



# Resilient Robot Teams: a Review Integrating Decentralised Control, Change-Detection, and Learning

David M. Bossens<sup>1</sup> · Sarvapali Ramchurn<sup>1</sup> · Danesh Tarapore<sup>1</sup>

Accepted: 13 April 2022 / Published online: 13 June 2022  
© The Author(s) 2022

## Abstract

**Purpose of Review** This paper reviews opportunities and challenges for decentralised control, change-detection, and learning in the context of resilient robot teams.

**Recent Findings** Exogenous fault-detection methods can provide a generic detection or a specific diagnosis with a recovery solution. Robot teams can perform active and distributed sensing for detecting changes in the environment, including identifying and tracking dynamic anomalies, as well as collaboratively mapping dynamic environments. Resilient methods for decentralised control have been developed in learning perception-action-communication loops, multi-agent reinforcement learning, embodied evolution, offline evolution with online adaptation, explicit task allocation, and stigmergy in swarm robotics.

**Summary** Remaining challenges for resilient robot teams are integrating change-detection and trial-and-error learning methods, obtaining reliable performance evaluations under constrained evaluation time, improving the safety of resilient robot teams, theoretical results demonstrating rapid adaptation to given environmental perturbations, and designing realistic and compelling case studies.

**Keywords** Change-detection · Decentralised control · Multi-robot systems · Swarm robotics · Machine learning · Multi-agent systems

## Introduction

From monitoring wildlife [1], exploring dangerous terrains [2], or collaboratively transporting items [3], robotic systems have the potential to transform society by performing tasks that are too dangerous, laborious, or repetitive for humans. In many such tasks, employing a *robot team*, which consists of multiple robots, often yields significant advantages over employing just a single robot. For example, robot teams can monitor larger areas due to a larger spatial spread and

provide robustness to failure due to redundancy (i.e. multiple robots performing the same or a similar subtask) [4].

In any given mission, robots in the team must control their own actuators and outgoing communications based on local sensory data and incoming communications. Decentralised control in this sense has been studied for multi-robot systems [5, 6], which have fewer robots with more powerful hardware and high-complexity algorithms such as deep neural networks (e.g. [7••]), and swarm robotics [8], which is scalable in team size but comes with simplistic hardware and low-complexity algorithms such as ant-inspired algorithms (e.g. [9]).

When robot teams are expected to become employed in long-term autonomy, they must be resilient to disruptions such as sensory-motor faults, weather conditions affecting the operating environment, or even adversarial cyber-attacks. The designer may anticipate some of these disruptions and hard-code their diagnoses and solutions; however, when there are complex or unanticipated disruptions, the robot team must detect these and adapt accordingly. While generic methods for change-detection [10, 11] and transfer

---

This article is part of the Topical Collection on *Group Robotics*

✉ David M. Bossens  
d.m.bossens@soton.ac.uk

✉ Danesh Tarapore  
dst1m17@soton.ac.uk

Sarpapali Ramchurn  
sdr1@soton.ac.uk

<sup>1</sup> School of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, UK

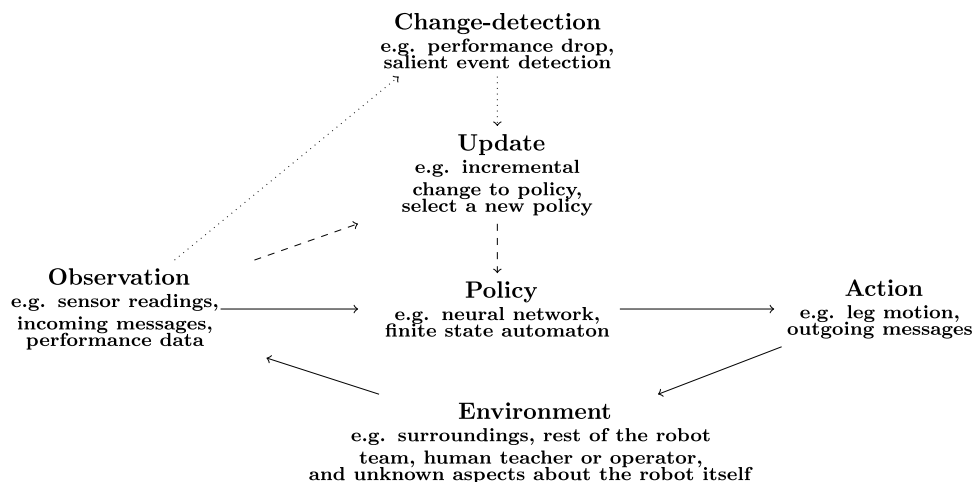
learning [12] have been applied in abstract machine learning problems, their application to robot teams comes with unique challenges such as communicating and integrating data from different robots' local error-prone observations and cooperative learning without incurring costs within the physical environment.

With this context in mind, this paper reviews ongoing research on resilient robot teams within a decision process framework integrating decentralised control, change-detection, and learning. Section 2 first presents the integrated decision process framework and different prototype approaches within the framework. Section 3 reviews how robot teams can cooperate in detecting faults as well as changes in the surrounding environment. Section 4 reviews how decentralised control strategies can be learned for allowing robot teams to adapt to changed environments. Finally, Section 5 provides the main conclusions from the review along with future research directions.

## Integrating Decentralised Control, Change-Detection, and Learning

From decentralised control to change-detection and adaptation to changes, robot teams face a challenging decision process. As illustrated in the integrated decision process framework shown in Fig. 1, each robot in the team must integrate its sensory readings and incoming communications to decide on which movements to take, which communications to send, and when and how to adapt to changes in the environment. To achieve this aim, each robot in the team uses a policy which selects an action to take at each control cycle, and performs change-detection and learning to adapt. Within our integrated decision process framework, the following section establishes a common vocabulary and key concepts for analysing the various methodologies for resilient robot teams.

**Fig. 1** Integrated decision process framework. The framework integrates decentralised control, change-detection, and learning into a single loop describing the decision problem faced by each robot in a robot team. Solid lines indicate the control cycle, dashed lines indicate the learning cycle performing policy updates periodically, and dotted lines indicate the optional change-detection cycle



## Communication

Communication is an essential part of the decision-making process and comes in two forms, namely explicit communication and implicit communication [6]. **Explicit communication** is an encoded message from a transmitter to receiver, and includes raw [13] or processed sensory readings [14], tasks [15], and the performance of team members [16]. Being communicated via wireless connections, they come with limitations such as bandwidth, cost, and delays. In **implicit communication**, the sender manipulates the environment, for example by using gestures [17] or by placing objects in the environment as signs [9], and the receiver observes this directly via sensory readings. It avoids communication overload but may have more limited range, making it preferable for high robot densities.

