Embodying swarm coordination in robotic cyber-physical systems

Mario G.C.A. Cimino
Dept Information Engineering
University of Pisa
Pisa, Italy
mario.cimino@unipi.it

Pierfrancesco Foglia
Dept Information Engineering
University of Pisa
Pisa, Italy
pierfrancesco.foglia@unipi.it

Salvatore Arancio Febbo
Dept Information Engineering
University of Pisa
Pisa, Italy
salvatore.aranciofebbo@ing.unipi.it

Cosimo A. Prete
Dept Information Engineering
University of Pisa
Pisa, Italy
antonio.prete@unipi.it

***Abstract*—The rise of deep learning and cloud computing have changed the design paradigms of swarm robotics, giving a higher level of embodiment to robotic cyber-physical systems. This paper shows how real-world scenarios can be developed in an integrated platform made of autopilot, robotic operating system and 3D simulator. We focus on two key swarm intelligence mechanisms: stigmergy, for indirect coordination via virtual pheromones, and flocking, for decentralized and flexible movement. Developing a series of pilot experiments on real-world scenarios, and a platform built on PX4, Gazebo, and ROS2, the embodiment of such mechanism is shown. Early results highlight the effectiveness of this approach in promoting fault tolerance, adaptability, and efficient exploration in scenarios such as wildfire detection, posts-earthquake recovery, and underwater monitoring.**

Keywords—swarm robotics; flocking; stigmergy; autopilot; robotic operating system; 3D simulator.

I. INTRODUCTION AND BACKGROUND

Nowadays unmanned robots have wide availability of deep learning, remote and proximity sensing, "on board and in the cloud". This makes novel design paradigms of swarm robotics necessary: swarms can operate in a surrounding cyber-physical environment having various sources of information. In addition, such information is characterized by dynamic levels of uncertainty and unreliability [1]. To fix ideas, let us consider some representative case studies: (i) *unmanned aerial vehicles* employed for surveillance, delivery, or mapping (e.g. early detection of wildfire) [2]; (ii) *unmanned ground vehicles* used in logistics, agriculture, waste collection (e.g. illegal dump detection) [3]; (iii) *unmanned underwater vehicles* adopted for oceanography or pipeline inspections (e.g. seagrass meadow monitoring) [4]; (iv) *unmanned marine surface vehicles* exploited for maritime operations (e.g. ocean plastic collection or oil spill incident) [5]. In all case studies, robots miniaturization is a key factor to move from a distributed robotic system, that is a team of robots with an orchestrated coordination logic, to a swarm robotic system, that is a swarm of drones guided by multi agent and swarm intelligence principles, promoting fault tolerance, scalability and adaptability [6].

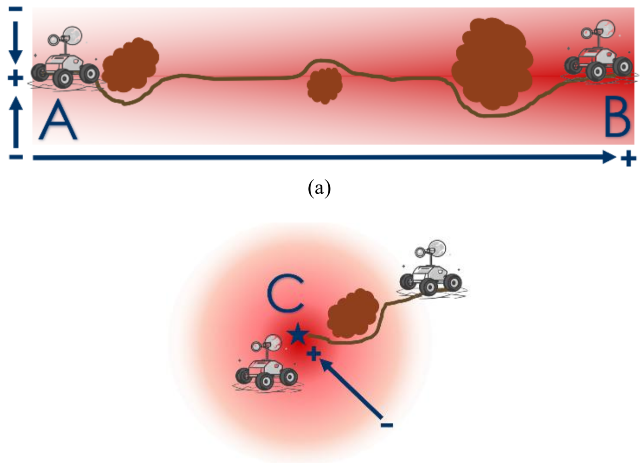
Robot miniaturization is also fostered by the *remote brain* paradigm, a cloud-computing approach for robots where computationally intensive tasks, like artificial intelligence-based decision-making, sensor processing, mission planning, are offloaded to a centralized cloud or edge server, rather than relying solely on the robot's onboard hardware [7]. This approach transforms a robot into a "thin client" that depends on a personal or shared remote brain for high-level intelligence, while retaining basic autonomy for low-level control (e.g., stabilization and obstacle avoidance). The remote brain paradigm is compliant with swarm robotics, in the sense that robots can still interact with the surrounding cyber-physical environment and give feedback to modify it, having only local perceiving and actuation, without exploiting centralized control and global knowledge. This sort of "information hiding" [8] in the cloud allows for a better scalability and exploration. As an example, let us think about a swarm of robots performing illegal dump detection: a single robot detecting illegal dump might recruit some neighbouring robots to complete its processing, not disturbing other remote robots' exploration.

