Self-organising Distributed Sensor Fusion Networks for Hierarchical Swarm Control and Supervision

Jonas D. Rockbach
Human-Machine Systems
Fraunhofer FKIE
Wachtberg, Germany

Isabel Schlangen
Sensor Data and Information Fusion
Fraunhofer FKIE
Wachtberg, Germany

Maren Bennewitz
Humanoid Robots Lab
University of Bonn
Bonn, Germany

***Abstract*—Mobile sensor networks can be realised using robot swarms where simple robots interact only locally to achieve swarm scalability and robustness. One of the main challenges is to develop suitable sensor fusion methods, both for autonomous swarms and human supervisory control, without gravely reducing the benefits of decentralisation. Hence, we introduce the Self-organising Hierarchical Extending (SHE) approach for distributed sensor fusion in robot swarms under communication and control constraints found in harsh, heterogeneous, and dynamic environments. The SHE approach is designed for fast autonomous swarm control in local environments, radially extending the sensor-effector range of a central node for global autonomous or human supervisory control. We discuss possible approaches to distributed object search and a SHE fusion architecture. An exemplary simulation based on the virtual pheromone approach provides a proof of concept for a self-organised, hierarchical, and extending fusion topology with a minimal robot controller.**

***Index Terms*—Swarm robotics, mobile sensor networks, distributed sensor fusion, hierarchical control, centralization vs. decentralization, human-swarm interaction, virtual pheromone.**

I. Introduction

Mobile sensor networks refer to platform collectives that can locally sense, compute, interact and move [1]. Such networks can be realised via robot swarms, i.e. groups of robots acting as one integrated agent based exclusively on local interactions [2]. Since the intelligence is located in the robot interaction network rather than in the individual agents, the computational capabilities of each robot can be low. Still, the combination of measurements collected by the swarm leads to broad coverage, improved accuracy, less false positives, and higher robustness against failure or interferences [3].

Robot swarms are envisioned in a range of scenarios, such as exploration and monitoring in search and rescue, space missions, sea exploration and monitoring, or military applications [4]. In such scenarios, a potentially large area needs to be covered simultaneously in challenging environments. The decentralised architecture of robot swarms in combination with the low asset cost of individual platforms are crucial advantages given the swarms' scalability and robustness.

In the above real-world applications, a human supervisor requires the capability of swarm control based on updated knowledge about the swarm's current state with regard to a specific mission [5]. In this context, one of the main challenges of swarm robotics is the design of a joint control and sensor fusion architecture for robot swarms [6] in accordance with the JDL data fusion model [7]. Therefore, this work proposes a Distributed Sensor Fusion (DSF) approach that enables global swarm control, however without significantly reducing the benefits of decentralisation.

Real-world applications, such as exploration and monitoring, pose constraints on sensor fusion capabilities. First, a central communication network such as Wireless Fidelity (WiFi) can be hard to apply robustly, hence enforcing the requirement of a flexible ad-hoc communication network between platforms (*communication constraint*) [1]. Second, some tasks can be time-sensitive, which would benefit from a faster, local sensor fusion as described e.g. in [8], in addition to the more time-consuming central fusion (*control constraint*) [9]. Given these constraints, the Self-organising Hierarchical Extending (SHE) approach to DSF is introduced in this paper, where the mobile sensor network self-organises into a topology that fuses information hierarchically through an extending tree around the central controller node as its root. We demonstrate that even a simple robot controller can lead to the desired fusion topology and that fusion nodes emerge based on the distance of objects, thereby implementing a form of embodied computation [10], [11]. In embodied computation, the world-embedded network is seen as the parallelised computing unit itself and the node locations are utilised as information. Taken together, this work contributes a high-level design for DSF, called SHE-DSF, and demonstrates that simple controllers can be utilised for this approach. Given its broad applicability to both autonomous and human supervisory swarm control, our approach is relevant for swarm control design in general.

