Adaptive Strategies for Team Formation in Minimalist Robot Swarms

Luigi Feola and Vito Trianni

Abstract—**The formation of collaborative robotic teams for task execution often requires coordination in both space and time, with robots gathering in close vicinity and concurrently executing a task. For decentralised robotic swarms constituted of minimalist agents unable to communicate and plan ahead, a probabilistic approach might ensure that tasks are executed at the maximum possible rate by means of opportunistic team formation. We consider here the case of strictly-collaborative tasks of two types—easy and hard—each requiring a specific number of agents to concurrently work in the same area. We show how task execution can be improved by adaptive behavioural strategies that (i) change the random movements of robots to bias their distribution towards areas where hard tasks are present, and (ii) specialise the robots' behaviour to facilitate the formation of teams tailored to the one or the other task type. Experiments with simulated and real swarms of Kilobots demonstrate the suitability of the proposed approach, opening to future applications in micro/nano-robotics.**

Index Terms—**Adaptive random walk, swarm robotics, team formation.**

I. Introduction

SWARM robotics advocates the usage of large numbers of relatively simple robots to solve complex problems, exploiting self-organisation as a guiding principle [1]. Far-reaching perspectives consider robots that are minimalist in their sensing, communication and computation, but are deployed in thousands to collaborate towards the accomplishment of tasks distributed in space and time. Think for instance to micro- and nano-robots for precision medicine [2], [3], which need to diffuse and disperse within the body carrying drugs [4] or stem cells [5] to be delivered when the right conditions are met (e.g., the presence of diseased cells). Similarly, robots deployed underwater [6] or in space [7] may not rely on high-end computation and communication, requiring minimal complexity of the individual behaviour while guaranteeing task execution. Generally speaking, future robot swarms might face harsh operating conditions where no communication is possible and no external infrastructure is available [8]. These robots might not be able to efficiently communicate with a central controller or with each other, and might not have the perceptual and computational abilities to self-localise or precisely plan their movements. It is therefore necessary to study collaborative strategies that do not rely on complex control and interaction rules.

Approaches have been developed that require little or no computation at all to achieve swarm-level coordination [9], [10]. In such approaches, direct mapping between sensory input and motor actions—suitably tuned by some optimisation algorithm—proves sufficient to obtain efficient and scalable behaviour, opening the way for implementation in micro-robots deprived of arithmetic logic units. Research in active colloids has also demonstrated self-organisation at the microscopic scale as a result of minimally complex behaviour and interactions [11], [12]. In such systems, stochastic processes determine collective dynamics such as the formation of ordered clusters. Importantly, it has been shown that active colloids can present different motion patterns in response to external signals or positions and orientations of neighbours [13]–[15]. This opens the way to control group behaviours at the nano-scale even when the single units cannot rely on complex mechatronics components. In these and similar minimalist contexts, collaboration within teams can be key, for instance to reach a critical mass that makes task execution effective (e.g., coordinately unloading a sufficient amount of drugs at diseased cells). However, the formation of teams for task execution cannot rest on more or less complex deliberation and recruiting strategies [16]. In similar contexts, teams must be formed opportunistically as a result of suitable stochastic transition rules [17], [18].

Tasks that require a sufficiently large team to ensure execution are referred to as *strictly collaborative tasks*. Optimal execution by simple non-communicating robots was first demonstrated by means of the so called "stick-pulling experiment" [19]. In the original study, two robots needed to cooperate to extract from the ground a long stick that a single robot could not effectively manipulate alone. In a more general formulation, two or more robots must concurrently work at a "collaboration site" (possibly undertaking complementary roles) [20], [21]. Coordination is strictly necessary for task execution, and team formation becomes particularly difficult when there are fewer robots than tasks. A minimalist solution suitable for robot swarms entails a random search for tasks and a fixed dwelling time at the

Manuscript received September 9, 2021; accepted January 22, 2022. Date of publication February 11, 2022; date of current version February 22, 2022. This letter was recommended for publication by Associate Editor S. Hauert and Editor M. Ani Hsieh upon evaluation of the reviewer's comments. This work was supported by the Office of Naval Research Global (ONRG) through the project Collective Decisions in Dynamic Environments (CODE) under Award N62909-18-1-2093. *(Corresponding author: Luigi Feola.)*

Luigi Feola is with the Department of Computer, Control, and Management Engineering Antonio Ruberti, Sapienza, University of Rome, 00185 Rome, Italy, and also with the Institute of Cognitive Sciences and Technologies (ISTC) of the National Research Council (CNR), University of Rome, 00185 Rome, Italy (e-mail: feola@diag.uniroma1.it).

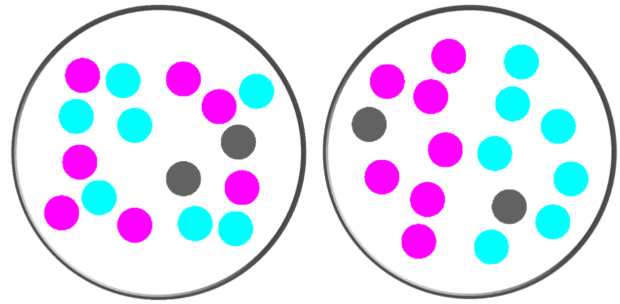
Vito Trianni is with the Institute of Cognitive Sciences and Technologies (ISTC) of the National Research Council (CNR), University of Rome, 00185 Rome, Italy (e-mail: vito.trianni@istc.cnr.it).