On the other hand, cyber-physical systems allow for *virtual sensing and actuation*. Let us consider two fundamental swarm coordination mechanisms: *stigmergy* and *flocking* [6]. Stigmergy is a bio-inspired communication and coordination pattern among robots, to achieve complex collective behaviours without direct communication, central control, or global awareness. For example, ants leave pheromone trails in the environment to guide each other to food sources. Agents in a cyber-physical system can deposit virtual pheromones on a digital map. An agent might release an attractive/repulsive pheromone to mark an interesting/non-interesting location. Other agents can follow/leave that trail to efficiently explore the environment. Thus, a single robot can sense a virtualized pheromone trail and release a virtualized pheromone mark in the environment. This pheromone map can be handled on the cloud to make available a local portion of it for the robot upon request. Pheromone properties, such as its temporal decay over time, can be managed as a service in the cloud [6]. Flocking is a self-organized, decentralized behaviour in which a group of robots moves together in a cohesive and coordinated manner, like a flock of birds or a school of fish, without a leader. It is based on three fundamental rules: separation (to maintain a minimum distance from neighbours), alignment (to match velocity to the average velocity of neighbours), and cohesion (to steer toward the average position of neighbours). Flocking enables search and rescue, environmental monitoring, surveillance, and formation control [6]. Thus, a single robot can see its neighbours in the virtualized map of robots, and compute average distance and velocity in the environment.

Another significant aspect of cyber-physical systems is the rise of *robotic digital twins*, i.e., virtual, dynamic replicas of physical robots mirroring their real-time state and behaviour. Robotic digital twins have also a view in 3D simulation environments, allowing for computational simulation of highly realistic missions. Software code developed for a digital twin can be compiled and deployed in the real robot, thus achieving several software engineering advantages, considering the heterogeneous available robotic platforms in the market. Further, the availability of a simulative environment allows for the development of testbeds where to perform parametric optimization on high performance computers [6]. Overall, this technological ecosystem can replicate real-world conditions, where embodiment and emergent phenomena can effectively occur. *Embodiment* in swarm intelligence refers to the principle that the physical form, sensory capabilities, and environmental interactions of individual agents fundamentally shape the collective behaviour of the swarm. Unlike cognitivist algorithms, developed with a top-down paradigm upon restrictive assumptions and tested on simplified simulators to verify some properties, embodiment allows for the development of bio-inspired logics in an ecosystem with a bottom-up paradigm. Hence, industrial simulators based on realistic environments and technology can highly foster emergent paradigms of design.

In the technological environment above depicted, a lot of information provided by information systems, deep learning, remote and proximity sensing, is characterized by uncertainty. For example, deep learning models are naturally affected by uncertainty caused by limited training data. In addition, data uncertainty arises from noisy or ambiguous data (e.g. granulated satellite data, under sampled sensory data, environmental noise). Human knowledge can also be uncertain for the limited historical experience under specific circumstances. In general, big data collected on specific context can provide only partial information when considered on a broader context. For example, in early fire detection on a specific forest, several natural and human-related factors contribute to the risk of fire: climate and weather, topography, vegetation type, land use, human activities, historical fire patterns, and so on. Some factors can be provided by exploiting satellite or geographical data classified or regenerated with deep learning. In general, uncertainty comes from many different sources, such as data collection, concept variance, multimodality, incompleteness, inconsistency of the available data and models. Even with accurate information, uncertainty arises from the environment. For example, while moving a robot from point A to B with centralized coordination, obstacles can easily limit the planned waypoint. In this case, continuous logic is very suitable to create flexible waypoints. A stigmergic waypoint in a 3D environment extends each segment as a sort of cylindric shape having greater intensity the greater the proximity to the axis and to the target. Fig. 1a shows a bidimensional example of such stigmergic coordination: here, stigmergic intensity increases from A to B and from borders to axis. When moving from A to B, the robot follows the direction of higher stigmergic intensity. While an obstacle is detected, the on-board obstacle avoidance is triggered, i.e. a control logic having higher priority with respect to coordination logic. In decentralized coordination, a single robot can release a stigmergic attractive potential to recruit other robots while detecting targets. Fig. 1b illustrates the concept: here, intensity increases towards the target C. The stigmergic paradigm includes also temporal decay, to enable emergent effects when operating in flocks [6]. Similarly, with respect to leaded formations, flocking can operate with environmental uncertainty and obstacles.

