



# Random Walk Exploration for Swarm Mapping

Miquel Kegeleirs<sup>(✉)</sup> , David Garzón Ramos<sup>(✉)</sup> , and Mauro Birattari<sup>(✉)</sup> 

IRIDIA, Université Libre de Bruxelles, Brussels, Belgium  
{mkegelei,dgarzonr,m biro}@ulb.ac.be

**Abstract.** Research in swarm robotics has shown that robot swarms are effective in the exploration of unknown environments. However, little work has been devoted to port the exploration capabilities of robot swarms into the context of mapping. Indeed, conceiving robot swarms that can map an unknown environment in a robust, scalable, and flexible way is an open issue. In this paper, we investigate a swarm mapping method in which robots first individually map the environment by random walk and then, we merge their maps into a single, global one. We focus on five variants of random walk and we compare the quality of the maps that a swarm produces when exploring the environment using these variants. Our experiments with ten e-puck robots show that, despite the individual maps being incomplete by themselves, it is possible to collectively map the environment by merging them. We found that the quality of the map depends on the exploration behavior of the individuals. Our results suggest that one of the variants of random walk, the ballistic motion, gives better mapping results for closed environments.

**Keywords:** Swarm mapping · Exploration · Random walk

## 1 Introduction

A robot swarm can collectively accomplish tasks that an individual robot could not accomplish alone. By its own nature, a robot swarm is a self-organizing system that operates autonomously without relying on a leader robot or on external infrastructure. In addition, it possesses desirable properties such as scalability, flexibility, and fault-tolerance [4] due to redundancy and locality of sensing and communication. Because of these properties, robot swarms are ideal candidates to perform missions that require to explore and map unknown environments in which the risk that individual robots fail or are lost is high. Yet, no well-defined methodology exists for swarm mapping (i.e. for exploring and mapping with robot swarms).

---

All experiments were performed by MK and DGR. The article was drafted by MK and DGR and revised by the three authors. The research was directed by MB.

The original version of this chapter was revised: The given name and the family name of the second author has been corrected. The correction to this chapter is available at [https://doi.org/10.1007/978-3-030-25332-5\\_50](https://doi.org/10.1007/978-3-030-25332-5_50)

© Springer Nature Switzerland AG 2019  
K. Althoefer et al. (Eds.): TAROS 2019, LNAI 11650, pp. 211–222, 2019.  
[https://doi.org/10.1007/978-3-030-25332-5\\_19](https://doi.org/10.1007/978-3-030-25332-5_19)

Mapping has been largely explored [23] in the last decades and it is usually the first task that a robot performs when it operates in an unknown environment. Most of the mapping methods have been designed to be general, platform independent, and use-case independent. However, they have been developed mostly for single robots and they cannot be directly adopted by centralized multi-robot systems and robot swarms.

State-of-the-art methods for mapping often conflict with some characteristics of robot swarms such as locality and the absence of global knowledge [4]. First, these methods usually require external infrastructure to ensure inter-robot communication or localization (a single point of failure that hinders fault tolerance). Second, either one or a small number of robots is used, and they are expensive and heavily equipped. This condition implies that the loss of a single robot seriously affects the whole system. As a result, adapting the research on mapping to swarm robotics is not straightforward and little attention has been devoted to close the gap between the two fields.

Some important questions still need to be addressed before effective swarm mapping can be achieved: How should the swarm explore and gather information on the environment? How should the robots share and spread the information gathered? How should the information be retrieved and used to produce maps? Our work aims to shed further light on the first of these questions. More precisely, we investigate the possibility of using random walk as a strategy to explore and gather information on the environment.

We are particularly interested in using random walk exploration in swarm mapping because random walk is a simple behavior that can be easily implemented in a robot swarm. Indeed, by its own nature, random walk is flexible, scalable and robust as it does not rely on localization or communication. For these reasons, though many different exploration strategies have been proposed for single robot and centralized multi-robot systems [9,18], random walk is still the most commonly adopted behavior for exploring with robot swarms [3,5,8].