Digital Object Identifier 10.1109/LRA.2022.3150479

collaboration site, waiting for other robots to join. The dwelling time is the only control parameter to be optimised: it must be long enough to grant other robots sufficient time to reach a collaboration site, but must also be short to maximise the rate of successful collaborations. The optimal dwelling time is a function of the robot density and the number of tasks to execute: the higher the ratio between robots and tasks, the higher the expectation of suitable team formation, and the smaller the optimal dwelling time [19], [22].

The stick pulling experiment has been generalised in multiple ways, in order to study the ability to form collaborative teams tailored to task execution. For instance, the requirements for collaboration may involve up to k robots to perform some action [20]. These actions can be sequential (e.g., k actions are performed one after the other by at least two robots that take turn) or parallel (e.g., k robots must be present concurrently at the same place). The task demand has a bearing on the optimal dwelling time, too: the higher the number of robots required by a task, the longer the time to wait for the formation of a suitable team. After task execution, collaboration sites may disappear or regenerate at a given rate. Also, multiple types of tasks can be present at the same time with different requirements, calling for heterogeneity of behaviour, specialisation and learning [20], [23], [24]. More complex behaviours have been considered too, allowing robots to deliberate about the need to get engaged or not when meeting at collaboration sites [21].

Improving over previous work [24], we consider in this study two different types of tasks—easy and hard—that must be executed by a swarm of minimalist robots. Robots can execute any type of task as long as they participate to a suitably-large team. We consider non-trivial operating conditions characterised by a high density of tasks to be executed which contrasts with a relatively low robot density. This means that task execution depends on the likelihood that enough robots concurrently visit the same collaboration site. Additionally, we consider both homogeneous and heterogeneous task distribution in space, hence accounting for the possibility that in a given region only tasks of a given type are found. To maximise the task execution rate as well as the balance between executed tasks of different type, either robots are capable of identifying the task and tuning their behaviour accordingly—i.e, achieving task specialisation—or the likelihood of forming useful teams must be maximised creating the right operating conditions—i.e., achieving optimal robot density, albeit locally. In both cases, robots need to adaptively alter their behaviour in response to local cues, so that the right team composition is attained with high probability. In this work, we show how this is possible with minimal complexity. To bias the spatial distribution towards areas demanding more work, we propose to adaptively vary the robot motion pattern by switching between an isotropic random walk producing Brownian motion and a Lévy walk [25] in response to local task requirements. To optimise the individual behaviour, we adopt a simple learning rule that adjusts the dwelling time of robots in response to the collaboration outcome [26], [27]. Through simulations and real-robot experiments, we demonstrate that the proposed adaptive strategies can create the local conditions necessary to balance the execution of hard and easy tasks, biasing the robot distribution towards areas containing hard tasks and leading to heterogeneous specialised swarms.



(a) Mixed tasks (b) Spatially separated tasks

Fig. 1. Visualisation of the experimental setup. Collaboration sites spawn easy (magenta) or hard tasks (cyan), and turn inactive (gray) after task execution.

The remainder of this letter is structured as follows. In Section II, we describe in detail the experimental setup and the type of adaptive strategies devised to address the team formation problem. In Section III, we provide evidence of the advantage of adaptive strategies with extensive simulations and real robot experiments. Section IV concludes the letter with discussions on the relevance of the proposed approach for future swarm robotics systems.

II. Experimental Setup

We performed an extensive set of simulated and real-world experiments, both sharing the same setup. Experiments are performed with Kilobots [28], both in simulation with the ARGoS framework [29], [30] and with physical robots exploiting ARK [31] to provide additional abilities to the Kilobots, such as perception/execution of virtual tasks and wall avoidance. Specifically, virtual sensors are implemented to (i) detect the presence of a collaboration site spawning easy or hard tasks, and (ii) detect the presence and direction of a nearby wall. Detection and avoidance of nearby robots is not considered, leaving Kilobots free to collide and slide against each other. A group of $N = 24$ robots is deployed within a circular arena of radius $R = 0.45\mathrm{m}$ following a uniform distribution. The $M = 16$ collaboration sites are represented as circular areas of radius $r = 0.06\,\mathrm{m}$, also uniformly distributed inside the arena and non overlapping, as shown in Fig. 1. Each collaboration site is characterised by a task type—easy or hard—that never changes. Easy tasks require at least $k_e = 2$ robots at the collaboration site; hard tasks require at least $k_h = 4$ robots. In each experiment, there is an equal share $(M/2)$ of collaboration sites spawning easy and hard tasks. We consider two different problems: either the task distribution is mixed within the arena (see Fig. 1(a)), or tasks are spatially separated, with the arena split in two regions where easy and hard tasks occupy opposite sides (see Fig. 1(b)).

The individual robot behaviour can be described by a finite state machine—shown in Fig. 2—with three states: walk (W), dwell (D) and leave (L). In the walk state W, a robot wanders in search of tasks to be executed. When it finds a task available for execution (i.e., it enters *inside* an active collaboration site), the

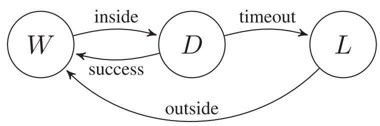


Fig. 2. Finite state machine of the individual robot behaviour.

robot switches to the dwell state D, in which it waits motionless for other robots to join. If a sufficient number of robots arrives at the site, the task is immediately executed. Otherwise, after waiting T_d seconds, a dwelling *timeout* expires and the robot switches to the leave state L, in which it moves away from the collaboration site ignoring the corresponding task. Once *outside*, the robot returns in state W. When a suitable team is formed, all robots that were in the dwell state D detect the *success* at task completion and resume their wandering (state W) in search of other tasks. Upon task completion, the corresponding collaboration site becomes inactive for $T_r = 30\,\mathrm{s}$. After this period, the collaboration site becomes active again, and a new task is spawned of the same type. Note that robots employ the same dwelling time for any task type. In this way, the behaviour is of minimal complexity and also generalises to problems where there are more than two task types.