In this paper stigmergy and flocking are embodied in 3D cyber-physical systems, to shows how such fundamental design patterns guarantee interoperability between control logic and coordination logic, as well as representation paradigms of sensing and actuation.



(a)

(b)

Fig. 1. Bidimensional pattern of stigmergic coordination.

Fig. 2 illustrates a logical representation of a coordination station in a robotic cyber-physical system. Here, on the right, a variety of robots provided by the market is represented. On the left, the coordination station provides the main logical components: digital twins, remote brains, stigmergy, virtual sensor/actuator, orchestrator, and optimization.

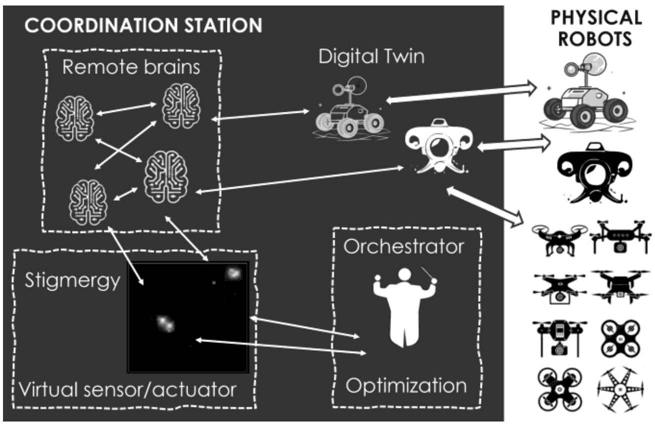


Fig. 2. A logical representation of a coordination station in robotic cyber-physical systems.

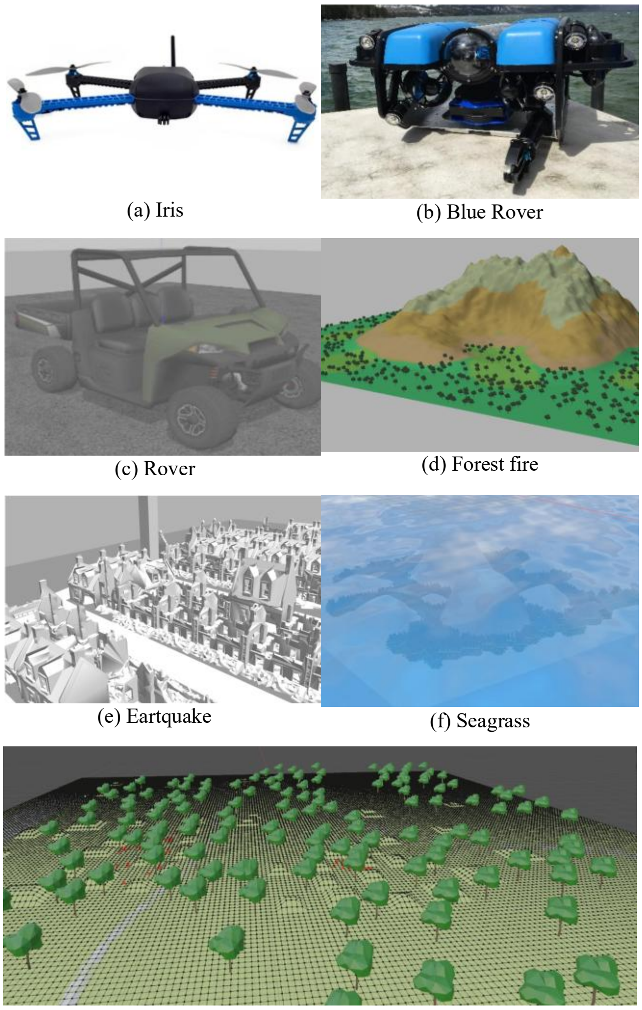
To show the effectiveness of the approach, a coordination platform has been developed and experimented on different real-world scenarios. Specifically, Section 2 covers materials and methods, whereas Section 3 conclusions and future works.

II. Materials and Methods

Table I summarizes the development frameworks and responsibilities adopted in the implemented system. Specifically, the digital twin is based on the PX4 autopilot framework (px4.io), an open-source control system that can provide hardware-in-the-loop simulation if integrated with simulators where digital twin of robots and its environment is modelled. As a robot simulation environment, Gazebo classic has been used (classic.gazebosim.org), an open-source simulator widely used for testing and developing autonomous systems. It provides a physics-based 3D simulator to model robots, sensors, and environments with high fidelity. Finally, the coordination logic has been developed under ROS2 - Robot Operating Systems 2 (github.com/ros2), an open-source framework for robotic systems. Fig. 3 shows vehicles (a-c) and environments (d-g) developed in the testbed. All environments are modelled from real-world cases. For instance, Earthquake is a realistic GIS model of a small town in central Italy, Amatrice, that in 2016 was hit by a 6.2 magnitude earthquake. Fig. 4 shows a snapshot of a blue rover equipped with a photo camera to detect seagrass in undersea environment. Here, it is also shown the picture taken by the camera, which is then processed by the control station to detect seagrass.