When the robot-environment interaction changes, for instance due to sensory-motor faults or communication disturbances, the best communication strategy may change over time; in this case, learning-how-to-communicate is an important capability. **Learning-how-to-communicate** approaches [17–20] include aspects of communication into the policy to be updated during the learning cycle.

## Adaptation to Changes

To allow rapid adaptation to changes, the integrated decision process framework requires (1) a policy space that captures the desired behaviours in the policy space without needlessly increasing the search space; (2) a learning algorithm with strong empirical and theoretical support; and (3) a change-detection algorithm that balances genericity of application with specificity of diagnosis to inform the learning algorithm.

Research in resilient robot teams has addressed these issues from different strategies, which we categorise based

on four prototypes, namely diagnose-and-solve, trial-and-error, teacher-student, and robust-by-design, as summarised in Table 1. In **diagnose-and-solve**, map a diagnose (e.g. localisation error) onto a specific “repair” (e.g. camera recalibration). This is the reasoning behind traditional fault-detection and fault-diagnosis methods [21] such as learning-based fault-diagnosis [16]. In **trial-and-error**, the robots adapt to the environment by trying out different policies and evaluating their performance. Key approaches include multi-agent reinforcement learning [22], embodied evolution [23], and offline evolution with online adaptation [24••]. In the **teacher-student**, the robots learn their policy from a teacher, typically using a large data set. The approach is typically based on imitation learning, i.e. by supervised learning from a data set of desired behaviours (e.g. [7••]), where resilience can be provided by generalisation or by learning on a newly provided data set. The source of the data set varies but common examples are simulated trajectories generated by centralised expert controllers which have global state information [7••, 25] and real-world video recordings of humans performing the desired behaviour [26]. In **robust-by-design**, each control cycle accounts for a range of environmental changes, for example by implicitly communicating occupancy map changes [9] or by explicitly communicating the availability and capability of team members [27].

The above-mentioned approaches are analysed in more fine-grained detail in Section 3 and Section 4. For now, we provide an overview of these approaches’ applicability to different change types in Table 2. Changes in **team members** include changes in team size or in the capability or functionality of team members. Changes in **communication** include communication noise and the limited range of communication potentially cutting off some team members. Changes in the **transition dynamics or task requirements** include moving targets, obstructions, task complexity increases, and even mission objective changes.

## Change-Detection

Although the topic of change-detection is broad, we focus on the unique opportunities and challenges in the context of resilient robot teams. In particular, we focus on identifying faults in team members and detecting and tracking changes in the surroundings. The former follows the diagnose-and-solve prototype while the latter often requires more generic adaptation methods.

### Detecting Faults in Team Members

At any time in their mission, members of a robot team may experience faults. Robots within a team may detect faults in themselves, or endogenous fault-detection. However, exogenous fault-detection, in which robots detect faults in each other based on data collected from the different team members, makes full use of the team’s joint capability to provide comparably higher efficiency and robustness. We distinguish here between four methods to detecting faults in team members, with an emphasis on exogenous fault-detection. The methods vary in their assumptions, genericity, and scalability, namely model-based fault-detection, feature-based anomaly detection, synchronisation, and cryptographic authentication.

Model-based methods model faults at design-time and detect them at run-time. Learning-based fault-diagnosis [16] pre-determines the causal links between symptom, fault, and solution for a priori known faults. It uses case-based reasoning to identify new cases by comparing them to the existing database of causal links (containing pre-defined or previously encountered faults) based on the most probable symptoms that are relevant for task completion. If the new case cannot be classified in terms of pre-existing causal links, then a human operator must communicate the recovery solution. Another approach analyses the difference of the

**Table 1** Adaptation prototypes within the integrated decision framework

Prototype	Policy space	Change-detection	Learning	Approaches
Diagnose-and-solve	Domain-specific	Diagnosis of a priori known faults	No learning or limited to generalising solutions to new faults	Fault-detection and fault-diagnosis
Trial-and-error	Generic	Generic change-detection	Learning by trial-and-error	Multi-agent reinforcement learning, embodied evolution, and offline evolution with online adaptation
Teacher-student	Generic	New data provided by a teacher, often a human operator	Imitation learning on provided data set	Learning perception-action-communication loops
Robust-by-design	Domain-specific	Implicit or explicit communication	Frequent updates on desired task allocation	Explicit task allocation, stigmergy in swarm robotics

Columns indicate the prototypes and their various properties: **Policy space** indicates the space of possible policies for decentralised control; **Change-detection** indicates how change-detection is typically implemented; **Learning** indicates how the policy is adapted upon a detected environmental change. **Approaches** lists the different classes of algorithms within the adaptation prototype

**Table 2** Change types and available methods

Change type	Examples	Change-detection	Policy learning
Team members	Team size change	Model-based fault-detection [16, 28, 29]	Embodied evolution [23, 35–38]
	Sensory-motor or software	Feature-based fault-detection [30–32]	Offline evolution with online adaptation [24••, 39]
	Fault	Synchronisation [33]	Adaptive specialisation [27, 40, 41••]
	Capability change	Cryptography [34•]	Explicit negotiation [15, 42] Ad hoc teamwork [43] Dynamic DCOPs [44, 45]
Communication	Communication noise	Model-based fault-detection [16, 28, 29]	Perception-action-communication loops [7••, 25]
	Local communication range		Independent Dec-POMDPs [46] Networking Dec-POMDPs [47, 48••]
Transition dynamics or task requirements	Dynamic targets	Task identification [49–51]	Multi-task multi-agent reinforcement learning [46]
	Obstructions	Dynamic and multi-vehicle mapping [14, 52, 53]	Embodied evolution [23, 35–38, 56]
	Task complexity increase	Multi-anomaly detection and tracking [13, 54, 55]	Offline evolution with online adaptation [24••, 39]
	Mission objective change		Dynamic DCOPs [44, 45]

observed dynamics to the dynamics expected under normal or faulty conditions. This method was first pursued in endogenous fault-detection using Kalman filters as models for linear dynamical systems [57, 58] or neural network models for more generic applicability [59, 60]. Later, in a method suitable for endogenous and exogenous fault-detection, Christensen et al. [28, 29] use time-delay neural networks to model faulty sensors or actuators by fitting sensory-motor data coming from trajectories under the fault and compare these to observed sensory-motor data observed from other robots. The expressivity of neural networks makes the approach ideal for detecting complex symptoms of failure. Unlike learning-based fault-diagnosis, the approach is sensitive to an arbitrary threshold parameter for the neural network classification and does not lend well to high-level behaviours such as path planning, localisation, and following behaviours. Scalability is a question mark, as the methodology was shown on the leader-follower task which only contains two robots.