In both states W and L, robots perform a random walk and avoid collisions with walls. Robots implement a Lévy-modulated correlated random walk (LMCRW) [25], which is characterised by two parameters controlling the distribution of turning angles and step lengths. Specifically, the turning angle is drawn from a wrapped Cauchy distribution with the following probability density function:

$$F(\theta;\rho) = \frac{1}{2\pi}\frac{1-\rho^2}{1+\rho^2+2\rho\cos(\theta)} \tag{1}$$

where ρ determines the distribution width: for $\rho = 0$ the distribution becomes uniform and provides no correlation between consecutive movements, while for $\rho = 1$ a Dirac distribution is obtained, corresponding to ballistic motion. The step length δ follows a Lévy distribution characterised by a power law $P(\delta) \approx \delta^{-(\alpha+1)}$, with $0 < \alpha \leq 2$. For $\alpha = 2$ the distribution becomes Gaussian, while for $\alpha \rightarrow 0$ the random walk reduces to ballistic motion. In this study, we limit the LMCRW to two sets of parameters: a **Brownian motion** ($\rho = 0$, $\alpha = 2$), which provides good local search abilities to robots, and a **persistent walk** ($\rho = 0.9$, $\alpha = 1.4$), characterised by directed motion and long relocations.

In order to match the local contingencies, robots adapt the type of random walk following encounters with hard or easy tasks. Specifically, after encountering an easy task, a robot chooses the parameters of the persistent walk. Conversely, after encountering a hard task, a robot chooses the parameters of a Brownian motion. In this way, we implement an **adaptive walk** just by changing the parameters ρ and α of the LMCRW. As a consequence, we expect that robots move away from easy tasks in search of hard tasks, which require more robots for completion, hence biasing the local density of robots in areas where hard tasks are present.

From previous studies on the stick pulling experiment and variations thereof, the dwelling time T_d is the parameter to be optimised in order to maximise the efficiency of the system, with the optimal value being dependent on the likelihood of robots to encounter a collaboration site, on the robot density, and on the number of robots required for task execution. When both hard and easy tasks are present, the choice of T_d may influence the trade-off between executing one or the other task type within a given time interval. In order to have robots specialise, we implement a simple learning mechanism through an **adaptive timeout**. In such conditions, T_d starts from 60 s and is decreased by a fixed amount $T_\Delta = 10\,\mathrm{s}$ when a task is successfully completed (without going below a minimum $T_m = 10\,\mathrm{s}$). Otherwise, an unsuccessful collaboration attempt would result in a corresponding increase of the dwelling time by T_Δ seconds. The rationale is that, upon success, robots can try to wait less as the dwelling time proved sufficient. Conversely, upon failure, robots try to wait longer at the next attempt.

III. Results

In order to measure the ability of minimalist robot swarms to form suitable teams for task execution, we measure the cumulative number of completed tasks within the total duration of an experiment, here fixed to $T_M = 1800\,\mathrm{s}$. Specifically, we count the number of easy tasks C_e and of hard tasks C_h successfully completed. On the one hand, our goal is to maximise the total number of completed tasks ($C_e + C_h$). On the other hand, considering that hard tasks require more robots to be completed, we are also interested in minimising the difference between the number of hard and easy tasks completed ($|C_e - C_h|$), as this corresponds to a balanced task execution.

We first consider the effect of a fixed timeout T_d on performance. In Fig. 3, we compare the completed tasks with robots executing different types of random walk—Brownian, persistent and adaptive—with $T_d \in [10, 600]\,\mathrm{s}$. Problem instances with both mixed and separated tasks are considered. Generally speaking, we observe that robots complete easy tasks more often than hard tasks. When tasks are mixed, the persistent walk makes robots able to reach and complete a high number of easy tasks, while still executing more hard tasks than in the Brownian motion case. This is because the persistent walk allows robots to quickly spread in space and reach different collaboration sites where other robots may be waiting for help. In other words, the persistent walk maximises the mixing of robots, increasing the likelihood that suitable teams are formed at any collaboration site. Given that easy tasks have lower requirements (i.e., $k_e < k_h$), they get completed at a higher rate. A similar pattern can be observed when tasks are separated, but in this case we note a higher variance in the execution of easy tasks when robots move according to a Brownian motion. In this case, diffusion is smaller and robots are less likely to change region.

The adaptive walk instead leads to a higher execution of hard tasks than the Brownian motion, and also outperforms the persistent walk for low values of T_d. However, the resources invested into hard tasks are distracted from the execution of easy tasks, and the total task completion substantially decreases.