TABLE I. DEVELOPMENT FRAMEWORKS AND RESPONSIBILITIES ADOPTED IN THE IMPLEMENTED SYSTEM

Sub-system	Development framework	Responsibility
Digital Twin	PX4 Autopilot	Robot control logic: onboard sensors, vehicle stability and speed management logic, actuators, etc.
Coordi nation Logic	ROS2	Robot coordination logic system: remote brain, stigmergy, visual sensor/actuator, orchestsrator, optimization
Simula tor	Gazebo Classic	Environment, robot, target, and equipment modelling system

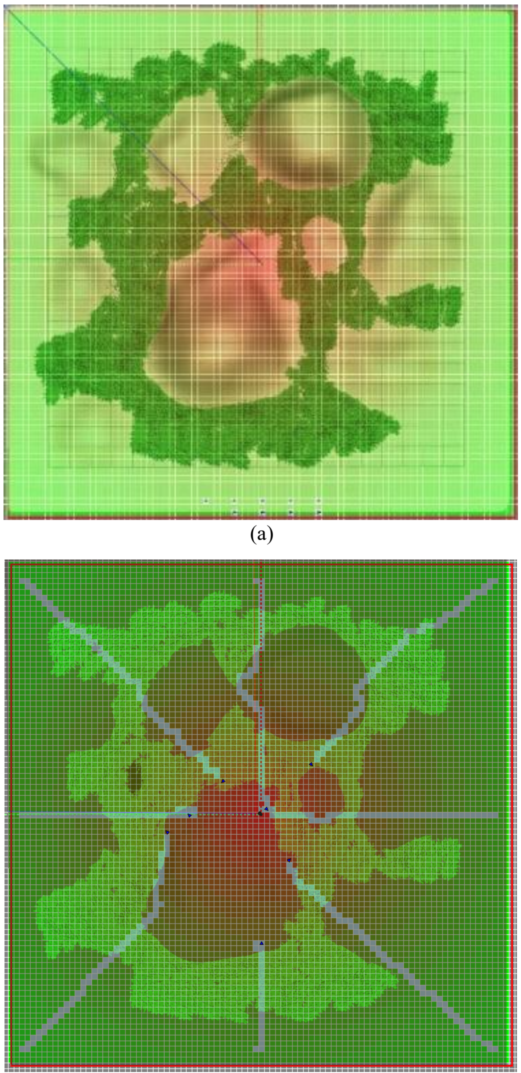


(a) Iris (b) Blue Rover (c) Rover (d) Forest fire (e) Eartquake (f) Seagrass (g) Illegal dump

Fig. 3. Vehicles (a-c) and environments (d-g) developed in the testbed

To analyse and understand the overall coordination logic, a swarm monitoring system has been developed, giving an overall bidimensional top view. Fig. 5a shows the stigmergic trial, in red colour, placed in the middle of the seagrass, acting as an attractive field, as clearly show in Fig. 5b by the robots' paths.

Fig. 4. A snapshot of a blue rover equipped with a photo camera to detect seagrass.



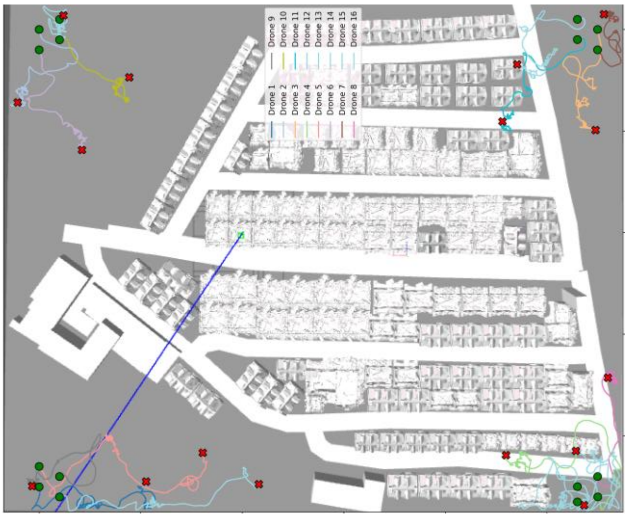
(a)