A generic and scalable approach to exogenous fault-detection is to perform distance- or probability-based anomaly detection based on team members' feature vectors which record features of interest received by explicit communications from team members (e.g. sensory-motor trajectories). In Lau et al. [30], robots check from communicated data whether or not other robots experience the same change and if not a fault is present. Their results find that the receptor density algorithm [61], a non-parametric kernel-based technique inspired by T-cell receptors, significantly outperformed 4 other statistical tests. In a related

immune system inspired method, faults are detected by individual robots applying a cross-regulation model [62, 63] on behavioural feature vectors of neighbouring robots and then voting on which robots are faulty [31, 32•]; the method has been evaluated on a variety of tasks and environmental changes (including physical experiments [32•]), showing tolerance to changing swarm behaviors as well as individual robots behaving anomalously. A potential downside of the immune-inspired approach is that rather than faults, differences in behavioural feature vectors might reflect localised environmental conditions such as reduced friction or the presence of obstacles.

Synchronisation [33] assumes robots are required to send out physical signals (e.g. LED light flashes) and require them back from their neighbouring team members; if a neighbouring robot does not synchronise, it must have a fault of some sort (e.g. light sensors, light actuators, or motion actuators). Unfortunately, the approach is rather hardware-specific and does not easily extend to fault-diagnosis.

Cryptographic approaches focus specifically on identifying team members that have been compromised by a cyber-attack. The approach by Ferrer et al. [34•] relies on a Merkle tree data structure, which ensures that each robot must share cryptographic proofs to verify their integrity before cooperating on the mission. While the approach is only applicable to security risks, it can detect such changes even when the robot under attack does not demonstrate any observable behavioural differences.

## Detecting Changes in the Surroundings

Compared to traditional change-detection in machine learning, robot teams are distributed in space and can actively reposition themselves to make sense of the surrounding environment, which has special relevance for applications such as disaster recovery, search-and-rescue, and environmental monitoring.

One domain of interest is detecting and tracking anomalies in the environment. In the approach by Saldana et al. [54], team members can sense anomalous regions directly in the environment, explicitly communicate their observations to each other, and try to surround multiple existing anomalies while exploring the map to find new anomalies. The approach by Li et al. [13] formulates a dynamic optimisation problem in which the optimum to be tracked is the maximum (or minimum) on a particular feature of interest. Robots function as distributed searchers that occupy promising areas more densely and that communicate each others' measurements. Unfortunately, the approach was only demonstrated on toy function optimisation problems, so a physical robotics demonstration could make the approach more convincing. For physical fields such as oceans, Salam et al. [55] present an algorithm for estimating the full state of a dynamic process based on robots within a team communicating their local observations of the quantity to be tracked (e.g. concentration of particles, temperature) and then recomputing the updated system dynamics. The approach was demonstrated to have high accuracy compared to radial basis function interpolation in a physical water tank.

Similarly, due to their distributed active sensing capabilities, cooperative techniques are also being investigated for simultaneous localisation and mapping (SLAM) within robot teams [52]. Traditional multi-vehicle SLAM has two main disadvantages, namely that forming joint map from the local observations has high computation and communication overhead and that the objects in the map formed by SLAM are assumed to be static. To counter these issues, one can track static and dynamic features in the map using dynamic occupancy grids [53] and communicate local maps rather than raw sensory data [14].

## Policy Learning for Decentralised Control

This section describes how policies can be learned for allowing robot teams to adapt to changed environments. We distinguish broadly between six approaches: one teacher-student approach called learning perception-action-communication loops; three trial-and-error approaches, namely multi-agent reinforcement learning, embodied evolution, and evolution with online adaptation; and two robust-by-design

approaches called explicit task allocation and stigmergy in swarm robotics.

## Learning Perception-Action-Communication Loops

Incorporating perception-action-communication loops into graph neural networks (GNNs) is a recent solution to decentralised control and learning-how-to-communicate [7••, 25]. In such works, the GNN takes as input a communication graph, which represents robots as nodes, communication links as edges, and communication delays as their distances. The GNN is learned via imitation learning, a teacher-student approach in which the policies of the robots are updated by supervised learning on trajectories collected by the teacher (e.g. a human operator who has evaluated the best joint policy for the new situation). As in Hu et al. [7••], which assumes each robot has a transceiver, the graph neural network can learn to process incoming messages and send outgoing messages, and integrate this with local visual observations, resulting in a system that is robust to visual degradation as well as changes in team size and communication graph topology. A key challenge for imitation learning is that expert knowledge of the teacher will not always be available. Although domain adaptive imitation learning [64] as explicitly targeted imitation learning across different tasks, this still relies on the availability of expert data sets for each task. Alternatively, few-shot imitation methods [26] provide an option for imitation learning with limited data but so far has been studied only in single-robot contexts.

## Multi-agent Reinforcement Learning

Multi-agent reinforcement learning (MARL) is a relatively mature field which is grounded in the sound theoretical foundations of Markov decision processes with various analytical convergence results (e.g. [65]). In MARL, different agents receive observations of the environment, perform an action, and receive rewards indicating how desirable the behaviour is, and by trial-and-error the agents learn to collectively optimise the long-term cumulative reward. In MARL, resilience is conceptualised as learning a near-optimal policy in response to a change in the decision process.

Among the variety of MARL frameworks, decentralised partially observable Markov decision processes (Dec-POMDPs; [66]) are of particular relevance to robot teams, as they integrate partial observability (e.g. due to limited sensory observations) into decentralised control. While there have been a variety of powerful alternative frameworks, such as Communicative Multi-agent Team Decision Problems [67], these have not been as widely investigated. Recent works within Dec-POMDPs has examined multi-task learning and robustness to communication failures. Decentralised hysteretic deep recurrent Q-network (Dec-HDRQN) [46] uses

hysteretic Q-learning [68], in which agents update their own independent Q-table with lower learning rate (without communication) when this is likely due to other agents' suboptimal actions, while using deep recurrent Q-network [69] for partial observability and policy distillation [70] for improved generalisation. Dec-HDRQN allows learning many tasks within a single policy without requiring explicit task identification, as demonstrated on tasks with different grid sizes and transition dynamics in a grid-world domain. While independent decentralised control has benefits when communication channels are unreliable or faulty, the strategy is suboptimal. Instead, “networking” Dec-POMDPs can additionally incorporate communication amongst the different agents; popular approaches include sharing parameters and then forming a consensus [65], factorisation [71, 72], and communication protocols learned concurrently with the policy [47, 73]. Among these, CommNet [47] accounts for dynamic variation of the type and number of agents communicating and has been demonstrated for energy sharing in multi-UAV systems for distributed data processing [48••].