In this paper, we evaluate five different variants of random walk on swarm mapping: Brownian motion, correlated random walk, Lévy walk, Lévy taxis, and ballistic motion. In our experiments, each robot individually maps the environment, driven by a random walk. Later on, the individual maps are merged to produce a single, global map of the environment. We assess the ability of the robot swarm to map the environment through the quality of the global map produced. The main original contribution we make in the paper is the evaluation of different variants of random walk in the context of swarm mapping. Moreover, to the best of our knowledge, this is the first paper in which random walk exploration, the *GMapping* algorithm [11,12], and the *multirobot\_map\_merge* algorithm [13] are used together to achieve swarm mapping.

The paper is structured as follows. In Sect. 2, we discuss related work in exploration and mapping with multi-robot systems. In Sect. 3, we present the swarm mapping method that we investigate in the paper. In Sect. 4, we describe the experimental setup. In Sect. 5, we illustrate the results of the experiments. In Sect. 6, we conclude the paper and we sketch future research.

## 2 Related Work

Traditionally, mapping with multi-robot systems has been addressed separately in the form of two sub-problems: multi-robot simultaneous localization and mapping (multi-robot SLAM) and multi-robot exploration [22].

Multi-robot SLAM concerns the collective production of maps. The work of Saeedi et al. [23] provides a review of the methods that have been proposed on this topic. The review describes advantages, problems, and challenges of widely used methods based on the Extended Kalman Filter (EKF-SLAM), particle filters (PF-SLAM), and map merging, among others. Though current SLAM methods could be implemented in robot swarms, they would introduce constraints that would affect the flexibility of the system: centralized mechanisms or complex inter-robot interactions. The issues that one can encounter by adopting SLAM on robot swarms are described by Barca et al. [1].

The literature on multi-robot mapping often assumes an existent multi-robot exploration and overlooks the relationship between exploration and SLAM. Rone et al. [22] described different exploration methods used in multi-robot mapping and their characteristics. The review presents some exploration methods that could be easily implemented in robot swarms such as potential field exploration, greedy mapping, and diffusion mapping, the last two relying on random walk behaviors.

Recent work on swarm robotics has brought attention to the impact of using different variants of random walk. For example, Dimidov et al. [3] studied the performances of different variants of random walk while exploring and searching for a static target in the environment. Schroeder et al. [24] conducted a similar research and compared different variants of random walk to cover the environment. The results of these studies highlight the relationship between the performance on the task and the ability of the swarm to explore the environment with a particular variant. Our hypothesis is that the same differences in performance should appear while performing swarm mapping with different variants of random walk. To corroborate our hypothesis, we investigate the mapping performance of a robot swarm that explores by using five different variants.

Mapping with a robot swarm has already been reported by Ramachandran [20], who evaluates the efficiency of the so-called Informed Correlated Lévy Walk (i.e. a variant of random walk). However, our research differs in both the method and objectives. First, Ramachandran proposed a method that computes the global maps online in which robots know their absolute position and communicate thanks to external infrastructure. In our method, the global maps are produced offline at the end of the experiment, the robots estimate their own position, and they do not need to communicate while exploring and mapping the environment. Second, in the experiments of Ramachandran, the code was modified when ported from simulation to the real robots. In our case, we use the same code in both. Finally, Ramachandran only evaluated the performances of Lévy walk and Informed Correlated Lévy walk. As mentioned before, we evaluate four more variants in addition to Lévy walk.

### 3 Swarm Mapping Method

The swarm mapping method we propose integrates three algorithms: random walk exploration, *GMapping* [11, 12], and *multirobot\_map\_merge* [13]. First, random walk exploration defines the behavior that the robots use for exploring the environment. Then, each robot uses the *GMapping* algorithm to gather information and produce an individual map of the area it has explored. Finally, *multirobot\_map\_merge* is used to merge the individual maps and to produce a global map of the environment.