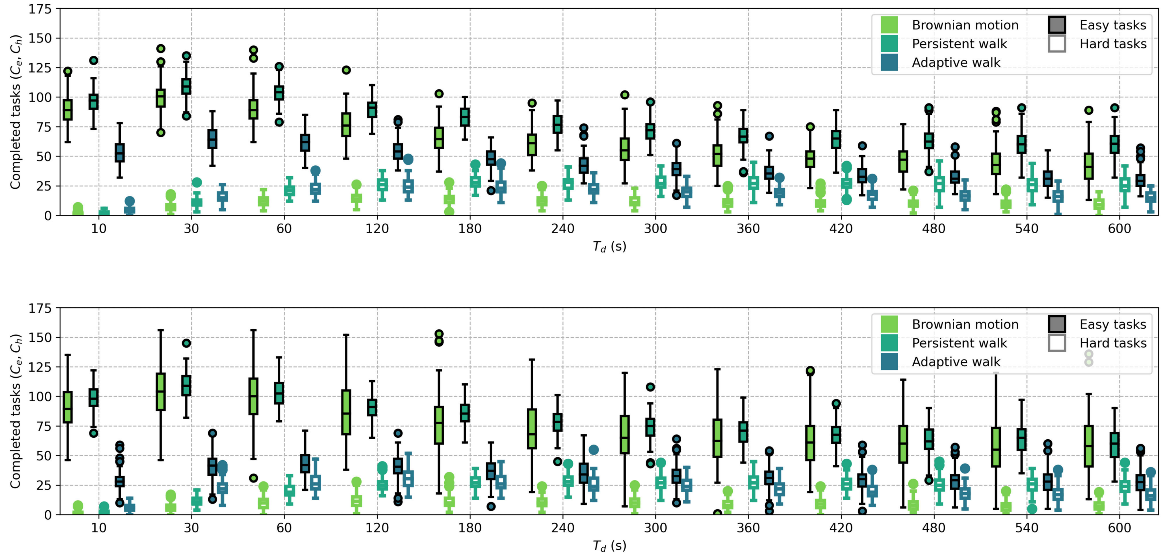


Fig. 3. Performance in terms of completed easy and hard tasks (respectively C_e and C_h) comparing different types of random walk. For each type of random walk and dwelling time T_d, we performed 100 independent simulations. Top: mixed tasks. Bottom: separated tasks.

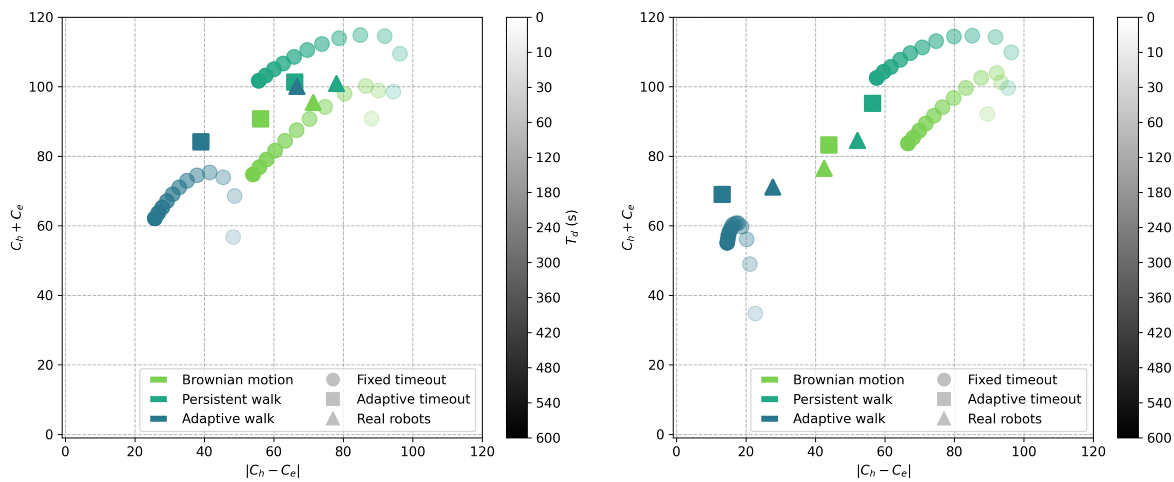


Fig. 4. Pareto efficiency of the different configurations proposed, with fixed (circles) and adaptive timeout (squares). The performance obtained from real robot experiments with adaptive timeout is also displayed (triangles). Appropriate configurations should maximise the total number of tasks completed $C_e + C_h$ (y axis, higher is better) and/or minimise the difference between easy and hard tasks accomplished $|C_e - C_h|$ (x axis, lower is better). Dominated configurations have at least one other configuration that is better in both objectives. Left: mixed tasks. Right: separated tasks.

This highlights the trade-off between the two objectives: maximising the total number of completed tasks and balancing between easy and hard tasks. This is perfectly visible in Fig. 4, where the Pareto efficiency is displayed with respect to these two objectives. The configurations exploiting persistent walk dominate those with the Brownian motion (i.e., they are better at both objectives), but do not dominate the adaptive walk, which is better off in minimising the gap between easy and hard tasks. It is also possible to note that high values of T_d provide non-dominated configurations, and only very small values result in fully dominated configurations. This is because a higher dwelling time provides an advantage for hard tasks but not for easy tasks, hence improves the balance but reduces the total task completion.