Assuming a change is detected in the decision process (e.g. using task identification methods [49–51]), MARL systems require extensive experience to learn a suitable policy if it is not already available from prior training. When a high-fidelity simulator is available, this may not be problematic and in this case one may also consider improving performance by applying centralised training with decentralised execution in the online phase (e.g. [74, 75]). Alternatively, for rapid adaptation, (theoretical or empirical) demonstrations of sample complexity are required rather than the asymptotic convergence guarantees for model-free MARL (e.g. methods based on Q-learning [76]).

## Evolution

Evolutionary algorithms (EAs) mimic natural evolution to generate a diverse population of genomes and progressively select them for fitness over subsequent generations. While it is popular to evolve policies with EAs — for example, using NEAT [77] to evolve the weights and topology of neural networks — in the context of resilient robot teams we mainly find two dominant approaches, namely embodied evolution and offline evolution with online adaptation.

### Embodied Evolution

Embodied evolution [23] evolves robots within their physical environment, which helps to avoid the “simulation-reality gap” [78] and to adapt to changes in the real world. Each robot executes its policy based on its current genotype but the policy has a limited lifespan based on a “virtual energy” quantity which represents the robot's own performance

estimate and which (in some implementations) improves the chances of reproduction. For reproduction, robots communicate with each other by broadcasting their own genotype, some genes of which are then integrated into the receiving robots' genotypes.

Unfortunately, this process of adaptation can take hours [36, 56]. Functional diversity can also be developed by using a quality-diversity approach to embodied evolution [35], which can be used as an implicit task allocation. With focus on real robot swarms, Silva et al. (2017) [37] demonstrate odNEAT [56], an online variant of NEAT, as being resilient to simulation-to-reality transfer, task change, and fault injection.

Rather than evolving genotypes, embodied evolution may also operate on memes, which are cultural utterances defined in robot teams as “contiguous sequences or packages of movements, or sounds, copied from one robot to another, by imitation” [38]. This approach uses implicit communication by observing each others' behaviours with sensory observations, which reduces communication overhead and more directly searches the policy space but is limited by observability.

### Offline Evolution with Online Adaptation

Embodied evolution is time-consuming, so another approach is to first evolve a large archive of policies offline and then perform rapid online adaptation by searching this policy space after a change (e.g. performance drop) is detected. In this context, offline evolution is based on quality-diversity (QD) algorithms (e.g. [79, 80]), which evolve an archive of behaviourally diverse and high-performing solutions, while online adaptation is based on Bayesian optimisation and has been studied primarily in single-robot adaptation to damaged actuators [81–83].

Two recent methods have expanded the approach to be applied in robot teams and in a wider set of environmental changes. Swarm Map-based Optimisation Decentralised (SMBO-Dec) [24••] forms a Gaussian process model for each subgroup within the team based on local environmental conditions. Different robots in the team function as different workers in an asynchronous batch-based Bayesian optimisation, which helps to speed up the search while avoiding to sample similar behaviours simultaneously. The empirical success was demonstrated by 80% performance improvements within a mere 30 evaluations on a large variety of perturbations, including food scarcity and different sensory-motor faults. Due to subgrouping the team based on local conditions, the approach combines naturally with diagnoses of change-detection algorithms, although this option has not been explored yet. Quality-Environment-Diversity (QED) [39] evolves behavioural diversity based on the type of environments that the policies solve. Since QED archives

represent solutions to different environments, they may be more efficient for online adaptation than traditional QD archives, especially when robots can provide information on their current environment.

### Explicit Task Allocation

Explicit task allocation explicitly defines subtasks within a mission, and assigns different robots to them based on task priority and robot capability. This reduces the complexity of the mission but limits behavioural flexibility and requires design-time knowledge as well as frequent explicit communication during run-time. The approach is robust-by-design as task allocation dynamically accounts for the unique and changing task priorities and robot capabilities.

In adaptive specialisation, robots change their roles if they detect insufficient task progress. Representative of this approach are ALLIANCE [27], in which each robot activates a high-level behaviour from its set based on how incoming communications affecting its internal motivations, and data-driven adaptive multi-robot task allocation [41••], which makes low (resp. high) performance of robot  $i$  on task  $j$  being followed by a reduction (resp. increase) in the task specialisation  $s_{ij}$ . The latter approach demonstrated successful task re-allocation in a team composed of ground robots and quadcopters in the Robotarium (see [84]) after parts of the space became no-go or no-fly zones [40, 41••]. Downsides of adaptive task specialisation are the sensitivity to hyperparameters (e.g. time until impatience), the frequent modelling, communication, and global information required to evaluate task progress, and the model assumptions (e.g. control-affine systems).

Ad hoc teamwork [43] considers the related problem of allocating a subset of agents from a pool to participate in solving a particular task, based on their capabilities in the domain from which the task is sampled. While traditionally these capabilities are pre-defined, recent work integrates convolutional neural network based change point detection of capability changes in non-stationary agents [85].

In explicit negotiation, agents bid for their preferred tasks, typically based on the contract-net protocol [86]. Within this approach, TraderBots [15] demonstrated adaptive task allocation on communication failure, partial and complete robot failure, and the reintroduction of a once failed robot, and MURDOCH [42] demonstrated adaptation to new tasks as well as individual robot failures.

Task allocation can be cast as a distributed constrained optimisation problem (DCOP) [44], by allocating each agent in the team to one or more mission variables to optimise the mission's objective, which is composed of different cost functions on subsets of mission variables. Of particular interest is the dynamic DCOP, which allows the DCOP to change over time. Within this framework, it is for example

possible to solve search-and-rescue missions with robustness to run-time addition or removal of tasks, defined as victim locations that are reachable within deadline [45]. Dynamic DCOPs do come with an increase in explicit communication and computation.

### Stigmergy in Swarm Robotics

Stigmergy is a form of implicit communication in which one agent drops signs in the environment to pass information to other agents, an approach which is scalable to the large team sizes found in swarm robotics. Most traditionally, ant colony algorithms leave pheromone trails in the environment to signal where other agents should come [9]. The approach is mostly applicable to foraging or search tasks, where it can account for road blockades, making the approach robust-by-design. However, physically implementing pheromones is always a challenge in real-world applications [87] as the physical signs, when dropped *en masse* should not incur environmental costs, interference with human activities, etc. Recent work has explored the use of light sources but requires either an overhead camera and light projector [88] or locally placed photochromic materials [89].