(b)

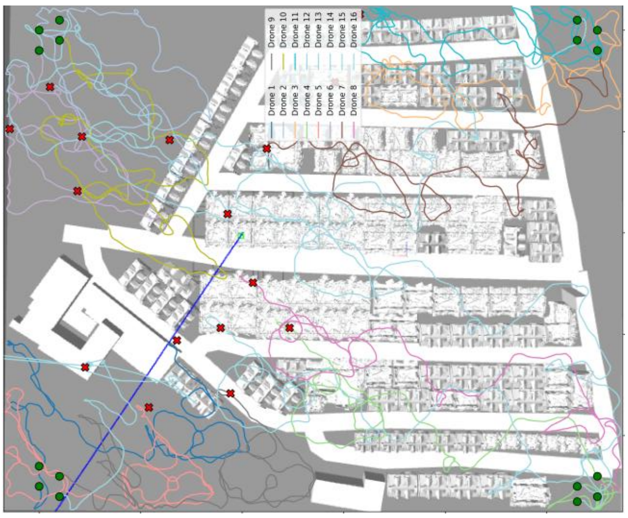
Fig. 5. A stigmergic trial acting as a positive attractive pheromone in the middle of the seagrass environment (a), with the robots's paths (b)

To analyse the emergent effect of flocking mechanisms for different parameters, the swarm monitoring system has been enriched. Fig. 6a and Fig. 6b show the path of 12 drones, split into 4 flocks, before and after the flocking parametric optimization, respectively. Specifically, in Fig. 6b separation, alignment and cohere radiuses of 10, 20 and 30 meters have been set. Here, it is apparent that the optimal parameters enabled an emergent effect with a very good exploration, in contrast to Fig. 6a characterized by poor exploration.

Fig. 7a-b show two flocking trails for a different number of robots, 8 and 16 respectively, keeping the same flocking parameters in the Seagrass scenario. Here, separation, alignment and cohere radiuses of 12, 17 and 22 meters have been set. To find the optimal values of each parameter, the percent coverage has been calculated: it is the percentage of area explored by the swarm. Specifically, Table II shows some representative cases evaluated by the optimization. For each case of Table II, Fig. 8 shows a corresponding exploration map to highlight qualitative aspects of the exploration. It is worth noting that, for a given number of robots, values of separation, alignment and cohere radiuses of 8, 13 and 18 meters achieve the best performance, highlighted in boldface style in Table II.

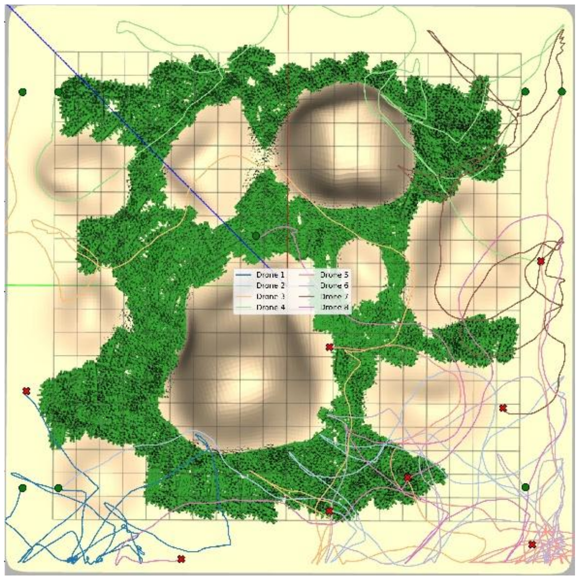


(a)

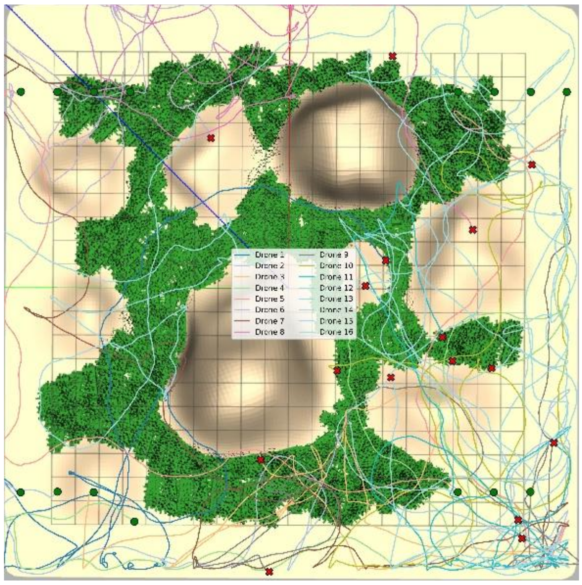


(b)

Fig. 6. A flocking trail before (a) and after (b) parametric optimization, in the Earthquale scenario.