This contribution is structured as follows. In Section II, an exploration and monitoring scenario is outlined from which communication and control constraints are derived. Based on this, the SHE approach to DSF is introduced in Section III. Possible approaches to both distributed object search and fusion network self-organisation in the context of SHE-DSF are discussed. Section IV describes an illustrative proof-of-concept simulation for hierarchical extending fusion network self-organisation based on the virtual pheromone approach. It is shown that fusion nodes emerge in the network based on the distance of objects, rendering SHE-DSF a form of embodied computation. Finally, Section V concludes this work.

Corresponding author's contact email: jonas.rockbach@fkie.fraunhofer.de.

II. Scenario and resulting constraints

A. Exploration and Monitoring Scenario

This work builds on scenarios in which an area, here described as $\mathbb{A} \subset \mathbb{R}^2$, needs to be covered by an artificial agent for exploration or monitoring a set of objects of interest O in harsh, heterogeneous, and dynamic environments. These include space, underwater, scenarios with bad weather conditions, or challenging anthropogenic conditions (e.g., interferences). The properties of the environment are defined as:

- **Harsh**: A harsh environment implements a high degree of noise or interferences that disturb reliability, freedom of movement, and communication of agents.
- **Heterogeneous**: A heterogeneous environment contains diverse objects of interest in different areas $A \subseteq \mathbb{A}$.
- **Dynamic**: A dynamic environment changes the features of regions or the objects of interest O on it.

This makes a theoretically conceivable swarm agent S a good choice since it can cover $\mathbb{A}$ simultaneously based on its scalability while being robust to a certain degree of node failures [2]. In general, S acts in the mission area $\mathbb{A}$ based on the commands of a central controller c at (x_c, y_c) (e.g. a human supervisor) and reports collected information to c.

In *exploration*, S needs to search a given area $\mathbb{A}$ in order to find objects of interest O. We assume that a map is either available or provided via other means, e.g. Simultaneous Localisation and Mapping (SLAM) [12]. In search and rescue, for example, S may explore an area of interest $\mathbb{A}$ in order to provide information of possible victims in terms of their states $o \in O$ to the human supervisor c, with $o = [x_o, y_o, w_o]$, where $[x_o, y_o]$ is the object's position on the map and w_o is the victim's severity or need for treatment (triage). The property w_o may also stand for other needs of attention, e.g. the severity of pollution sources or the threat level of an object to the own system, such as in space or military missions. An exploration mission is successfully performed if all objects in O have been reported accurately to c within a given time limit.

In turn, *monitoring* is the observation of changes in O. For example, in underwater monitoring, an area A may be observed through S for changes in pollution concentration by monitoring a set of contaminating sources O, where the property w_o represents the object's polluting impact. The success of a monitoring mission is defined by the accurate report of changes in O to c in a certain time interval.

Exploration and monitoring missions are often combined, for example in pollution monitoring where the contaminating sources need to be found first and are then observed over a given time window. Taken together, the swarm S must search for objects of interest O as well as monitor changes in O while reporting these to the central controller c within a given time frame. A simplified version of exploration and monitoring is the *best-of-n* problem that is studied in swarm robotics [13], where, instead of the whole set O, only the most important object needs to be identified. This approach is used e.g. in the search-and-rescue domain, where the agent needs to decide where to go first based on a triage [9].

B. Communication and Control Constraints

Harsh, heterogeneous, or dynamic environments can lead to robot and communication failures. Furthermore, it can become impossible to achieve a reliable central communication network for wireless report and control. As an alternative, the swarm's local communications can be used as a self-organising ad-hoc network between relevant nodes that can adapt to their own states and the environment [1].

Based on the swarm communication network, sensor fusion is to be implemented for both autonomous swarm and human supervisory control. For global command and control, the sensor fusion could be directly placed at a central node. However, it has been argued that this would reduce swarm capabilities [6], [9]. Therefore, we envision an embodied DSF for robot swarms. For autonomous swarm control in harsh, heterogeneous, or dynamic environments, locally fused information is required directly in local regions A for a fast adaptation to the current situation [9]. Local fusion results can then be combined successively to global information for overall autonomous or human supervisory control.