### Conclusions

Integrating decentralised control, change-detection, and learning is an important challenge facing robot teams for resilience in real-world applications such as search-and-rescue, environmental monitoring, exploration, and pickup-and-delivery.

Change-detection in robot teams comes with unique opportunities and challenges. Team members can cooperatively identify faults in each other as well as actively seek, identify, and track anomalies in the surrounding environment. They share a limited space which comes with localisation and communication constraints (e.g. deadlocks and communication overload), and they must seek to diagnose the faults and provide a solution in a generic manner.

Robot teams face a challenging decision process with decentralised control of actuators and communication channels. Perception-action-communication loops can be learned such that incoming and outgoing messages are processed by the policy of the robot. These rely on imitation learning, which requires an expert to provide new data when the environment changes. MARL is pursued for robot teams in Dec-POMDPs, which has been recently investigated in the context of multi-task RL to improve generalisation to new environments. Despite only demonstrating asymptotic convergence (rather than transfer and sample efficiency), the existence of theoretical guarantees distinguishes the MARL framework from

other frameworks. Embodied evolution evolves the robots directly in the physical environment which avoids the simulation-reality gap but can take many hours to achieve adaptation. The recently emerging field of offline evolution with online adaptation achieves rapid adaptation across a wide range of perturbations by using collaborative learning to update performance priors from behaviours evolved offline. Explicit task allocation breaks the mission down into smaller subtasks and allocates team members to them, which simplifies the adaptation problem but reduces the flexibility of the team by the pre-defined roles. Finally, stigmergy is based on dropping signs in the environment, which scales favourably with team size but is challenging to implement in physical robot teams.

While the above has addressed resilience to a wide array of change types, including team members' capabilities or functionalities, limitations of communication, and environmental dynamics, we still foresee significant challenges in the field of resilient robot teams. First, the integration of change-detection and trial-and-error methods has not been widely explored; to start the adaptation process within a subset of environments, inspiration could be drawn from task identification methods [49–51]. Second, there is a need for methods to obtain reliable performance evaluations without requiring too much evaluation time; there, the evaluation time could be large for promising solutions and small for clearly unsuccessful solutions, similar to the use of virtual energy in embodied evolution. Third, safety is still widely overlooked in the context of robot teams. A promising avenue is to extend single-agent methods for safe reinforcement learning (e.g. shielding [90]) and offline evolution with safe online adaptation (e.g. map-based constrained Bayesian optimisation [83]) to the decentralised multi-agent setting. Indeed, shielding, which replaces any unsafe actions for a given state with a safe action, has recently been extended to factorised shielding [91], which maintains shields for subsets of agents based on a factorisation of the state space. Fourth, there is the challenge of designing algorithms for adaptive robot teams with formal proofs on their sample complexity under various classes of environmental perturbations. Fifth, since realistic applications are ultimately the driver of research in robotics, the design and implementation of convincing case studies is of critical importance. Such studies will highlight new challenges and opportunities, as well as demonstrate the trustworthiness of resilient robot teams to key players in industry and government.

**Funding** This work has been supported by the Engineering and Physical Sciences Research Council under the New Investigator Award grant, EP/R030073/1, and the UKRI Trustworthy Autonomous Systems Hub, EP/V00784X/1, and Axa Research Fund.

## Declarations

**Conflict of Interest** The authors declare no competing interests.

**Human and Animal Rights and Informed Consent** This article does not contain any studies with human or animal subjects performed by any of the authors.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

Papers of particular interest, published recently, have been highlighted as:

- Of importance
- Of major importance

1. Dunbabin M, Marques L. Robotics for environmental monitoring. *IEEE Robot Autom Mag.* 2012;20–23. <https://doi.org/10.2307/j.ctt46nrzt.12>.
2. Rouček T, Pecka M, Čížek P, Petříček T, Bayer J, Šalanský V, et al. DARPA subterranean challenge: Multi-robotic exploration of underground environments. In: Mazal J, Fagiolini A, Vasik P, editors., et al., *Modelling and simulation for autonomous systems*. Cham: Springer International Publishing; 2020. p. 274–90.
3. Montemayor G, Wen JT. Decentralized collaborative load transport by multiple robots. In: *Proceedings of the IEEE international conference on robotics and automation (ICRA 2005)*; 2005;372–377.
4. Brambilla M, Ferrante E, Birattari M, Dorigo M. Swarm robotics: A review from the swarm engineering perspective. *Swarm Intelligence.* 2013;7(1):1–41. <https://doi.org/10.1007/s11721-012-0075-2>.
5. Farinelli A, Iocchi L, Nardi D. Multirobot systems: A classification focused on coordination. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics.* 2004;34(5):2015–28. <https://doi.org/10.1109/TSMCB.2004.832155>.
6. Yan Z, Jouandeau N, Cherif AA. A survey and analysis of multi-robot coordination. *Int J Adv Robot Syst.* 2013;10. <https://doi.org/10.5772/57313>.
- 7.●● Hu TK, Gama F, Chen T, Zheng W, Wang Z, Ribeiro A, et al. Scalable perception-action-communication loops with convolutional and graph neural networks. 2021;(i):1–12. **Recent method for learning perception-action-communication loops in partially observable environments that provides robustness to visual degradation, and changes in team size and communication graph.** [arXiv:2106.13358](https://arxiv.org/abs/2106.13358)
8. Bayindir L. A review of swarm robotics tasks. *Neurocomputing.* 2016;172:292–321. <https://doi.org/10.1016/j.neucom.2015.05.116>.