(a)



(b)

Fig. 7. A flocking trail for a different number of robots, in the Segrass scenario

TABLE II. EXPLORATION RATE FOR DIFFERENT PARAMETERS

Fig.8	**# bots**	**sep (m)**	**align (m)**	**coh (m)**	**coverage (%)**
(a)	8	5	10	15	45.86
(b)	8	**8**	**13**	**18**	**54.95**
(c)	8	10	15	20	54.14
(d)	12	5	10	15	56.02
(e)	12	**8**	**13**	**18**	**70.59**
(f)	12	10	15	20	67.38
(g)	16	5	10	15	71.39
(h)	16	**8**	**13**	**18**	**80.48**
(i)	16	10	15	20	72.06

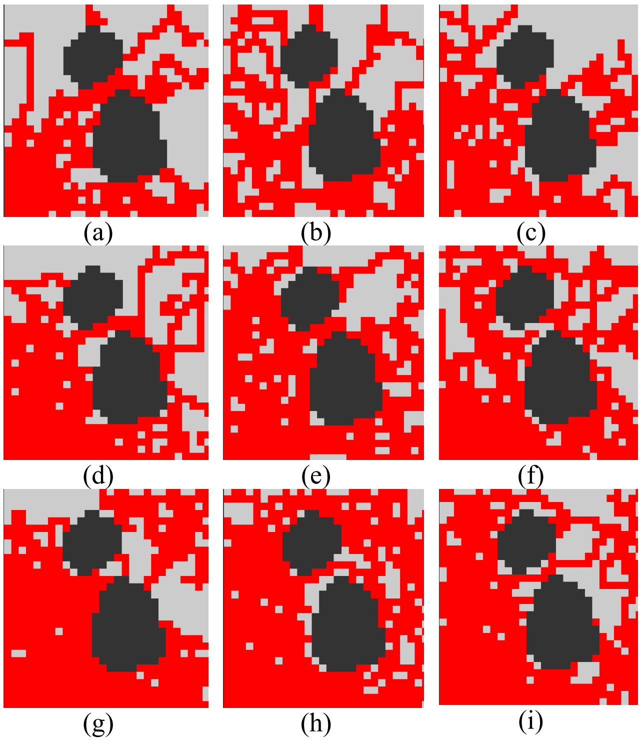
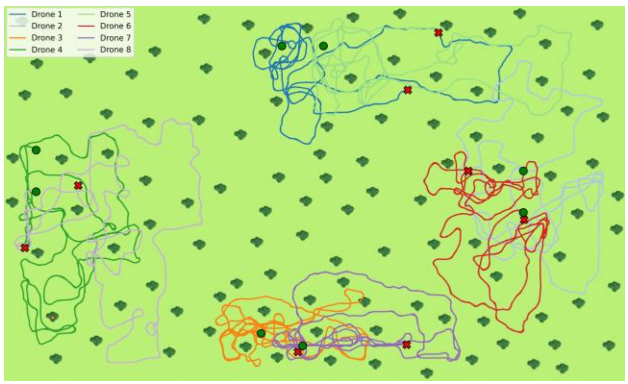
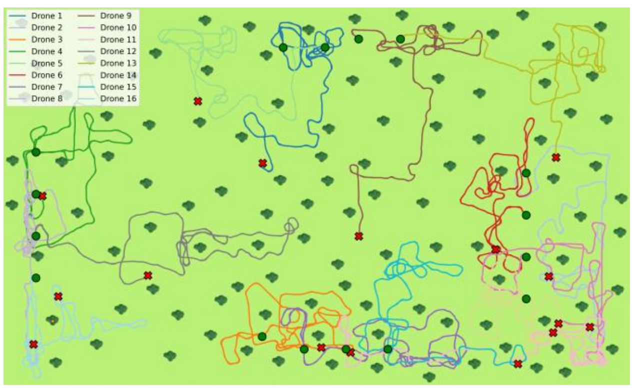


Fig. 8. Exploration maps for different parameters as detailed in Table II.

Fig. 9a-b show two flocking trails for a different number of robots, 8 and 16 respectively, with different flocking parameters in the Illegal dump scenario. Here, separation, alignment and cohere radiuses of 15, 20 and 25 meters have been set for 8 drones, and 25, 30 and 35 for 16 drones, respectively.