We developed the swarm mapping method for a swarm of e-puck [15] robots. The random walk behaviors are implementations based on models previously proposed in the literature. *GMapping* and *multirobot\_map\_merge* are default implementations that we adopted from the Robot Operating System (ROS) [19]. In the following, we provide a brief description of the algorithms and their contribution to the swarm mapping method.

#### 3.1 Random Walk Exploration

The first component of the swarm mapping method concerns the behavior that drives the robots to explore the environment.

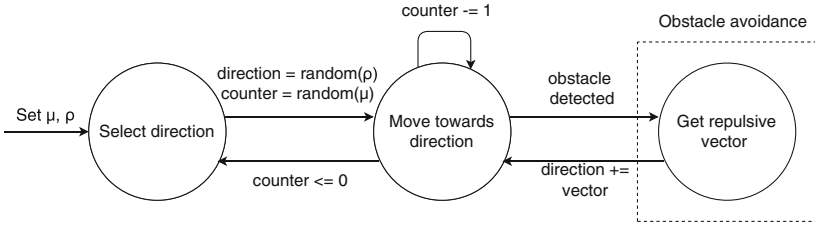
In our study, we consider five variants of random walk: Brownian motion [6], correlated random walk [21], Lévy walk [25], Lévy taxis [16] and ballistic motion. We selected these variants because they have been reported as completely random, specialized in *intensification* (i.e., intensive exploration of a small area of the environment), and/or specialized in *coverage* (i.e., moderate exploration of a large area of the environment). Indeed, we are interested in studying the differences between intensification-oriented and coverage-oriented exploration in swarm mapping.

Despite their different exploration abilities, Brownian motion, correlated random walk, Lévy walk, and Lévy taxis share the same mathematical model [3, 16]. They can be described within a general behavior: the robot selects a direction at random and moves towards it until a new direction is selected. The characteristics of the movement for each variant are defined by the parametrization of the model. In particular, the aforementioned variants differ in the  $\rho$  and  $\mu$  parameters: the former associated to the turning angle when the robot selects a direction at random, and the later associated to the time the robot keeps moving along that direction. In our implementations, we used the parametrization described as optimal by Pasternak et al. [16] and Dimidov et al. [3] as shown in Table 1. Figure 1a describes the general behavior of Brownian motion, correlated random walk, Lévy walk and Lévy taxis.

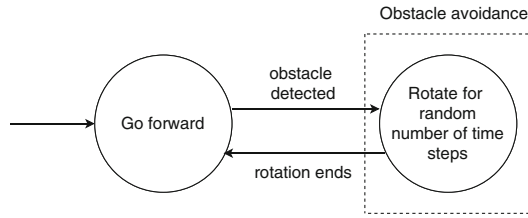
In addition to the four variants described above, we included the ballistic motion behavior used by Francesca et al. in AutoMoDe-Vanilla [8]. The ballistic motion used in the exploration module of Vanilla drives the robots with a constant straight motion, only changing their direction when an obstacle is detected. We consider here the ballistic motion as a random walk behavior because the robots select the new direction at random. Figure 1b describes the general behavior of the ballistic motion.

**Table 1.** Step length, turning angle, and specialty for the variants of random walk

Variant	Step length	Turning angle	Specialty
Brownian motion	$\mu = 3$	$\rho = 0$	None
Correlated random walk	$\mu = 3$	$\rho = 0.05$	Intensification
Lévy walk	$\mu = 2.8$	$\rho = 0$	Coverage
Lévy taxis	$\mu = 2.8$	$\rho = 0.05$	Mix of coverage/intensification
Ballistic motion	N/A	N/A	Coverage



(a) Brownian motion, correlated random walk, Lévy walk and Lévy taxis



(b) Ballistic motion

**Fig. 1.** Finite state machines of the variants of random walk

Figures 1a, b also show that the five variants of random walk include a basic obstacle avoidance behavior. Brownian motion, correlated random walk, Lévy walk, and Lévy taxis integrate a repulsive force model [2] that drives the robots away from any object in their detection range. In the case of the ballistic motion, the obstacle avoidance is an intrinsic property of the behavior and it allows the robots to avoid objects in the same range as the others.

