation, self-adaptive systems, and software testing and debugging. During his career, he has written more than 380 peer-reviewed papers for journals, books, conferences, and workshops. He supervised 90 master and 36 PhD students. He is a member of the Academia Europaea, the IEEE Computer Society, ACM, the Austrian Computer Society (OCG), and the Austrian Society for Artificial Intelligence and a Senior Member of the AAAI.

Manfred Füllsack received his PhD in philosophy at the University of Vienna, Austria. He is currently professor for systems sciences at the Institute for Systems Sciences, Innovation and Sustainability Research at the University of Graz, Austria. In his work, he focuses on game and network theory, cybernetics, the simulation of complex systems and multi-agent systems, and artificial intelligence.

Thomas Schmickl is full professor at the Institute of Biology at the University, Graz, Austria, where he supervises also the Artificial Life Lab, which he founded in 2007, after returning from an HHMI visiting professorship in the United States. In 2012, he was appointed the Basler Chair of Excellence at the East Tennessee State University (ETSU), USA. His research focusses on biology, ecological modeling, bioinspired engineering including swarm-, modular-, hormone-, and evolutionary-robotics. He teaches also ecological modeling and multi-agent simulations of biological and ecological systems. He was/is partner in the international EU-funded projects I-SWARM, SYMBRION, REPLICATOR, FloraRobotica and leading scientist and consortium coordinator of the EU grants CoCoRo, ASSISIbf, subCULTron, ATEMPGRAD, and HIVEOPOLIS. These projects all aimed at improving the current state of the art in robotics to let them be more similar to animals or plants in order to be adaptive, resilient, and flexible. Recently, living organisms became parts of his new bio-hybrid robotic systems, in order to form a sustainable technological symbiosis. Recently, he founded the Field of Excellence COLIBRI (Complexity of Life in Basic Research and Innovation) at the University of Graz, which is a network of 20+ full professors who research complexity sciences and join forces across various disciplines.