9. Dorigo M, Bonabeau E, Theraulaz G. Ant algorithms and stigmergy. *Future Generation Computer Systems*. 2000;16(8):851–71. [https://doi.org/10.1016/S0167-739X\(00\)00042-X](https://doi.org/10.1016/S0167-739X(00)00042-X).
10. Kotu V, Deshpande B. Anomaly detection. *Data Science*. 2019;447–465. <https://doi.org/10.1016/b978-0-12-814761-0.00013-7>.
11. Yang J, Zhou K, Li Y, Liu Z. Generalized out-of-distribution detection: A survey. *arXiv preprint*. 2021;1–20. [arXiv:2110.11334](https://arxiv.org/abs/2110.11334).
12. Lazaric A. In: Wiering M, van Otterlo M, editors. *Transfer in reinforcement learning: A framework and a survey*. Berlin Heidelberg: Springer; 2012. p. 143–73.
13. Li F, Zhou M, Ding Y. An adaptive online co-search method with distributed samples for dynamic target tracking. *IEEE Transactions on Control Systems Technology*. 2018;26(2):439–51. <https://doi.org/10.1109/TCST.2017.2669154>.
14. Moratuwage D, Vo BN, Wang D. A hierarchical approach to the Multi-Vehicle SLAM problem. *Proceedings of the international conference on information fusion (FUSION 2012)*. 2012;1119–1125.
15. Dias MB, Zinck M, Zlot R, Stentz A. Robust multirobot coordination in dynamic environments. In: *Proceedings of the IEEE international conference on robotics and automation (ICRA 2004)*; 2004;3435–3442.
16. Parker LE, Kannan B. Adaptive causal models for fault diagnosis and recovery in multi-robot teams. In: *Proceedings of the IEEE international conference on intelligent robots and systems (IROS 2006)*; 2006;2703–2710.
17. Bullard K, Meier F, Kiela D, Pineau J, Foerster J. Exploring zero-shot emergent communication in embodied multi-agent populations. 2020;1–20. [arXiv:2010.15896](https://arxiv.org/abs/2010.15896).
18. Peng P, Wen Y, Yang Y, Yuan Q, Tang Z, Long H, et al. Multi-agent bidirectionally-coordinated nets: Emergence of human-level coordination in learning to play starcraft combat games. *arXiv preprint*. 2017;1–10. [arXiv:1703.10069](https://arxiv.org/abs/1703.10069).
19. Foerster JN, Assael YM, de Freitas N, Whiteson S. Learning to communicate with deep multi-agent reinforcement learning. In: *Proceedings of the conference on neural information processing systems (NeurIPS 2016)*. Barcelona, Spain; 2016;11–16.
20. Jiang J, Lu Z. Learning attentional communication for multi-agent cooperation. *Advances in Neural Information Processing Systems*. 2018;2018-Decem(NeurIPS):7254–7264. [arXiv:1805.07733](https://arxiv.org/abs/1805.07733).
21. Khalastchi E, Kalech M. Fault detection and diagnosis in multi-robot systems: A survey. *Sensors*. 2019;19(18). <https://doi.org/10.3390/s19184019>.
22. Canese L, Cardarilli GC, Di Nunzio L, Fazzolari R, Giardino D, Re M, et al. Multi-agent reinforcement learning: A review of challenges and applications. *Appl Sci*. 2021;11(11). <https://doi.org/10.3390/app11114948>.
23. Ficici SG, Watson RA, Pollack JB. Embodied evolution: A response to challenges in evolutionary robotics. In: *Proceedings of the eighth european workshop on learning robots*; 1999;14–22.
24. ●● Bossens DM, Tarapore D. Rapidly adapting robot swarms with Swarm map-based bayesian optimisation. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2021)*; 2021. p. 9848–9854. **This recent paper demonstrates a first approach to offline evolution with online adaptation, with strong empirical results on various faults and environmental changes.**
25. Gama F, Li Q, Tolstaya E, Prorok A, Ribeiro A. Decentralized control with graph neural networks. 2020;1–14. [arXiv:2012.14906](https://arxiv.org/abs/2012.14906).
26. Yu T, Finn C, Dasari S, Xie A, Zhang T, Abbeel P, et al. One-shot imitation from observing humans via domain-adaptive meta-learning. In: Kress-Gazit H, Srinivasa SS, Howard T, Atanasov N, editors. *Robotics: Science and Systems (RSS 2018)*; 2018;1–10.
27. Parker LE. ALLIANCE: An architecture for fault tolerant multi-robot cooperation. *IEEE Transactions on Robotics and Automation*. 1998;14(2):220–40. <https://doi.org/10.1109/70.681242>.
28. Christensen AL, O’Grady R, Birattari M, Dorigo M. Exogenous fault detection in a collective robotic task. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. 2007;4648 LNAI:555–564. [https://doi.org/10.1007/978-3-540-74913-4\\_56](https://doi.org/10.1007/978-3-540-74913-4_56).
29. Christensen AL, O’Grady R, Birattari M, Dorigo M. Fault detection in autonomous robots based on fault injection and learning. *Autonomous Robots*. 2008;24(1):49–67. <https://doi.org/10.1007/s10514-007-9060-9>.
30. Lau H, Bate I, Cairns P, Timmis J. Adaptive data-driven error detection in swarm robotics with statistical classifiers. *Robotics and Autonomous Systems*. 2011;59(12):1021–35.
31. Tarapore D, Lima PU, Carneiro J, Christensen AL. To err is robotic, to tolerate immunological: fault detection in multi-robot systems. *Bioinspiration & Biomimetics*. 2015;10(1):1–19.
32. ● Tarapore D, Timmis J, Christensen AL. Fault detection in a swarm of physical robots based on behavioral outlier detection. *IEEE Trans Robot*. 2019;35(6):1516–1522. **Recent study demonstrating exogenous fault detection using team members’ feature vector and a cross-regulation model, with strong empirical evidence on physical experiments in a variety of faults in several different tasks.**
33. Christensen AL, Grady RO, Dorigo M. From fireflies to fault-tolerant swarms of robots. *IEEE Transactions on Evolutionary Computation*. 2009;13(4):754–66. <https://doi.org/10.1109/TEVC.2009.2017516>.
34. ● Ferrer EC, Hardjono T, Pentland A, Dorigo M. Secure and secret cooperation in robot swarms. *Sci Robot*. 2021;6(56):1–10. **Recent study demonstrating secure cooperation by detecting compromised robots in the team.** <https://doi.org/10.1126/scirobotics.abf1538>. [arXiv:1904.09266](https://arxiv.org/abs/1904.09266)
35. Hart E, Steyven ASW, Paechter B. Evolution of a functionally diverse swarm via a novel decentralised quality-diversity algorithm. In: *Proceedings of the genetic and evolutionary computation conference (GECCO 2018)*. Kyoto, Japan; 2018;101–108.
36. Bredeche N, Montanier JM, Liu W, Winfield AFT. Environment-driven distributed evolutionary adaptation in a population of autonomous robotic agents. *Mathematical and Computer Modelling of Dynamical Systems*. 2012;18(1):101–29. <https://doi.org/10.1080/13873954.2011.601425>.
37. Silva F, Correia L, Christensen AL. Evolutionary online behaviour learning and adaptation in real robots. *Royal Society Open Science*. 2017;4(7):1–15. <https://doi.org/10.1098/rsos.160938>.
38. Winfield AFT, Erbas MD. On embodied memetic evolution and the emergence of behavioural traditions in Robots. *Memetic Computing*. 2011;3(4):261–70. <https://doi.org/10.1007/s12293-011-0063-x>.
39. Bossens DM, Tarapore D. QED: Using quality-environment-diversity to evolve resilient robot swarms. *IEEE Trans Evol Comput*. 2021;25(2):346–357. <https://doi.org/10.1109/TEVC.2020.3036578>. [arXiv:2003.02341](https://arxiv.org/abs/2003.02341).
40. Emam Y, Mayya S, Notomista G, Bohannon A, Egerstedt M. Adaptive task allocation for heterogeneous multi-robot teams with evolving and unknown robot capabilities. *Proceedings of the IEEE international conference on robotics and automation (ICRA 2020)*. 2020;7719–7725. <https://doi.org/10.1109/ICRA40945.2020.9197283>. [arXiv:2003.03344](https://arxiv.org/abs/2003.03344).