### 3.2 Individual Mapping

The second component of the swarm mapping method concerns the ability of the robots to gather information and to individually map different areas of the environment.

We use the *GMapping* algorithm to produce an individual map for each robot in the swarm. *GMapping* is a single-robot, SLAM algorithm that takes sensor information and produces a two-dimensional occupancy grid of the environment.

In our method, we consider a model of robot that explores, computes its own odometry (i.e. an estimation of its own displacement), and detects and locates objects in a short range. This information is sufficient for *GMapping* to produce a map that describes the empty and obstructed areas that the robot finds.

The odometry on each robot is first initialized with the deployment position of the robot. Then, the robot continuously estimates its position in open loop by integrating the movement commanded by the random walk behavior. Providing the robots with knowledge about their starting position does not have any effect on their ability to produce the individual maps. The information was included in order to enable the merging of the individual maps in the following step of the swarm mapping method.

### 3.3 Global Mapping

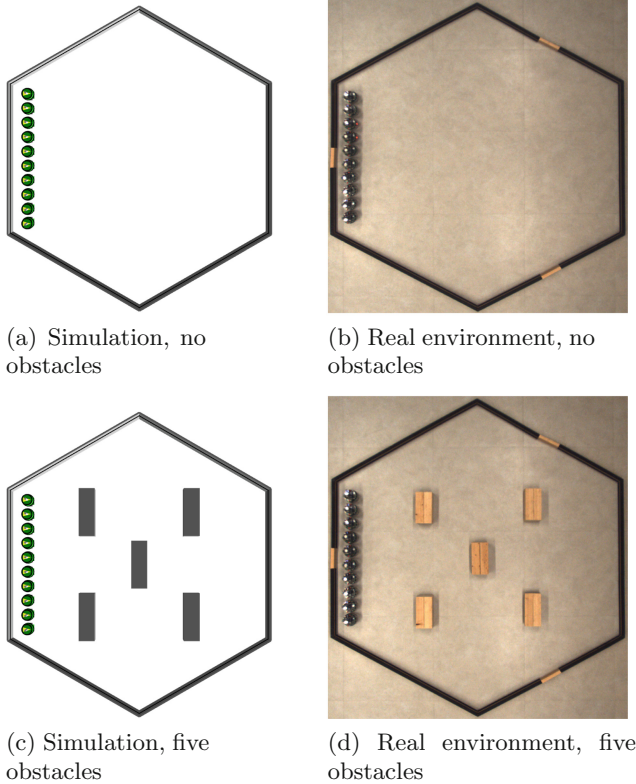
The third component of the swarm mapping method concerns the combination of the individual contribution of each robot in the swarm into a single global map of the environment.

In our swarm mapping method, we merge the individual maps produced by each robot into a single map by using the *multirobot\_map\_merge* algorithm. *Multirobot\_map\_merge* is originally intended for merging an arbitrary number of individual maps at run time. The maps are merged only when the robots have finished the exploration of the environment. We chose *multirobot\_map\_merge* because it has been used in previous research [13] to merge individual maps produced by multi-robot systems and *GMapping*.

*Multirobot\_map\_merge* has two merging modes that differ on whether the initial position of the robots is known or not, the former being more robust. On the one hand, providing the algorithm with no positioning information is a closer approach to the self-organizing nature of the robot swarms. On the other hand, we consider reasonable that the initial relative position of the robots could be known *a priori* when the robots are deployed in a fixed location. We consider both cases in our swarm mapping method.