Taken together, two types of constraints for DSF in robot swarms are defined [9]:

- **Communication constraint**: Challenging environments make the application of a central communication network difficult. Communication is therefore implemented by the swarm network itself through self-organising into a robust communication network between relevant nodes.
- **Control constraint**: Challenging environments require reactive closed-loop control at local regions. Swarm control therefore requires DSF both at local nodes and at a central node to achieve autonomous local as well as global autonomous and human supervisory control.

III. Self-organising Hierarchical Extending DSF

A. Swarm Definition

A swarm agent is defined as a collection of locally interacting robot agents $S = \{r_1 ... r_N\}$, N being the swarm size. A homogenous collection of robots is assumed, which are placed onto $\mathbb{A}$ with positions $[x_r, y_r]$ and heading $\theta_r \in \{0 ... 2\pi\}$. The speed v_r applied to the robot configuration defines its forward kinematics. Note that the controller c is not contained in S by default since in general we have no control over its actions.

Robots can estimate their relative bearing $b_{r'}, b_o \in \{0 ... 2\pi\}$ to other swarm members r' and to objects in $o \in O$ and measure the state w_o of an object if it is in the robot's sensor range d_{sensor} in terms of the Euclidean distance (Fig. 1). Bearing may for example be estimated via directed infrared [11], or image classification [14]. The measurement $\hat{w}_o$ of w_o depends on the object type and scenario. Here, the sensor estimates $\rho_o = [\hat{b}_o, \hat{w}_o]$ are treated as random variables subject to noise.

Communication connections between robots in S (including the controller c) are established if the nodes' relative Euclidean distance is smaller than their bi-directional communication range d_{com} (Fig. 1). Thus, the communication graph G_{com} at a certain time step has elements of S and the root c as its

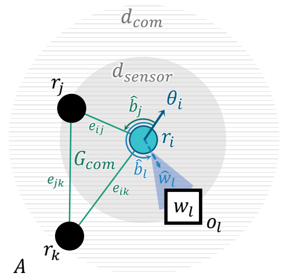

Fig. 1. Different interactions as used in this work. The blue node r_i with heading θ_i measures the relative bearing $\hat{b}_l$ and the state $\hat{w}_l$ of an object o_l with true state w_l, as well as the bearing $\hat{b}_j$ of another node r_j inside its detection range d_{sensor}. Node r_k cannot be sensed but is still within r_i's communication range d_{com}, hence belonging to the same graph G_{com}.

nodes and establishes communication connections as its edges E, $e \in E$. Taken together, the swarm configuration is defined by the robot configuration set $q_S = \{[x_r, y_r, \theta_r], r \in S\}$, and $q_S \cup c$ yields a graph G_{com} by consideration of d_{com}.

B. SHE-Approach to Distributed Sensor Fusion

Based on the constraints defined in Section II-B, the swarm is designed as a Self-organising Hierarchical Extending DSF (SHE-DSF) network, inspired by the self-organising nervous system architecture that successively fuses local sensor data into holistic percepts over layers [9], [17]. SHE is defined as

- **Self-organising**: The placement of nodes w.r.t. c, O, and $\mathbb{A}$ is fundamentally based on local robot information, with the possibility to use global data as guidance if available.
- **Hierarchical**: The swarm fusion and control network follows a hierarchy, meaning that local sensor fusion results are combined successively to global information based on object positions for local and global control.
- **Extending**: The swarm network extends the sensor-effector range of the central controller c by implementing a radial tree topology around it.

A *self-organising* network mainly uses local interactions to look for objects of interest and to build a suitable swarm configuration for DSF. While the communication graph G_{com} includes all possible inter-node connections as its edges based on the maximum communication distance d_{com}, the self-organising network must form a suitable sensor fusion network $G_{\text{sf}} \subset G_{\text{com}}$ with c as its root to attend the objects in O.

Fig. 2 shows an illustration of *hierarchical* DSF in the SHE approach. The fusion network is a directed subgraph $G_{\text{sf}} \subset G_{\text{com}}$ with its gradient pointing towards c. In order to allow for a fast information fusion and transmission, a movement constraint for robots r^* in G_{sf}, e.g. $v_{r^*} = 0$, should be implemented in order to stabilise the communication stream while free robots can continue their search [9].