41. ●● Emam Y, Notomista G, Glotfelter P, Egerstedt M. Data-driven adaptive task allocation for heterogeneous multi-robot teams using robust control barrier functions. In: Proceedings of the IEEE international conference on robotics and automation (ICRA 2021). IEEE; 2021. p. 9124–9130. **Recent paper on adaptive task specialisation with strong empirical results for heterogeneous teams in Robotarium experiments with capability loss.**
42. Gerkey BP, Mataric MJ. Sold!: Auction methods for multirobot coordination. *IEEE Transactions on Robotics and Automation*. 2002;18(5):758–68. <https://doi.org/10.1109/TRA.2002.803462>.
43. Stone P, Kaminka GA, Rosenschein JS. Ad hoc autonomous agent teams: Collaboration without pre-coordination. In: Proceedings of the AAAI conference on artificial intelligence (AAAI 2010); 2010;1504–1509.
44. Fioretto F, Pontelli E, Yeoh W. Distributed constraint optimization problems and applications: A survey. *J Artif Intell Res*. 2018;61:623–698. <https://doi.org/10.1613/jair.5565>. [arXiv:1602.06347](https://arxiv.org/abs/1602.06347).
45. Ramchurn SD, Farinelli A, MacArthur KS, Jennings NR. Decentralized coordination in RoboCup Rescue. *Computer Journal*. 2010;53(9):1447–61. <https://doi.org/10.1093/comjnl/bxq022>.
46. Omidshafiei S, Pazis J, Amato C, How JP, Vian J. Deep decentralized multi-task multi-agent reinforcement learning under partial observability. In: Proceedings of the international conference on machine learning (ICML 2017), 2017;4108–4122.
47. Sukhbaatar S, Szlam A, Fergus R. Learning multiagent communication with backpropagation. In: Proceedings of the conference on neural information processing systems (NeurIPS 2016). Barcelona, Spain; 2016;1–9.
48. ●● Jung S, Yun WJ, Kim J, Kim JH. Coordinated multi-agent deep reinforcement learning for energy-aware UAV-based big-data platforms. *Electronics*. 2021;10(5):1–15. **Recent study that demonstrates CommNet for energy sharing in multi-UAV systems for distributed data processing.** <https://doi.org/10.3390/electronics10050543>.
49. Fifty C, Amid E, Zhao Z, Yu T, Anil R, Finn C. Efficiently identifying task groupings for multi-task learning. In: Proceedings of the conference on neural information processing systems (NeurIPS 2021); 2021;1–22.
50. Lomonaco V, Desai K, Culurciello E, Maltoni D. Continual reinforcement learning in 3D non-stationary environments. In: IEEE computer society conference on computer vision and pattern recognition workshops (CVPRW 2020); 2020;999–1008.
51. Milan K, Veness J, Kirkpatrick J, Hassabis D, Koop A, Bowling M. The forget-me-not process. In: Proceedings of the conference on neural information processing systems (NeurIPS 2016); 2016;3702–3710.
52. Saeedi S, Trentini M, Seto M, Li H. Multiple-robot simultaneous localization and mapping: A review. *Journal of Field Robotics*. 2014;33(1):1–17. <https://doi.org/10.1002/rob>.
53. Tipaldi GD, Meyer-Delius D, Burgard W. Lifelong localization in changing environments. *International Journal of Robotics Research*. 2013;32(14):1662–78. <https://doi.org/10.1177/0278364913502830>.
54. Saldana D, Assuncao R, Campos MFM. A distributed multi-robot approach for the detection and tracking of multiple dynamic anomalies. In: Proceedings of the IEEE international conference on robotics and automation (ICRA 2015); 2015;1262–1267.
55. Salam T, Hsieh MA. Adaptive sampling and reduced-order modeling of dynamic processes by robot teams. *IEEE Robotics and Automation Letters*. 2019;4(2):477–84. <https://doi.org/10.1109/LRA.2019.2891475Y>.
56. Silva F, Urbano P, Oliveira S, Christensen AL. OdNEAT: An algorithm for distributed online, onboard evolution of robot behaviours. In: Proceedings of the international conference on the simulation and synthesis of living systems (ALIFE 2012); 2012;251–258.
57. Roumeliotis SI, Sukhatme GS, Bekey GA. Sensor fault detection and identification in a mobile robot. In: Proceedings of the IEEE international conference on intelligent robots and systems (IROS 1998), 3; 1998;1383–1387.
58. Van Eykeren L, Chu QP. Nonlinear model-based fault detection for a hydraulic actuator. In: AIAA guidance, navigation, and control conference; 2011;1–8.
59. Skoundrianos EN, Tzafestas SG. Fault diagnosis on the wheels of a mobile robot using local model neural networks. *IEEE Robotics & Automation Magazine*. 2004;11(3):83–90. <https://doi.org/10.1109/mra.2004.1337829>.
60. Terra MH, Tinós R. Fault detection and isolation in robotic manipulators via neural networks: A comparison among three architectures for residual analysis. *Journal of Robotic Systems*. 2001;18(7):357–74. <https://doi.org/10.1002/rob.1029>.
61. Owens NDL, Greensted A, Timmis J, Tyrrell A. The receptor density algorithm. *Theoretical Computer Science*. 2013;481:51–73. <https://doi.org/10.1016/j.tcs.2012.10.057>.
62. Carneiro J, Leon K, Caramalho Í, Van Den Dool C, Gardner R, Oliveira V, et al. When three is not a crowd: A Crossregulation Model of the dynamics and repertoire selection of regulatory CD4+ T cells. *Immunological Reviews*. 2007;216(1):48–68. <https://doi.org/10.1111/j.1600-065X.2007.00487.x>.
63. León K, Pérez R, Lage A, Carneiro J. Three-cell interactions in t cell-mediated suppression? a mathematical analysis of its quantitative implications. *The Journal of Immunology*. 2001;166(9):5356–65. <https://doi.org/10.4049/jimmunol.166.9.5356>.
64. Kim K, Gu Y, Son J, Zha S, Ermo S. Domain adaptive imitation learning. In: Proceedings of the international conference on machine learning (ICML 2020); 2020;5242–5251.
65. Zhang K, Yang Z, Liu H, Zhang T, Başar T. Fully decentralized networked agents. In: Proceedings of the international conference on machine learning (ICML 2018); 2018; 5872–5881.
66. Oliehoek FA. Decentralized POMDPs. In: *Reinforcement Learning: State of the Art*; 2013;471–503.
67. Pynadath DV, Tambe M. The communicative multiagent team decision problem: Analyzing teamwork theories and models. *Journal of Artificial Intelligence Research*. 2002;16:389–423.
68. Matignon L, Laurent GJ, Le Fort-Piat N. Hysteretic Q-Learning : An algorithm for decentralized reinforcement learning in cooperative multi-agent teams. In: Proceedings of the IEEE international conference on intelligent robots and systems (IROS 2007). IEEE; 2007;64–69.
69. Hausknecht M, Stone P. Deep recurrent q-learning for partially observable MDPs. In: AAAI fall symposium series. AAAI; 2015;29–37.
70. Rusu AA, Colmenarejo SG, Gülçehre Ç, Desjardins G, Kirkpatrick J, Pascanu R, et al. Policy distillation. In: Proceedings of the international conference on learning representations (ICLR 2016); 2016;1–13.
71. Sunehag P, Lever G, Gruslys A, Czarnecki WM, Zambaldi V, Jaderberg M, et al. Value-decomposition networks for cooperative multi-agent learning based on team reward. In: Proceedings of the international joint conference on autonomous agents and multiagent systems (AAMAS 2018), 3;2018;2085–2087.
72. Rashid T, Samvelyan M, Faruqhar CSdWG, Foerster J, Whiteson S. QMIX: Monotonic value function factorisation for deep multi-agent reinforcement learning. In: Proceedings of the