## 4 Experimental Setup

We evaluate the swarm mapping method with a swarm of e-pucks mapping a closed environment. The robots operate in an hexagonal closed environment referred to as the arena. The time available to the robots for mapping the arena is 180s. The arena comprises an area of 2.30 m<sup>2</sup> and it is surrounded by walls of 0.94 m in length. On a per-experiment basis, the arena could contain none or five rectangular obstacles of 0.02 m<sup>2</sup>. We expect that differences in the object density in the arena should have an effect on the ability of the swarm to explore the environment, and therefore, in the quality of the global map. The deployment position of the robots in the arena is fixed in all the experiments: the robots are aligned along the west wall of the arena with 0.10 m between their center and the wall, and 0.09 m between each other's center.



**Fig. 2.** Arenas used in the experiments and starting position of the robots

Figure 2 shows the arenas and robots in their deployment position for both the simulated and real environments. All the simulations are performed using ARGoS3, beta 48 [17]. In the following, we describe the reference model of the robots and the protocol we used in our experiments.

#### 4.1 Robot

We consider an extended version of the e-puck [10, 15]. The e-puck is a differential wheeled mobile robot with a 0.08 m diameter and maximum velocity of  $0.12 \text{ m s}^{-1}$ . It is equipped with an embedded computer and a set of eight infrared proximity sensors. The proximity sensors are distributed around the robot and they detect objects in a range of 0.10 m. We configured the robots to avoid objects in a range of 0.08 m to let them map obstacles before avoiding them.

For the purpose of this study, we integrated ROS Indigo into the embedded computer of the robot. We developed a ROS-based controller that drives the robot by using the variants of random walk described in Sect. 3.1. The controller

transforms the desired direction of movement into appropriate velocity commands for the e-puck. In parallel, the controller also computes the odometry of the robot and receives the readings from the proximity sensors. This information is passed to *GMapping* at runtime to enable the robot mapping.

## 4.2 Protocol

We evaluate the five variants of random walk along with our swarm mapping method, both in simulation and real environment.

We consider a swarm of 10 e-puck robots mapping the arena described in Sect. 4. In all experiments, we execute our method for swarm mapping to produce one global map of the arena. In simulation, the swarm mapping is executed 30 times for each variant. In the real robot experiments, the swarm mapping is executed 10 times for each variant. We repeated the experiments in the two arena configurations: with and without obstacles. We report the individual and global maps obtained in a per-experiment basis.

We assess the quality of the global maps by visual inspection. For the purpose of our experiments, the quality metric represents the completeness and representativeness of the maps when compared to a reference map. In this case, the reference map is the ideal map to be produced if the robots map perfectly the environment. We consider visual inspection a sufficient metric to qualitatively assess the exploration abilities of the robots in swarm mapping. A well-established methodology for quantitative assessment is still missing and should be the objective for future work.