To understand how employing the adaptive walk can bias the execution of tasks towards hard ones, it is useful to look at the distribution of robots in space. In Fig. 5, we show the evolution over time of the fraction of robots found closer to collaboration sites spawning hard tasks. At the beginning, half of the robots are closer to hard tasks, due to the initial homogeneous distribution. With time, more robots are found closer to hard tasks for all cases, because robots wait longer for a sufficiently large team to form. However, when the random walk is adaptive, the fraction of robots closer to hard tasks grows and stabilises to higher values, meaning that robots naturally gravitate in regions around hard tasks. This is particularly visible when tasks are separated, with about 80% of the robots dedicated to executing hard tasks. Indeed, by executing a Brownian motion after encountering a hard task, robots search locally with a small diffusion constant. At the same time, executing a persistent walk after encountering an easy task makes robot move farther away, possibly ending up in a different region. Especially when tasks are separated,

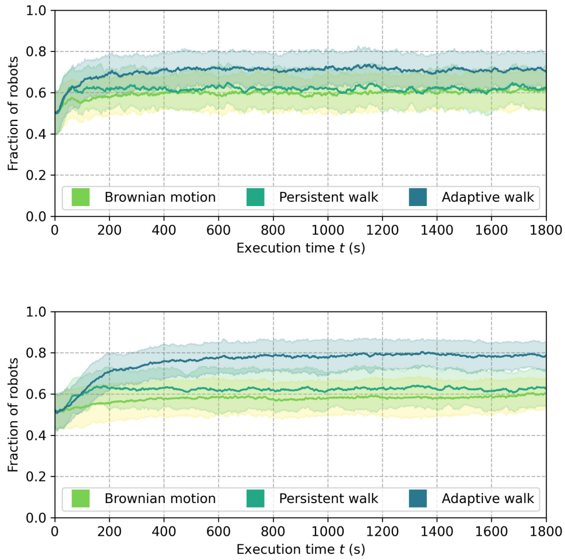


Fig. 5. Fraction of robots found closer to hard tasks over time, for different types of random walk and $T_d = 60$ s. Top: mixed tasks. Bottom: separated tasks.

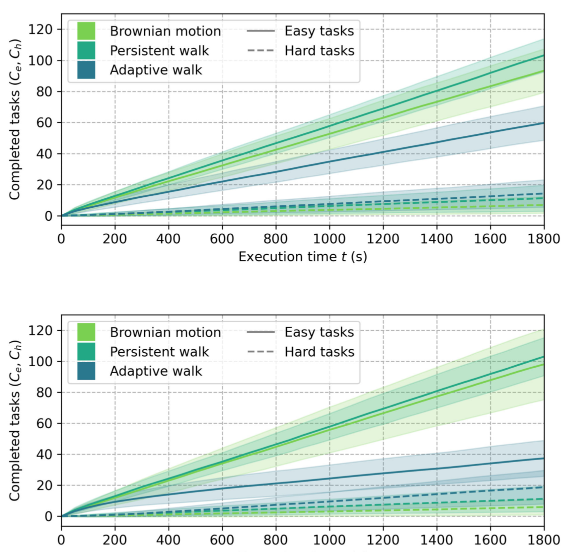


Fig. 6. Rate of task completion for different random walks and $T_d = 60$ s. Top: mixed tasks. Bottom: separated tasks.

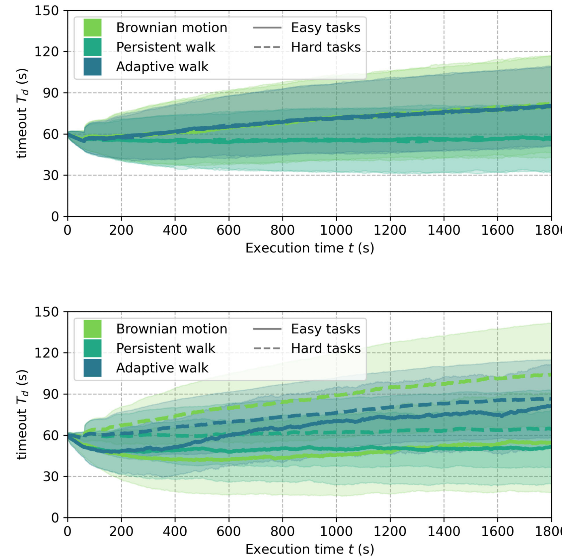


Fig. 7. Variation of the adaptive timeout T_d over time. At each time t, the robot population is partitioned between robots closer to easy or hard tasks, and the averages and standard deviations are computed for each sub-populations. The high standard deviation is a result of both (i) the different histories of adaptation from each robot, and (ii) the frequent switch of robots between sub-populations, especially when tasks are mixed. Top: mixed tasks. Bottom: separated tasks.

this represents a very simple and elegant mechanism to bias the spatial distribution of agents, implemented with minimal complexity at the behavioural level.

From Fig. 5 we also note that the spatial distribution remains substantially stable over time after the initial transitory period, meaning that there is an equal flux of robots between regions containing hard and easy tasks. This means that both types of tasks keep being executed, but at a rate that depends on the local density. This is particularly relevant for the case of separated tasks, because hard tasks can be executed at a rate that matches the one of easy tasks, as shown in Fig. 6. When tasks are mixed, instead, the likelihood of encountering easy tasks is larger than in the case of separated tasks, hence their execution rate remains higher.

Looking at Fig. 3, it is possible to note that the timeout T_d must be accurately tuned to find the best value. Additionally, easy and hard tasks have different requirements, the former demanding a smaller dwelling time than the latter. Considering that the adaptive walk makes some robots spend more time around hard tasks, it would be beneficial that they get specialised with longer dwelling times. Conversely, robots remaining close to easy tasks should shorten their dwelling time. By using an adaptive timeout as introduced in Section II, we aim at optimising the task execution rate by learning the best dwelling time for each robot. Fig. 4 shows that this mechanism properly works in increasing the number of completed tasks, but only for the adaptive walk. For the Brownian motion, the adaptive timeout leads to minor improvements in the task completion balance, the configurations being non dominated both when tasks are mixed and separated. For the persistent walk, instead, robots change often between hard and easy tasks and therefore the adaptive timeout does not provide any advantage.