(a)



(b)

Fig. 9. A flocking trail for a different number of robots, in the Illegal dump scenario

To find the optimal values of each parameter, the percentage of area explored by the swarm has been calculated. Specifically, Table III shows some representative cases evaluated by the optimization. For each case of Table III, Fig. 10 shows a corresponding exploration map to highlight qualitative aspects of the exploration.

Fig. 11a-b show two flocking trails for a different number of robots, 16 and 32 respectively, with different flocking parameters in the Forest fire scenario. Here, separation, alignment and cohere radiuses of 25, 30 and 35 meters have been set for 16 drones, and 15, 20 and 25 for 32 drones, respectively. To find the optimal values of each parameter, the percentage of area explored by the swarm has been calculated. Specifically, Table IV shows some representative cases evaluated by the optimization. For each case of Table IV, Fig. 12 shows a corresponding exploration map to highlight qualitative aspects of the exploration.

TABLE III. EXPLORATION RATE FOR DIFFERENT PARAMETERS

Fig.10	**# bots**	**sep (m)**	**align (m)**	**coh (m)**	**coverage (%)**
(a)	8	10	15	20	74.49
(b)	8	**15**	**20**	**25**	**85.35**
(c)	8	20	25	30	82.60
(d)	8	25	30	35	82.83
(e)	16	10	15	20	85.91
(f)	16	15	20	25	86.48
(g)	16	20	25	30	88.20
(h)	16	**25**	**30**	**35**	**90.02**

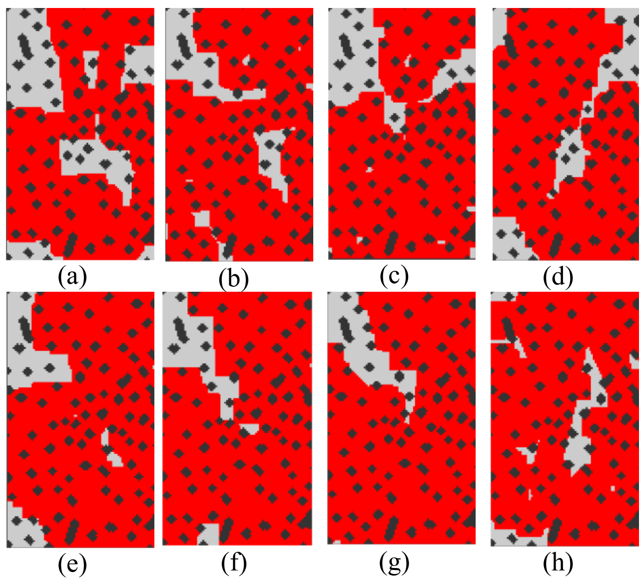


Fig. 10. Exploration maps for different parameters as detailed in TABLE III.

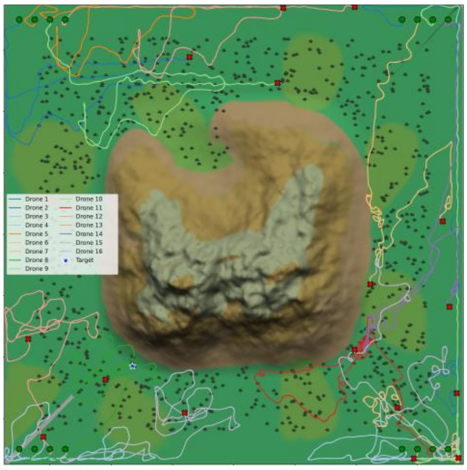
III. CONCLUSIONS

This paper introduces the main pillars of a novel approach to swarm robotics, integrating the principles of stigmergy and flocking within a cyber-physical system. By offloading complex computational tasks to a remote brain, the proposed framework enables robots to operate as thin clients, a design choice that significantly enhances scalability and adaptability. The use of a digital twin and a high-fidelity simulation environment is crucial for developing and testing the coordination logic, offering a robust platform for parametric optimization and software validation.