The swarm network organises itself in a way that individual sensor streams are fused near object clusters first, i.e, at local regions $A \subseteq \mathbb{A}$, such that local fusion results can be used directly by the swarm agent for autonomous local control. Therefore, the SHE approach to DSF is an example of embodied computation since the sensor nodes' spatial distances

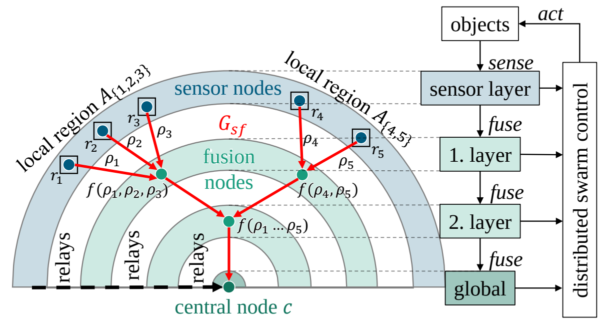

Fig. 2. Hierarchical sensor fusion as a form of embodied computation. Robots close to objects inject sensor input into the network, while local information is fused over several layers towards the central controller node based on the distance between sensed objects. The layered fusion can be used for distributed swarm control close to the region of injection. Relay nodes are not shown.

determine the proximity of fusion nodes and therefore the availability of local swarm control. For example, if a local fusion node receives multiple victim detections in a search-and-rescue scenario, it can rapidly attract further deployments.

Although S is homogenous, it is helpful to differentiate between three distinct roles for robots in G_{sf}. A node in G_{sf} is called *sensor node* if at least one $o \in O$ is in its sensor range d_{sense}. A sensor node injects its sensor estimations ρ of an object into the fusion network G_{sf}. A robot that receives a message and forwards it according to G_{sf} towards c is called a *relay node*. In contrast, a *fusion node* is a robot that receives multiple incoming streams and integrates isolated local data into fused information with some fusion function f, whose output is again injected into G_{sf}. Thus, there are two types of messages: sensor estimates ρ and fused information f. Given a limited communication channel, a fusion node may only forward the fused information, although providing raw sensor estimates can be desirable, e.g. for human supervision. In sum, local information is fused hierarchically over fusion layers in the direction of the central controlling node c, where each layer is defined by all fusion nodes with the same number of hops between themselves and c, not counting relay nodes. Note that node roles are assumed mutually exclusive in this paper, which could however be generalised in future work.

Extending DSF in turn is shown in Fig. 3, which shows that the swarm is seen as a radial tree extension of a central controller c. The central controller can be interpreted like a spider located in the centre of its web [15], receiving sensory information from all directions through the extending swarm net. Depending on the situation, only parts of the surrounding area can be covered, with the constraint that there is always a connection between c and S or potential subswarms [16].

For each real-world application, the central controller node c may be implemented differently depending on the context. In proximal control, where controller and swarm are deployed in the same area, the controller could either be a sophisticated computing unit [17], a human operator being part of the swarm [9], or a mixture between the two, such as a manned system. For example, a search-and-rescue operator could look

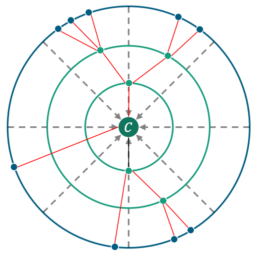

Fig. 3. Extending sensor fusion as a radial tree around the central node. Thereby, the swarm extends the sensor-effector range of the central controller c. The colour coding is the same as in Fig. 2.

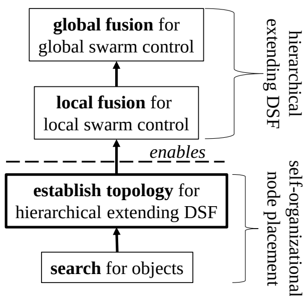

Fig. 4. Summary of the steps for SHE-DSF. Establishment of a hierarchical extending fusion topology by self-organisation is the focus of this work.

for survivors in the same area as the swarm while the swarm extends the operator's capabilities [16]. For remote control, where controller and swarm are located in different areas, the central node could be realised as a relay between controller and swarm [18], e.g. via satellite communication. In underwater mission supervision, for example, a remote command and control station is coupled to a communication relay swimming on the sea's surface that takes the role of the central node. Both in proximal and remote control, the central controller implements a leader functionality for the swarm [5].