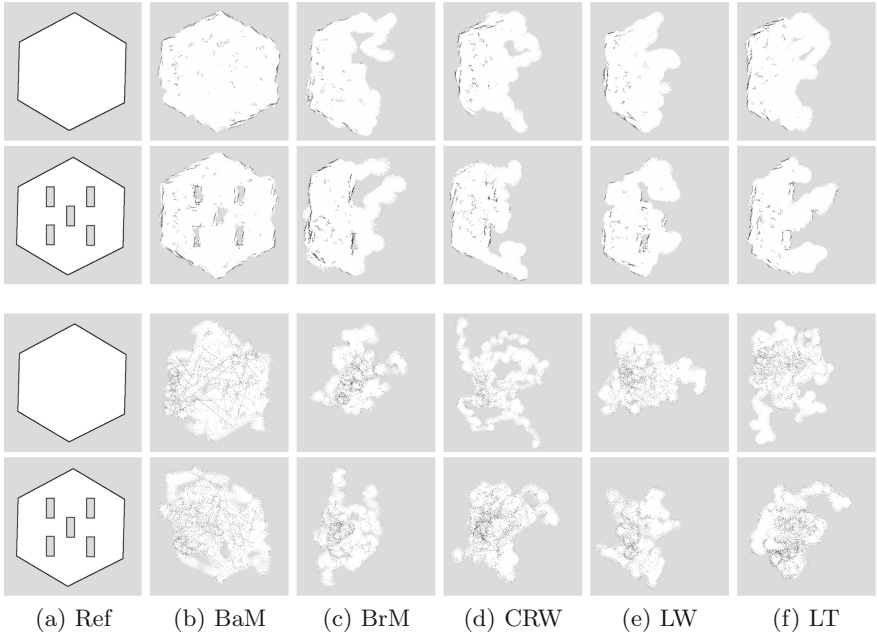
## 5 Results

We present the qualitative analysis of the results for both simulated and real experiments. In addition, we communicate the experiences made and the insights gained while adopting *GMapping* and *multirobot\_map\_merge* in swarm mapping. Individual and global maps, demonstrative videos, code, and ROS parameter files are available in [14]. Figure 3 shows a sample of the maps produced in swarm mapping for each variant of random walk.

With each variant, the robot swarm explores similarly the arena both in simulation and real environment. The main difference between simulation and real environment lies on the sensitivity of the proximity sensors: in real environment, robots are more sensitive and tend to detect and avoid obstacles from a further distance. As a result, then tend to cover more the environment. The maps created in simulation match the areas of the arena explored by the swarm, and provide usable information about its content. The maps obtained with real e-pucks, on the contrary, do not reflect the explored areas and are hardly useful. We compared the performance of the variants on the basis of the results obtained in simulation.

The results show that the ballistic motion provides maps with a better quality than the other variants. Indeed, robots using the ballistic motion tend to cover better the arena while the other variants focus more on intensification.





**Fig. 3.** Swarm mapping results in simulation (above) and real robot experiments (below): (a) Reference map; (b) Ballistic motion; (c) Brownian motion; (d) Correlated random walk; (e) Lévy walk; (f) Lévy taxis.

The ballistic motion is able to provide (nearly) complete maps while the maps produced with the other variants are mostly half-complete. In both cases, the exploration behavior was not affected by the presence of obstacles in the arena. We acknowledge that these observations are however only valid in the context of closed space experiments. It is easy to convince oneself that the ballistic motion cannot work in an open space without a high risk of losing robots.

We also found that Brownian motion, correlated random walk, Lévy walk and Lévy taxis performed similarly. In overall, the maps produced with these variants do not differ neither in completeness or representativeness. As a matter of fact, we did not find substantial differences in the coverage and intensification abilities of these behaviors. The four variants responds to the same mathematical model and hence, their behavior does not differ considerably. In this sense, we argue that the parametrization of the model considered as optimal in previous research [3] cannot be generalized and it is not optimal for the e-pucks.

We successfully integrated *GMapping* and *multirobot\_map\_merge* in swarm mapping. The results in simulation show that, despite their limited resources, a swarm of e-pucks performing ballistic motion can map its environment. Still, we think that porting algorithms from single robot and centralized multi-robot systems is not necessarily the best approach to achieve swarm mapping.

First, *GMapping* was designed to work with robots equipped with dense long-range sensors and good localization systems; swarm robots like the e-pucks provide neither of them. Although we succeeded in obtaining the maps in simulation, *GMapping* failed when ported to the real robots. Real e-pucks are prone to have more noisy proximity readings and considerable errors in the estimation of the odometry. Consequently, *GMapping* produced noisy and erroneous individual maps that eventually could not be merged in good quality global maps.

Second, *multirobot\_map\_merge* was designed to merge maps regardless how they were produced. Previous experiments with this algorithm were performed with individual maps with a higher number of features and few overlapping areas [13]. In our swarm mapping method, on the contrary, the swarm succeeded in mapping due to the redundancy of small contributions from each individual map. Each robot in the swarm only maps a small area of the environment and the random walk exploration leads to increase the overlapping in those areas. As a consequence, the *multirobot\_map\_merge* algorithm was not able to produce the global map without the deployment position of the robots. Moreover, the ROS implementation of the algorithm crashed in some experiments disregarding the exploration behavior or the environment (real or simulated). Our results show that *multirobot\_map\_merge* could be used to some extent in swarm mapping, however a more suitable option should be explored in future work.