The role of the adaptive timeout is to make robots specialised for the execution of easy or hard tasks, also making the swarm heterogeneous with different robot types: those that wait long for a team to form, and those that frequently switch from one collaboration area to another. To understand if specialisation emerges, we plot in Fig. 7 the variation of T_d over time for the sub-population of robots found closer either to hard or to easy tasks. When tasks are mixed, robots frequently switch from easy to hard tasks, and no real specialisation is observed, whatever type of random walk is employed. The average values are lower for the persistent walk as it is the one that leads to the highest task completion rate, hence the learning rule leads to smaller dwelling times. When tasks are separated instead, we note a stronger tendency to specialisation, with hard tasks promoting longer

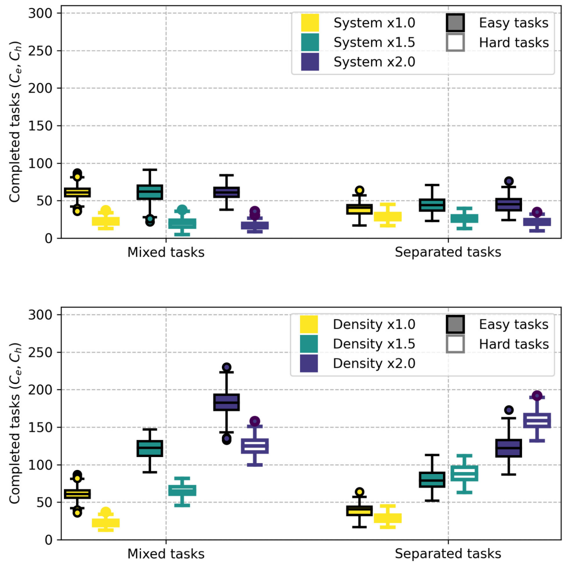


Fig. 8. Completed tasks for larger instances of the problem ($\times 1.0$, $\times 1.5$ and $\times 2.0$). Top: the swarm size is increased ($N \in \{24, 36, 48\}$) together with the arena size ($R \in \{R_0, \sqrt{1.5}R_0, \sqrt{2}R_0\}$, with $R_0 = 0.45\,\mathrm{m}$) and the number of collaboration sites ($M \in \{16, 24, 32\}$), resulting in a constant robot/task density across problem instances. Bottom: only the swarm size is increased ($N \in \{24, 36, 48\}$, $R = 0.45\,\mathrm{m}$ and $M = 16$), resulting in varying robot density.

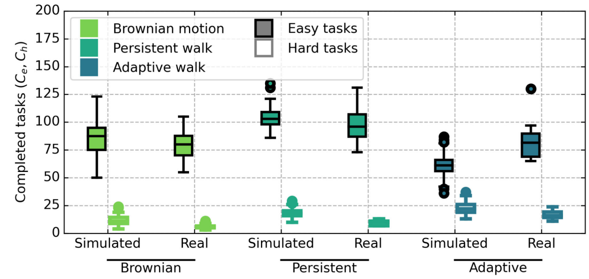


Fig. 9. Completed tasks when the adaptive timeout is used together with different types of random walk. Experiments are performed in simulation (100 independent runs) and with real robots (12 independent runs).

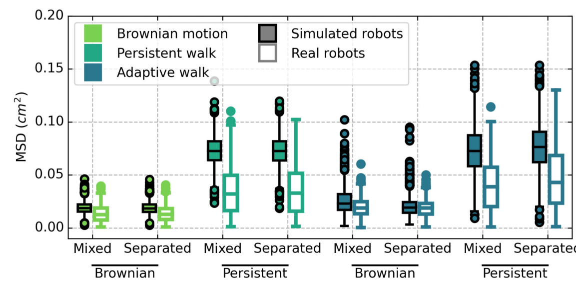


Fig. 10. Comparison of the Mean Square Displacement (MSD) between simulated and real robots, computed over portions of trajectories from robots executing either the Brownian motion or the persistent walk. For the adaptive walk, both types of trajectories have been considered.

dwelling times. Note the increase of T_d for robots found closer to easy tasks in the case of adaptive walk. This is because the reduced density of robots in the region with easy tasks requires a longer timeout, and the learning rule leads to a corresponding adaptation.

We also tested the scalability of the adaptive strategies with larger problem instances (see Fig. 8), testing how a larger number of robots perform when (i) the problem scale is also increased with a larger arena and a higher number of collaboration areas—hence maintaining the density of robots and tasks constant—and (ii) the environment and collaboration areas are kept at the original size—hence increasing the robot density. In the former case, we observe that the task execution rate is not varied, as expected from a well-behaving system (see the top panel in Fig. 8) In the latter case (bottom panel), the higher robot density facilitates the formation of teams, increasing the task execution rate linearly. Additionally, when tasks are separated, the adaptive walk leads to an even higher density of robots in the region with hard tasks, substantially increasing their execution rate, which becomes larger than for easy tasks. Overall, the adaptive strategies scale very well in both conditions.