The high-level steps for SHE-DSF are summarised in Fig. 4. The first two steps describe the self-organisational node placement for distributed object search and perception, as well as the establishment of the desired hierarchical extending fusion network topology. The node placement is fundamental for SHE-DSF, both for autonomous and human supervisory control. Possible approaches to self-organisational node placement are described below, whereas the detailed implementation of the fusion function f depends on the specific application and is out ot scope for this work. The focus here is on the self-organisational establishment of the SHE fusion topology.

C. Distributed Object Search

First, the robots in S must be placed such that objects of interest are in their sensor range. Given a sufficiently large swarm in a constrained area $\mathbb{A}$, a naive strategy is to randomly explore $\mathbb{A}$ [9] or to disperse into $\mathbb{A}$ for full coverage [16]. Dispersion can be based on local repulsion-attraction models, which are common in swarm robotics [2], [11].

A controlled approach for informed sensor coverage is described in [19]. In general, swarm robotics applies models based on local feedback loops, in which robots are allocated to object concentrations [2]. An interesting approach to object localisation was proposed in [20]. Here, robots exchange their local beliefs of object locations when they are in each others' communication range, while the resulting heat map can be visualised for the human supervisor. An optimal swarm strategy for target tracking was discussed in [21].

Even with scalable swarms, the assumption of reasonable coverage of $\mathbb{A}$ given a time limit can be invalid, e.g. if $\mathbb{A}$ is large. A strategy for extending object search with small sets of robots was suggested by Nouyan et al. [14]. Here, a dynamic robot chain is extended from a central node and rotated around it. In the context of the SHE-DSF, one could establish robot chains that sweep around the central node like a scanning radar beam and recruit robots for sensor stream establishment as soon as an object has been found.

D. Fusion Network Self-organisation

The desired hierarchical extending G_{sf} topology should be the result of self-organising the communication topology G_{com}, i.e. the mobile sensor network. Establishing a sensor stream between sensor nodes and the central node corresponds to establishing robot chains. For example, Nouyan et al. showed the self-organisation of robot chains between a central node and targets of interest by local interactions [14].

A hint towards the self-organisation of a hierarchical extending topology was provided by Payton et al. [11]. In the virtual pheromone approach, a signal is injected into the communication network by the central node and diffuses radially into the network. The distributive algorithm selects shortest paths between nodes and thereby forms a radial tree around the central node. Even a random, i.e. uninformed, search process can be sufficient to establish the radial tree given enough robots in a confined environment as shown in [9]. In the following, we show that hierarchical extending fusion indeed emerges based on the spatial distance of objects based on the virtual pheromone approach.

IV. Example of SHE-DSF

A. Simulation Setup

In line with swarm robotics, we search for minimal microlevel rules that equip the swarm with the desired macrolevel SHE-DSF capability. For this purpose, a simple two-dimensional grid world simulation is implemented in MATLAB 2020b for proof-of-concept investigations, with grid nodes located at $1\,\mathrm{m}$ distances in each dimension. The total area of interest $\mathbb{A}$ spans $100\,\mathrm{m} \times 100\,\mathrm{m}$. A static controller node is placed in the centre point $(50\,\mathrm{m}, 50\,\mathrm{m})$ of the grid, with N robots randomly initiated at a maximum distance of $10\,\mathrm{m}$ to it. Furthermore, we set $d_{\text{sensor}} = d_{\text{com}} = 10\,\mathrm{m}$.