## 6 Conclusions

Robot swarms are suitable for the exploration and mapping of unknown environments. Still, swarm mapping is a field under development and no well-defined methodology exists to achieve it. In this article, we investigated the possibility of using random walk exploration with *GMapping* and *multirobot\_map\_merge* as a method to achieve swarm mapping. The robots explored and individually mapped the environment by random walk, and then the individual maps were merged to produce one single, global map. The redundancy of the individual maps contributes to obtain a good quality representation of the environment. This method can be ported with minimal effort to other ROS-based robot swarms since it does not require complex interactions between robots.

We conducted experiments, both in simulation and real environment, with a robot swarm mapping a closed environment while using five variants of random walk: Brownian motion, correlated random walk, Lévy walk, Lévy taxis, and ballistic motion. The robot swarm was able to successfully map the environment in simulation. However, experiments in real environment were less satisfying and highlighted the necessity to find strategies to better transfer the control software from simulation to real environment. Results in simulation showed that ballistic motion produces the best maps due to the better ability of the swarm to cover the environment. Yet, these results are only valid for closed environments and could not be extended to open ones. The other variants, on the contrary, showed a dominant intensification behavior that provided mostly half-complete maps.

We conclude that the selection and parametrization of the appropriate random walk behavior for swarm mapping is a topic that requires further research.

Future work will be devoted to use modular automatic design methods like AutoMoDe [8] for this purpose. We expect a twofold contribution from AutoMoDe: first, it would provide a framework to assess the selection and parametrization of the behaviors in different environments; second, it would allow to port better the control software from simulation to the real robots. With regards to the mapping method, we will investigate the possibility of adopting concepts of distributed mapping [7] into swarm mapping. Indeed, merging individual maps centralizes the mapping process and, to some extent, affects the flexibility of the system. We expect that distributed mapping will provide alternatives to produce and retrieve useful partial maps in a fully distributed way.

**Acknowledgements.** The project has received funding from the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (grant agreement No 681872). Mauro Birattari acknowledges support from the Belgian Fonds de la Recherche Scientifique – FNRS. David Garzón Ramos acknowledges support from the Colombian Administrative Department of Science, Technology and Innovation – COLCIENCIAS.

## References

1. Barca, J.C., Sekercioglu, Y.A.: Swarm robotics reviewed. *Robotica* **31**(3), 345–359 (2013)
2. Borenstein, J., Koren, Y.: Real-time obstacle avoidance for fast mobile robots. *IEEE Trans. Syst. Man Cybern.* **19**(5), 1179–1187 (1989)
3. Dimidov, C., Oriolo, G., Trianni, V.: Random walks in swarm robotics: an experiment with kilobots. In: Dorigo, M., et al. (eds.) ANTS 2016. LNCS, vol. 9882, pp. 185–196. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-44427-7\\_16](https://doi.org/10.1007/978-3-319-44427-7_16)
4. Dorigo, M., Birattari, M., Brambilla, M.: Swarm robotics. *Scholarpedia* **9**(1), 1463 (2014)
5. Dorigo, M., et al.: a novel concept for the study of heterogeneous robotic swarms. *IEEE Robot. Autom. Mag.* **20**(4), 60–71 (2013)
6. Feynman, R.P., Leighton, R.B., Sands, M.L.: *The Feynman Lectures on Physics Volume 1: Mainly Mechanics, Radiation, and Heat*. Basic Books, New York (2011)
7. Fox, D., Ko, J., Konolige, K., Limketkai, B., Schulz, D., Stewart, B.: Distributed multirobot exploration and mapping. *Proc. IEEE* **94**(7), 1325–1339 (2006)
8. Francesca, G., Brambilla, M., Brutschy, A., Trianni, V., Birattari, M.: AutoMoDe: a novel approach to the automatic design of control software for robot swarms. *Swarm Intell.* **8**(2), 89–112 (2014)
9. Galceran, E., Carreras, M.: A survey on coverage path planning for robotics. *Rob. Auton. Syst.* **61**(12), 1258–1276 (2013)
10. Garattoni, L., Francesca, G., Brutschy, A., Pinciroli, C., Birattari, M.: Software infrastructure for e-puck (and TAM). Technical report. TR/IRIDIA/2015-004, IRIDIA, Université libre de Bruxelles, Belgium (2015)
11. Grisetti, G., Stachniss, C., Burgard, W.: Improving grid-based slam with rao-blackwellized particle filters by adaptive proposals and selective resampling. In: *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*. pp. 2432–2437. IEEE Press (2005)
12. Grisetti, G., Stachniss, C., Burgard, W.: Improved techniques for grid mapping with rao-blackwellized particle filters. *IEEE Trans. Rob.* **23**(1), 34–46 (2007)