Experiments performed with real Kilobots qualitatively confirm the simulation results (see also the video in the supplementary material), although a quantitative deviation from the simulated experiments can be observed in Fig. 4 and 9. Specifically, we observe a small reduction of the completed tasks for both the Brownian motion and the persistent walk. For the adaptive walk, we instead observe an increased completion of easy tasks C_e as well as a reduced completion of the hard tasks C_h. To understand the causes of such deviation, we compute the mean square displacement (MSD) of simulated and real robots. Specifically, for each robot, we consider multiple segments of trajectories of $40\,\mathrm{s}$ while in states W and L, and we distinguish between segments belonging to the Brownian and the persistent portion of the walk. The comparison between simulated and real robots in Fig. 10 reveals that the latter diffuse slightly less, especially while executing the persistent walk. The reason for this is the higher difficulty in moving in a straight line, due to noise in the motor calibration and in small asperities of the ground, which deeply affect Kilobots in efficiently moving. Hence, real Kilobots diffuse less than simulated ones. This leads to lower task execution rate when the spatial distribution is not biased. In the case of the adaptive walk, instead, robots are less efficient in getting away from easy tasks, which justifies why C_e is higher with real robots than with simulated ones. Consequently, C_h is slightly lower (see Fig. 9).

IV. Conclusion

In this letter, we demonstrated how simple adaptive strategies can lead to improved team formation, privileging the completion of hard tasks and improving a balanced task execution. Given the minimalist approach undertaken, team formation must be opportunistic and exploit the stochastic nature of the system by biasing the behaviour of robots in favour of hard tasks. As task execution is dependent on the local robot density, we demonstrate how switching between Brownian motion and persistent walks can lead to a biased spatial distribution of robots. Additionally, a simple learning rule contributes to obtain robots that are specialised in executing specific tasks. These simple mechanisms are sufficient to optimise the local conditions for the formation of suitable teams. While the proposed approach may appear specifically tailored to the problem instances proposed here, it is easy to imagine how it can generalise to different problem instances within the context of team formation and beyond. Indeed, similar adaptive mechanisms can be exploited to promote self-organised aggregation by varying the random walk persistence as a function of the number of detected neighbours, as well as other collective behaviours in micro/nano-robots or active colloids where motion can be controlled only in terms of statistical properties.

With slightly more complexity, adaptive strategies can be developed to finely tune the local density of robots, and therefore the task execution rate. This is possible if other cues can be recognised beyond the presence of easy or hard tasks. For instance, robots could recognise the rate of encounter of tasks, or the rate of encounter of other robots [32]. The rationale stays the same: the local density increases in regions where robots perform Brownian motion, and decreases where robots perform a more persistent walk. Different parameterisations for the two types of random walk could also provide interesting results, enabling better control of the effective transition rates of robots between areas of high and low robot density. To this end, future work can target the development of macroscopic models of the system dynamics, which can link the diffusivity of the individual random walk to the emergent macroscopic pattern. Given such models, it is possible to choose the best parameters for the different modes of random walk, creating adaptive strategies that optimise desired macroscopic measures and further generalise towards more complex application scenarios.

The adaptive timeout we implemented corresponds to a simple learning rule, and we showed that it brings advantages especially when a biased spatial distribution is achieved. In such conditions, robots do not frequently encounter different types of tasks, and can therefore benefit from specialisation. Similar learning rules are therefore suitable in swarm robotics when different operating conditions might be encountered that require situated adaptation, and should be investigated further because heterogeneity of behaviour can lead to higher performance [8].

References

[1] M. Dorigo, G. Theraulaz, and V. Trianni, “Swarm robotics: Past, present, and future [Point of View],” *Proc. IEEE*, vol. 109, no. 7, pp. 1152–1165, Jul. 2021.

[2] H. Xie *et al.*, “Reconfigurable magnetic microrobot swarm: Multimode transformation, locomotion, and manipulation,” *Sci. Robot.*, vol. 4, no. 28, 2019, Art. no. eaav8006.

[3] M. Z. Miskin *et al.*, “Electronically integrated, mass-manufactured, microscopic robots,” *Nature*, vol. 584, no. 7822, pp. 557–561, 2020.

[4] S. M. Douglas, I. Bachelet, and G. M. Church, “A logic-gated nanorobot for targeted transport of molecular payloads,” *Science*, vol. 335, no. 6070, pp. 831–834, 2012.

[5] S. Jeon *et al.*, “Magnetically actuated microrobots as a platform for stem cell transplantation,” *Sci. Robot.*, vol. 4, no. 30, 2019, Art. no. eaav4317.

[6] F. Berlinger, M. Gauci, and R. Nagpal, “Implicit coordination for 3D underwater collective behaviors in a fish-inspired robot swarm,” *Sci. Robot.*, vol. 6, no. 50, 2021, Art. no. eabd8668.

[7] Z. Tao *et al.*, “The progress of extraterrestrial regolith-sampling robots,” *Nature Astron.*, vol. 3, no. 6, pp. 487–497, 2019.

[8] M. Dorigo, G. Theraulaz, and V. Trianni, “Reflections on the future of swarm robotics,” *Sci. Robot.*, vol. 5, no. 49, 2020, Art. no. eabe4385.

[9] M. Gauci, J. Chen, W. Li, T. J. Dodd, and R. Gross, “Self-organized aggregation without computation,” *Int. J. Robot. Res.*, vol. 33, no. 8, pp. 1145–1161, 2014.

[10] A. Özdemir, M. Gauci, S. Bonnet, and R. Groß, “Finding consensus without computation,” *IEEE Robot. Automat. Lett.*, vol. 3, no. 3, pp. 1346–1353, Jul. 2018.

[11] J. Palacci, S. Sacanna, A. P. Steinberg, D. J. Pine, and P. M. Chaikin, “Living crystals of light-activated colloidal surfers,” *Science*, vol. 339, no. 6122, pp. 936–940, 2013.