B. Robot Behaviour

The robots' heading θ_r and speed v_r are calculated as a summed potential field based on a random walk force, object

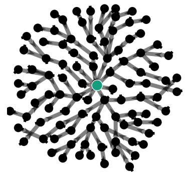

Fig. 5. Example of a virtual pheromone diffusion graph G_{dif}. The central node injects a message into G_{com}, thus forming a radially extending tree.

attractor force, and a dispersion force. The robot movement is implemented as a simplified correlated random walk with $v_r \in \{0,1\}$, in which robots either keep their heading or spontaneously turn by $\pm\pi/4$. While an object is in the sensor range of a robot, the object attractor vector points towards that object, otherwise it is zero. The dispersion force ensures the dispersion of the robots into the environment while holding communication lines. Robots repel each other if their relative Euclidean distance is $\leq 9\,\text{m}$, and attract each other if they are between $10\,\text{m}$ and $11\,\text{m}$ apart [11]. Robot communications can fail in each time step with an individual probability of 0.001. No noise is implemented for proof-of-concept purposes.

C. Algorithm for Fusion Network Self-organisation

Algorithm 1 shows the pseudocode for SHE-DSF. A virtual pheromone diffusion algorithm [11] is used to determine which robots participate in a shortest path between c and objects in O. At each time step, c injects a diffusion message into G_{com}. Each robot saves the bearing $\hat{b}_{r'}$ towards the transmitting robot, however only the first one received in a single time step. Thereby, the message diffuses radially along the shortest paths into G_{com} and forms the directed diffusion graph G_{dif} like in Fig. 5. If a robot is a sensor node, it injects a return message into G_{com} in the direction of the received diffusion message as given by G_{dif}. A robot that receives both a diffusion and a sensor node message is part of G_{sf} and sets its own speed $v = 0$. An estimation on how long the swarm needs to build G_{sf} for different swarm sizes can be found in [9]. In general, larger swarms enable faster G_{sf} establishment.

Algorithm 1 Pseudocode for SHE-DSF

Initialize $\mathbb{A}, S, O$
while Self-organise DSF **do**
  $G_{\text{com}}, G_{\text{dif}}, G_{\text{sf}} = \{\}$
  Determine G_{com} based on $(q_S \cup c, d_{com})$
  Inject diffusion message via c into G_{com}
  Add visited nodes to G_{dif} with primary $\hat{b}_{r'}$
  if Sensor node receives diffusion message **then**
    Inject return message via sensor node into G_{dif}
    Add visited nodes to G_{sf}
  end if
  Inject ρ from sensor nodes into G_{sf}
  Set $v = 0$ for all robots in G_{sf}
  Move swarm S
end while

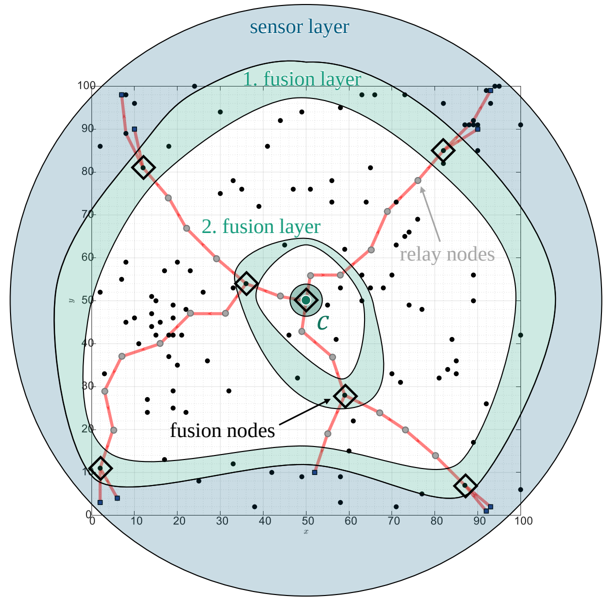

Fig. 6. Example run of the simulation after 299 time steps with $N = 150$ and nine objects randomly chosen at the corners of the grid. The colour coding is the same as in Fig. 2, with the edges of G_{sf} shown. It can be seen that a hierarchical extending fusion network topology emerges.

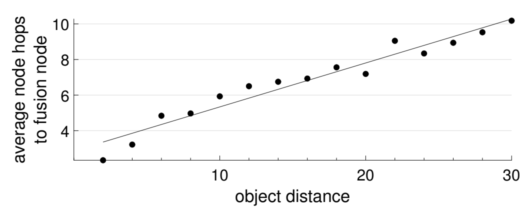

Fig. 7. Validation of embodied computation in SHE-DSF showing average relay node hop distance between sensor nodes and fusion node as a function of the distance between two objects. The linear relationship confirms that the network tends to fuse information of adjacent objects earlier compared to that of objects which are further apart.