13. Hörner, J.: Map-merging for multi-robot system. Master's thesis, Univerzita Karlova (2016)
14. Kegeleirs, M., Garzón Ramos, D., Birattari, M.: Random walk exploration for swarm mapping: supplementary material (2019). <http://iridia.ulb.ac.be/supp/IridiaSupp2019-003/index.html>
15. Mondada, F., et al.: The e-puck, a robot designed for education in engineering. In: Gonçalves, P., Torres, P., Alves, C. (eds.) Proceedings of the 9th Conference on Autonomous Robot Systems and Competitions. Instituto Politécnico de Castelo Branco, pp. 59–65, Portugal (2009)
16. Pasternak, Z., Bartumeus, F., Grasso, F.W.: Lévy-taxis: a novel search strategy for finding odor plumes in turbulent flow-dominated environments. *J. Phys. A Math. Theoret.* **42**(43), 434010 (2009)
17. Pinciroli, C., et al.: ARGoS: a modular, parallel, multi-engine simulator for multi-robot systems. *Swarm Intell.* **6**(4), 271–295 (2012)
18. Portugal, D., Rocha, R.: A survey on multi-robot patrolling algorithms. In: Camarinha-Matos, L.M. (ed.) DoCEIS 2011. IAICT, vol. 349, pp. 139–146. Springer, Heidelberg (2011). [https://doi.org/10.1007/978-3-642-19170-1\\_15](https://doi.org/10.1007/978-3-642-19170-1_15)
19. Quigley, M., et al.: ROS: an open-source robot operating system. In: ICRA Workshop on Open Source Software. vol. 3, p. 5. IEEE Press (2009)
20. Ramachandran, R.K.: Exploration, mapping and scalar field estimation using a swarm of resource-constrained robots. Ph.D. thesis, Arizona State University (2018)
21. Renshaw, E., Henderson, R.: The correlated random walk. *J. Appl. Probab.* **18**(02), 403–414 (1981)
22. Rone, W., Ben-Tzvi, P.: Mapping, localization and motion planning in mobile multi-robotic systems. *Robotica* **31**(1), 1–23 (2013)
23. Saeedi, S., Trentini, M., Seto, M., Li, H.: Multiple-robot simultaneous localization and mapping: a review. *J. Field Rob.* **33**(1), 3–46 (2016)
24. Schroeder, A., Ramakrishnan, S., Kumar, M., Trease, B.: Efficient spatial coverage by a robot swarm based on an ant foraging model and the lévy distribution. *Swarm Intell.* **11**(1), 39–69 (2017)
25. Zaburdaev, V., Denisov, S., Klafter, J.: Lévy walks. *Rev. Mod. Phys.* **87**(2), 483–530 (2015)