- international conference on machine learning (ICML 2018); 2018;4295–4304.
73. Foerster JN, Assael YM, de Freitas N, Whiteson S. Learning to communicate to solve riddles with deep distributed recurrent Q-networks. arXiv preprint. 2016;1–10. [arXiv:1602.02672](https://arxiv.org/abs/1602.02672).
  74. Foerster JN, Farquhar G, Afouras T, Nardelli N, Whiteson S. Counterfactual multi-agent policy gradients. In: The AAAI conference on artificial intelligence (AAAI 2018); 2018;2974–2982.
  75. Gupta JK, Egorov M, Kochenderfer M. Cooperative multi-agent control using deep reinforcement learning. In: Sukthankar G, Rodriguez-Aguilar J, editors. Proceedings of the international joint conference on autonomous agents and multiagent systems (AAMAS 2017); 2017;66–83.
  76. Watkins CJCH, Dayan P. Q-learning. *Machine Learning*. 1992;8(3–4):279–92. <https://doi.org/10.1007/BF00992698>.
  77. Stanley KO, Miikkulainen R. Evolving neural networks through augmenting topologies. *Evolutionary Computation*. 2002;10(2):99–127. <https://doi.org/10.1162/106365602320169811>.
  78. Jakobi N, Husbands P, Harvey I. Noise and the reality gap: The use of simulation in evolutionary robotics. In: Morán F, Moreno A, Merelo JJ, Chacón P, editors. *Advances in Artificial Life (ECAL 1995)*. Lecture Notes in Computer Science (Lecture Notes in Artificial Intelligence). vol. 929. Springer, Berlin, Heidelberg; 1995;704–720.
  79. Mouret JB, Clune J. Illuminating search spaces by mapping elites. arXiv preprint. 2015;1–15.
  80. Lehman J, Stanley KO. Evolving a diversity of creatures through novelty search and local competition. In: Proceedings of the genetic and evolutionary computation conference (GECCO 2011). ACM, New York; 2011;211–218.
  81. Cully A, Clune J, Tarapore D, Mouret JB. Robots that can adapt like animals. *Nature*. 2015;521(7553):503–7. <https://doi.org/10.1038/nature14422>[arXiv:1407.3501](https://arxiv.org/abs/1407.3501).
  82. Dalin E, Desreumaux P, Mouret JB, Dalin EE. Learning and adapting quadruped gaits with the “Intelligent Trial & Error” algorithm. In: IEEE ICRA 2019 workshop on learning legged locomotion. Montreal, Canada; 2019;1–2.
  83. Papaspyros V, Chatzilygeroudis K, Vassiliades V, Mouret JB. Safety-aware robot damage recovery using constrained bayesian optimization and simulated priors. In: NeurIPS 2016 Workshop on Bayesian Optimization; 2016;1–5.
  84. Pickem D, Glotfelter P, Wang L, Mote M, Ames A, Feron E, et al. The Robotarium: A remotely accessible swarm robotics research testbed. Proceedings of the IEEE international conference on robotics and automation (ICRA 2017). 2017;1699–1706. <https://doi.org/10.1109/ICRA.2017.7989200>. [arXiv:1609.04730](https://arxiv.org/abs/1609.04730).
  85. Ravula M, Alkoby S, Stone P. Ad hoc teamwork with behavior switching agents. In: Proceedings of the international joint conference on artificial intelligence (IJCAI 2019); 2019;550–556.
  86. Smith RG. The contract net protocol: High-level communication and control in a distributed problem solver. *IEEE Transactions on Computers*. 1980;C-29(12):1104–13.
  87. Zedadra O, Jouandeau N, Seridi H, Fortino G. Multi-Agent Foraging: state-of-the-art and research challenges. *Complex Adapt Syst Model*. 2017;5(1). <https://doi.org/10.1186/s40294-016-0041-8>.
  88. Hunt ER, Jones S, Hauert S. Testing the limits of pheromone stigmergy in high-density robot swarms. *R Soc Open Sci*. 2019;6(11). <https://doi.org/10.1098/rsos.190225>.
  89. Salman M, Garzón Ramos D, Hasselmann K, Birattari M. Phormica: Photochromic pheromone release and detection system for stigmergic coordination in robot swarms. *Front Robot AI*. 2020;7:1–15. <https://doi.org/10.3389/frobt.2020.591402>.
  90. Alshiekh M, Bloem R, Ehlers R, Könighofer B, Niekum S, Topcu U. Safe reinforcement learning via shielding. In: Proceedings of the AAAI conference on artificial intelligence (AAAI-18); 2018;2669–2678.
  91. ElSayed-Aly I, Bharadwaj S, Amato C, Ehlers R, Topcu U, Feng L. Safe multi-agent reinforcement learning via shielding. In: Proceedings of the international joint conference on autonomous agents and multiagent systems (AAMAS 2021); 2021;483–491.
- Publisher’s Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.