D. Results

Fig. 6 illustrates how simple rules can already lead to the desired hierarchically extending topology. The embedded swarm extends the sensor range of the central node c by self-organising into a $360°$ hierarchical fusion network as a function of O on $\mathbb{A}$. Comparing Figs 5 and 6, it becomes evident that $G_{\text{sf}} \subseteq G_{\text{dif}} \subset G_{\text{com}}$. The directed networks may be used as a movement gradient by agents [11], [16].

Embodied computation of the embedded fusion network is demonstrated by Fig. 7, which shows the average relay node hops from sensor nodes to the fusion node combining their measurements as a function of the distance between two measured objects. In this scenario, the objects are placed $30\,\text{m}$ from c over 100 Monte Carlo simulations. As can be seen, a larger spatial distance between objects tends to result in the data fusion occurring further apart and hence closer to c.

In exploration and monitoring, the fusion network provides c with information on the location and state of objects. The estimation of the bearing angle of an object w.r.t. c can be geometrically determined if the network forwards individual bearing estimations of robots. Node hops can additionally be used as distance approximations. Alternatively, the gradient of G_{sf} can be used for an approximate bearing estimation [11].

The fusion network can also be used to prioritise objects by injecting and fusing the measurement of their parameter w and implementing a suitable distance measure [22]. An illustrative scenario is implemented in which four objects are randomly initiated at different positions in $20\,\mathrm{m} \times 20\,\mathrm{m}$ areas at the corners of $\mathbb{A}$ with $w = 1$, run over 500 samples with $N = 120$ nodes. In this *best-of-4* problem, the distance is defined via the cost to reach equally valuable objects from the central node, meaning that the closest object should be prioritised. Distance measurements are simplified as hop counts of the sensor streams. The Monte Carlo experiment is in accordance with [22] and shows that in 92% of the runs, following the streams with the lowest hop count leads to the closet target. Errors can occur if robot chains are not minimal, i.e. if the already established chain takes a detour. However, the error decreases with time assuming that enough robots are still free to search for shorter paths. The error can be decreased further by straightening the pathways [14].

Note that as demonstrated in [16], the described approach is adaptive to changes in object position and can deal with communication and robot failures since the algorithm establishes shortest paths repeatedly. However, the described method does not yet implement robustness of the fusion topology: an unfavourable single robot failure in G_{sf} can lead to the loss of a whole branch. To prevent such events, pathway robustness can be strengthened by attracting other robots nearby to implement node redundancy by using the virtual gradient of the network [16].

V. Conclusion

In this work, the Self-organising Hierarchical Extending (SHE) approach to Distributed Sensor Fusion (DSF) was introduced to cope with communication and control constraints of swarms found in real-world scenarios. In SHE-DSF, the robot swarm distributively searches for objects of interest and self-organises into a hierarchical topology that radially extends around a central controller node. A proof-of-concept simulation has shown how even a simple robot controller leads to the self-organisation of the hierarchical extending fusion network topology, making it suitable for swarms made of robots with limited computational power and sensor reach. The approach could be extended in future work by including a sophisticated distributed object search, the optimisation of inter-robot communications, shortening streams via dynamic node placement, increasing robustness of streams by robot redundancy, as well as considering a moving central node.

SHE-DSF has implications for both autonomous and human supervisory swarm control. A sophisticated machine node can fuse information and allocate subswarms to local regions in a centralised way, while the subswarms remain autonomous for local, distributed computations. Similarly, a swarm can act as a semi-autonomous extension of the human supervisor, resolving time-critical situations itself while still being under task-based control of the operator. Moreover, SHE-DSF is suitable for both proximal and remote swarm control, which makes it a versatile tool for many applications in swarm robotics.

Acknowledgement

J.D. Rockbach thanks Heiko Hamann for helpful feedback on an earlier version of this manuscript.

References

[1] A. Howard, M. J. Matarić, and G. S. Sukhatme, "An incremental self-deployment algorithm for mobile sensor networks," *Autonomous Robots*, vol. 13, no. 2, pp. 113–126, 2002.