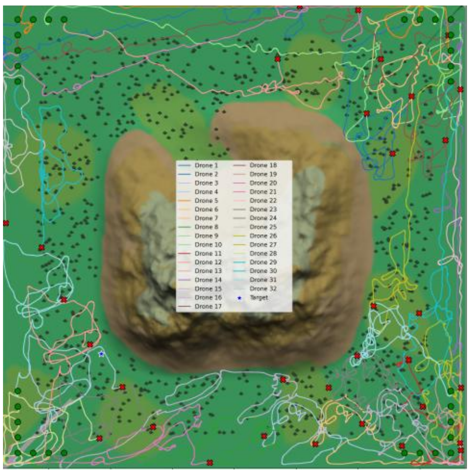
Early experimental results show the capability of this method to manage environmental uncertainty and promotes emergent, self-organized behaviours. Specifically, when optimized, the stigmergic coordination demonstrate an effective way for robots to navigate and explore using virtual pheromones, while the flocking mechanisms enable efficient area coverage. This work provides a strong foundation for future extensive research in decentralized, bio-inspired swarm robotics, particularly in applications where robots must operate autonomously in dynamic and unpredictable real-world scenarios.

IV. Acknowledgment

The authors thank Dr. Matteo Del Seppia, Dr. Federico Cavedoni, and Dr. Riccardo Fiorini for their experimental support. Work partially supported by: (i) the Italian Ministry of Education and Research (MIUR) in the framework of the FoReLab project (Departments of Excellence); (iii) the Italian Ministry of Enterprises and Made in Italy, in the framework of the "Agreements for Innovation" Projet "4DDS - 4D Drone Swarms" Ref. no. F/310097/01-04/X56.



(a)



(b)

Fig. 11. A flocking trail for a different number of robots, in the Forest fire scenario

TABLE IV. Exploration rate for different parameters

Fig.12	# bots	sep (m)	align (m)	coh (m)	coverage (%)
(a)	16	10	15	20	64.64
(b)	16	15	20	25	65.97
(c)	16	20	25	30	67.70
(d)	16	**25**	**30**	**35**	**69.94**
(e)	32	10	15	20	80.28
(f)	32	**15**	**20**	**25**	**86.00**
(g)	32	20	25	30	83.67
(h)	32	25	30	35	83.71

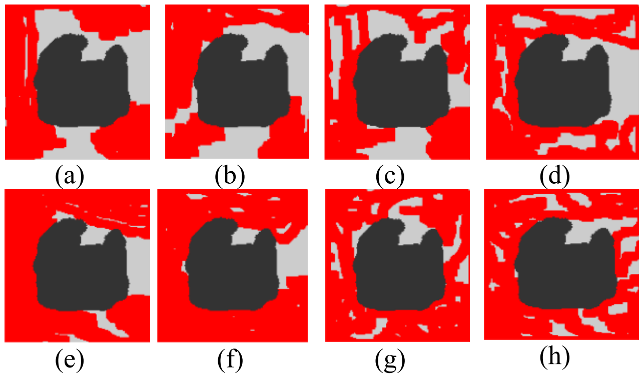


Fig. 12. Exploration maps for different parameters as detailed in Table IV.

References

[1] Hariri, R.H., Fredericks, E.M. & Bowers, K.M. Uncertainty in big data analytics: survey, opportunities, and challenges. J Big Data 6, 44 (2019). https://doi.org/10.1186/s40537-019-0206-3

[2] Bouguettaya, A., Zarzour, H., Taberkit, A. M., & Kechida, A. (2022). A review on early wildfire detection from unmanned aerial vehicles using deep learning-based computer vision algorithms. Signal Processing, 190, 108309.

[3] Youme, O., Bayet, T., Dembele, J. M., & Cambier, C. (2021). Deep learning and remote sensing: detection of dumping waste using UAV. Procedia computer science, 185, 361-369.

[4] Elma, E., Gaulton, R., Chudley, T. R., Scott, C. L., East, H. K., Westoby, H., & Fitzsimmons, C. (2024). Evaluating UAV-based multispectral imagery for mapping an intertidal seagrass environment. Aquatic Conservation: Marine and Freshwater Ecosystems, 34(8), e4230.

[5] Bayirhan, İ., & Gazioğlu, C. (2020). Use of Unmanned Aerial Vehicles (UAV) and Marine Environment Simulator in Oil Pollution Investigations. Baltic Journal of Modern Computing, 8(2).

[6] Cimino, M. G., Minici, D., Monaco, M., Petrocchi, S., & Vaglini, G. (2021). A hyper-heuristic methodology for coordinating swarms of robots in target search. Computers and Electrical Engineering, 95, 107420.

[7] Sheta, A. F., Ghatasheh, N., Faris, H., & Rodan, A. (2019). Robotics Evolution: from Remote Brain to Cloud. arXiv preprint arXiv:1901.03297.

[8] Avvenuti, M., Cesarini, D., & Cimino, M. G. (2013). Mars, a multi-agent system for assessing rowers' coordination via motion-based stigmergy. Sensors, 13(9), 12218-12243.