[12] F. Ginot, I. Theurkauff, F. Detcheverry, C. Ybert, and C. Cottin-Bizonne, “Aggregation-fragmentation and individual dynamics of active clusters,” *Nature Commun.*, vol. 9, no. 1, 2018, Art. no. 696.

[13] H. Karani, G. E. Pradillo, and P. M. Vlahovska, “Tuning the random walk of active colloids: From individual run-and-tumble to dynamic clustering,” *Phys. Rev. Lett.*, vol. 123, 2019, Art. no. 208002.

[14] M. Fernandez-Rodriguez *et al.*, “Feedback-controlled active brownian colloids with space-dependent rotational dynamics,” *Nature Commun.*, vol. 11, no. 1, 2020, Art. no. 4223.

[15] K. Dietrich, N. Jaensson, I. Buttinoni, G. Volpe, and L. Isa, “Microscale marangoni surfers,” *Phys. Rev. Lett.*, vol. 125, 2020, Art. no. 098001.

[16] D. Albani, W. Hönig, D. Nardi, N. Ayanian, and V. Trianni, “Hierarchical task assignment and path finding with limited communication for robot swarms,” *Appl. Sci.*, vol. 11, no. 7, 2021, Art. no. 3115.

[17] S. Berman, A. Halasz, M. A. Hsieh, and V. Kumar, “Optimized stochastic policies for task allocation in swarms of robots,” *IEEE Trans. Robot.*, vol. 25, no. 4, pp. 927–937, Aug. 2009.

[18] S. Wilson, T. P. Pavlic, G. P. Kumar, A. Buffin, S. C. Pratt, and S. Berman, “Design of ant-inspired stochastic control policies for collective transport by robotic swarms,” *Swarm Intell.*, vol. 8, no. 4, pp. 303–327, 2014.

[19] A. J. Ijspeert, A. Martinoli, A. Billard, and L. M. Gambardella, “Collaboration through the exploitation of local interactions in autonomous collective robotics: The stick pulling experiment,” *Auton. Robots*, vol. 11, no. 2, pp. 149–171, 2001.

[20] L. Li, A. Martinoli, and Y. S. Abu-Mostafa, “Learning and measuring specialization in collaborative swarm systems,” *Adaptive Behav.*, vol. 12, no. 3/4, pp. 199–212, 2004.

[21] A. Kanakia, J. Klingner, and N. Correll, “A response threshold sigmoid function model for swarm robot collaboration,” in *Distributed Autonomous Robotic Systems*, N.-Y. Chong and Y.-J. Cho, Eds. Berlin, Germany: Springer, 2016, pp. 193–206.

[22] K. Lerman, A. Galstyan, A. Martinoli, and A. Ijspeert, “A macroscopic analytical model of collaboration in distributed robotic systems,” *Artif. Life*, vol. 7, no. 4, pp. 375–393, 2001.

[23] T. W. Mather, M. A. Hsieh, and E. Frazzoli, “Towards dynamic team formation for robot ensembles,” in *Proc. IEEE Int. Conf. Robot. Automat.*, 2010, pp. 4970–4975.

[24] Á. M. Halász, Y. Liang, M. A. Hsieh, and H.-J. Lai, *Emergence of Specialization in a Swarm of Robots*. Berlin, Germany: Springer, 2013, pp. 403–416.

[25] C. Dimidov, G. Oriolo, and V. Trianni, “Random walks in swarm robotics: An experiment with Kilobots,” in *Proc. Swarm Intell., 10th Int. Conf., Ser. Lecture Notes Comput. Sci.*, vol. 9882. Brussels, Belgium: Springer, 2016, pp. 185–196.

[26] G. Theraulaz, E. Bonabeau, and J. N. Denuebourg, “Response threshold reinforcements and division of labour in insect societies,” *Proc. Roy. Soc. London. Ser. B, Biol. Sci.*, vol. 265, no. 1393, pp. 327–332, 1998.

[27] E. Castello *et al.*, “Adaptive foraging for simulated and real robotic swarms: The dynamical response threshold approach,” *Swarm Intell.*, vol. 10, no. 1, pp. 1–31, 2015.

[28] M. Rubenstein, C. Ahler, N. Hoff, A. Cabrera, and R. Nagpal, “Kilobot: A low cost robot with scalable operations designed for collective behaviors,” *Robot. Auton. Syst.*, vol. 62, no. 7, pp. 966–975, 2014.

[29] C. Pinciroli *et al.*, “ARGoS: A modular, parallel, multi-engine simulator for multi-robot systems,” *Swarm Intell.*, vol. 6, no. 4, pp. 271–295, 2012.

[30] C. M. Pinciroli, S. Talamali, A. Reina, J. A. R. Marshall, and V. Trianni, “Simulating Kilobots within ARGoS: Models and experimental validation,” in *Swarm Intelligence, Ser. Lecture Notes in Computer Sciences*, M. Dorigo, M. Birattari, C. Blum, A. L. Christensen, A. Reina, and V. Trianni, Eds., vol. 11172. Cham, Switzerland: Springer, 2018, pp. 176–187.

[31] A. Reina, A. J. Cope, E. Nikolaidis, J. A. R. Marshall, and C. Sabo, “ARK: Augmented reality for Kilobots,” *IEEE Robot. Automat. Lett.*, vol. 2, no. 3, pp. 1755–1761, Jul. 2017.

[32] S. Mayya, S. Wilson, and M. Egerstedt, “Closed-loop task allocation in robot swarms using inter-robot encounters,” *Swarm Intell.*, vol. 13, no. 2, pp. 115–143, 2019.