[2] H. Hamann, *Swarm robotics: A formal approach*. Springer, 2018.

[3] S. S. Blackman and R. Popoli, "Design and analysis of modern tracking systems," *(No Title)*, 1999.

[4] M. Schranz, M. Umlauft, M. Sende, and W. Elmenreich, "Swarm Robotic Behaviors and Current Applications," *Frontiers in Robotics and AI*, vol. 7, p. 36, 2020.

[5] A. Kolling, P. Walker, N. Chakraborty, K. Sycara, and M. Lewis, "Human Interaction with Robot Swarms: A Survey," *IEEE Transactions on Human-Machine Systems*, vol. 46, no. 1, pp. 9–26, 2016.

[6] J. C. Barca and Y. A. Sekercioglu, "Swarm robotics reviewed," *Robotica*, vol. 31, no. 3, pp. 345–359, 2013.

[7] D. Hall and J. Llinas, *Handbook of multisensor data fusion*. CRC press, 2001.

[8] D. Nicholson, C. Lloyd, S. Julier, and J. Uhlmann, "Scalable distributed data fusion," in *Proceedings of the Fifth International Conference on Information Fusion. FUSION 2002.(IEEE Cat. No. 02EX5997)*, vol. 1. IEEE, 2002, pp. 630–635.

[9] J. D. Hasbach and M. Bennewitz, "The design of self-organizing human–swarm intelligence," *Adaptive Behavior*, vol. 30, no. 4, pp. 361–386, 2022.

[10] H. Hamann and H. Wörn, "Embodied computation," *Parallel Processing Letters*, vol. 17, no. 03, pp. 287–298, 2007.

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

[12] M. Kegeleirs, G. Grisetti, and M. Birattari, "Swarm SLAM: Challenges and perspectives," *Frontiers in Robotics and AI*, vol. 8, p. 618268, 2021.

[13] G. Valentini, E. Ferrante, and M. Dorigo, "The best-of-n problem in robot swarms: Formalization, state of the art, and novel perspectives," *Frontiers Robotics AI*, vol. 4, no. MAR, 2017.

[14] S. Nouyan, A. Campo, and M. Dorigo, "Path formation in a robot swarm," *Swarm Intelligence*, vol. 2, no. 1, pp. 1–23, 2008.

[15] H. F. Japyassú and K. N. Laland, "Extended spider cognition," *Animal Cognition*, vol. 20, no. 3, pp. 375–395, 2017.

[16] J. D. Rockbach, "Enhancing Human Self-Regulation with Controllable Robot Swarms Acting as Extended Bodies," in *2023 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2023, *in press*.

[17] N. Mathews, A. L. Christensen, R. O'Grady, F. Mondada, and M. Dorigo, "Mergeable nervous systems for robots," *Nature Communications*, vol. 8, no. 1, pp. 1–7, 2017. [Online]. Available: http://dx.doi.org/10.1038/s41467-017-00109-2

[18] S. Bashyal and G. K. Venayagamoorthy, "Human swarm interaction for radiation source search and localization," *2008 IEEE Swarm Intelligence Symposium, SIS 2008*, 2008.

[19] J. Cortes, S. Martinez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," *IEEE Transactions on robotics and Automation*, vol. 20, no. 2, pp. 243–255, 2004.

[20] M. Divband Soorati, J. Clark, J. Ghofrani, D. Tarapore, and S. D. Ramchurn, "Designing a User-Centered Interaction Interface for Human–Swarm Teaming," *Drones*, vol. 5, no. 4, p. 131, 2021.

[21] H. L. Kwa, J. L. Kit, and R. Bouffanais, "Optimal swarm strategy for dynamic target search and tracking," in *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, 2020, pp. 672–680.

[22] A. Campo, Á. Gutiérrez, S. Nouyan, C. Pinciroli, V. Longchamp, S. Garnier, and M. Dorigo, "Artificial pheromone for path selection by a foraging swarm of robots," *Biological cybernetics*, vol. 103, no. 5, pp. 339–352, 2010.